



LUND UNIVERSITY

Robust Estimation of Motion Parameters and Scene Geometry Minimal Solvers and Convexification of Regularisers for Low-Rank Approximation Valtonen Örnthag, Marcus

2021

Document Version:
Publisher's PDF, also known as Version of record

[Link to publication](#)

Citation for published version (APA):
Valtonen Örnthag, M. (2021). *Robust Estimation of Motion Parameters and Scene Geometry: Minimal Solvers and Convexification of Regularisers for Low-Rank Approximation*. [Doctoral Thesis (compilation), Mathematics (Faculty of Engineering)]. Lund University, Faculty of Engineering.

Total number of authors:
1

General rights

Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

Robust Estimation of Motion Parameters and Scene Geometry

Minimal Solvers and Convexification of
Regularisers for Low-Rank Approximation

Marcus Valtonen Örnå



LUND
UNIVERSITY

ACADEMIC THESIS

which, by due permission of the Faculty of Engineering of Lund University, will be publicly defended on Friday the 14th of May, 2021, at 13:15 in lecture hall MH:H, Centre for Mathematical Sciences, Sölvegatan 18, Lund, for the degree of Doctor of Philosophy in Engineering.

Faculty opponent

Dr. Andrew Fitzgibbon, Microsoft Research Cambridge, United Kingdom

Organization Centre for Mathematical Sciences Faculty of Engineering Lund University Box 118 SE-221 00 Lund	Document name DOCTORATAL THESIS IN MATHEMATICAL SCIENCES	
	Date of issue May 2021	
	Sponsoring organization	
Author(s) Marcus Valtonen Örnthög	Supervisors Anders Heyden, Carl Olsson	
Title and subtitle Robust Estimation of Motion Parameters and Scene Geometry: Minimal Solvers and Convexification of Regularisers for Low-Rank Approximation		
Abstract In the dawning age of autonomous driving, accurate and robust tracking of vehicles is a quintessential part. This is inextricably linked with the problem of <i>Simultaneous Localisation and Mapping</i> (SLAM), in which one tries to determine the position of a vehicle relative to its surroundings without prior knowledge of them. The more you know about the object you wish to track—through sensors or mechanical construction—the more likely you are to get good positioning estimates. In the first part of this thesis, we explore new ways of improving positioning for vehicles travelling on a planar surface. This is done in several different ways: first, we generalise the work done for monocular vision to include two cameras, we propose ways of speeding up the estimation time with polynomial solvers, and we develop an auto-calibration method to cope with radially distorted images, without enforcing pre-calibration procedures. We continue to investigate the case of constrained motion—this time using auxiliary data from <i>inertial measurement units</i> (IMUs) to improve positioning of <i>unmanned aerial vehicles</i> (UAVs). The proposed methods improve the state-of-the-art for partially calibrated cases (with unknown focal length) for indoor navigation. Furthermore, we propose the first-ever real-time compatible minimal solver for simultaneous estimation of radial distortion profile, focal length, and motion parameters while utilising the IMU data. In the third and final part of this thesis, we develop a bilinear framework for low-rank regularisation, with global optimality guarantees under certain conditions. We also show equivalence between the linear and the bilinear framework, in the sense that the objectives are equal. This enables users of <i>alternating direction method of multipliers</i> (ADMM)—or other subgradient or splitting methods—to transition to the new framework, while being able to enjoy the benefits of second order methods. Furthermore, we propose a novel regulariser fusing two popular methods. This way we are able to combine the best of two worlds by encouraging bias reduction while enforcing low-rank solutions.		
Key words computer vision, visual odometry, simultaneous localization and mapping, minimal solvers, convex relaxations, structure from motion		
Classification system and/or index terms (if any)		
Supplementary bibliographical information		Language English
ISSN and key title 1404-0034		ISBN 978-91-7895-769-9 (printed) 978-91-7895-770-5 (electronic)
Recipient's notes	Number of pages xx+367	Price
	Security classification	

I, the undersigned, being the copyright owner of the abstract of the above-mentioned dissertation, hereby grant to all reference sources the permission to publish and disseminate the abstract of the above-mentioned dissertation.

Signature: Marcus Valtonen Örnthög

Date: 2021-03-25

ROBUST ESTIMATION OF MOTION PARAMETERS AND SCENE GEOMETRY

MINIMAL SOLVERS AND CONVEXIFICATION OF
REGULARISERS FOR LOW-RANK APPROXIMATION

MARCUS VALTONEN ÖRNHAG



LUND UNIVERSITY

Centre for Mathematical Sciences
Mathematics

Mathematics
Centre for Mathematical Sciences
Lund University
Box 118
SE-221 00 Lund
Sweden
<http://www.maths.lu.se/>

Doctoral Theses in Mathematical Sciences 2021:1
ISSN 1404-0034

ISBN 978-91-7895-769-9 (printed)
ISBN 978-91-7895-770-5 (electronic)
LUTFMA-1070-2021

© Marcus Valtonen Örnberg, 2021

Printed in Sweden by MediaTryck, Lund 2021

Abstract

In the dawning age of autonomous driving, accurate and robust tracking of vehicles is a quintessential part. This is inextricably linked with the problem of *Simultaneous Localisation and Mapping* (SLAM), in which one tries to determine the position of a vehicle relative to its surroundings without prior knowledge of them. The more you know about the object you wish to track—through sensors or mechanical construction—the more likely you are to get good positioning estimates. In the first part of this thesis, we explore new ways of improving positioning for vehicles travelling on a planar surface. This is done in several different ways: first, we generalise the work done for monocular vision to include two cameras, we propose ways of speeding up the estimation time with polynomial solvers, and we develop an auto-calibration method to cope with radially distorted images, without enforcing pre-calibration procedures.

We continue to investigate the case of constrained motion—this time using auxiliary data from *inertial measurement units* (IMUs) to improve positioning of *unmanned aerial vehicles* (UAVs). The proposed methods improve the state-of-the-art for partially calibrated cases (with unknown focal length) for indoor navigation. Furthermore, we propose the first-ever real-time compatible minimal solver for simultaneous estimation of radial distortion profile, focal length, and motion parameters while utilising the IMU data.

In the third and final part of this thesis, we develop a bilinear framework for low-rank regularisation, with global optimality guarantees under certain conditions. We also show equivalence between the linear and the bilinear framework, in the sense that the objectives are equal. This enables users of *alternating direction method of multipliers* (ADMM)—or other subgradient or splitting methods—to transition to the new framework, while being able to enjoy the benefits of second order methods. Furthermore, we propose a novel regulariser fusing two popular methods. This way we are able to combine the best of two worlds by encouraging bias reduction while enforcing low-rank solutions.

Populärvetenskaplig sammanfattning

De praktiska tillämpningarna av datorseende och artificiell intelligens har ökat markant det senaste decenniet och finns bland annat inom säkerhetsbranschen, fordonsindustrin, sjukvården, jordbruksindustrin och finansvärlden. Även om applikationerna är många är den grundläggande matematiken bakom ofta densamma—detta ådagalägges i praktiken av denna avhandling där allt från industrirobotar, drönare och diverse deformande objekt huserar.

Avhandlingen är uppdelad i tre delar. I den första delen ägnar vi oss åt att studera ett problem som uppstår då man önskar att navigera i ett område karakteriserat av plana ytor. Typiskt finner vi oss i sådana situationer då vi är inomhus; här finns väggar, golv och tak. Vi kan t.ex. tänka oss en industrirobot som rör sig på ett golv i en fabrik, för att nämna en praktiskt applicerbar situation. Flertalet metoder existerar redan idag för självstyrande fordon och nästan alltid är det fördelaktigt att använda oss av all information vi kan få om den tänkta miljön. Om vi nu vet att det finns plana ytor, då bör vi också använda detta antagande när vi skapar algoritmerna. Vi studerar det monokulära fallet, d.v.s. när en kamera finns tillgänglig, samt det binokulära fallet, då två kameror finns tillgängliga. Det visar sig att när vi tar hänsyn till de plana ytorna blir metoderna mer stabila och positionsbestämningarna mer exakta. I det binokulära fallet visar vi att det är möjligt att förbättra resultaten ytterligare, genom att anta att kamerorna verkar i en stelkroppsrörelse, vilket innebär att kamerornas orientering och avstånd till varandra förblir oförändrade oberoende av hur fordonet rör sig. Detta kan garanteras genom en mekanisk konstruktion som låser fast kamerorna i dessa lägen.

I den andra delen fördjupar vi oss återigen i inomhuspositionering, men denna gång specifikt för drönare. Den stora skillnaden i detta fallet är att drönare har andra sensorer som vi kan dra nytta av. I vårt fall antar vi att drönaren har en IMU (ibland tröghetsmåttighet på svenska), som kan hjälpa till att uppskatta drönarens rotation relativt gravitationsriktningen. Detta gör att antalet frihetsgrader minskar, men ger samtidigt upphov till att de styrande ekvationerna försvåras och ej längre är angripbara med elementära metoder. Den bakomliggande matematiken som gör det möjligt att lösa des-

sa typer av ekvationer är densamma som i fallet för industriroboten. Vi visar att de nya ekvationerna också går att lösa, samt att detta kan göras tillräckligt snabbt för att i realtid kunna processera resultaten på drönaren. Dessutom är den föreslagna metoden numeriskt stabil och noggrannheten i positioneringsestimaten är bättre än för befintliga metoder.

Slutligen studerar vi ett område inom matematiken som kallas lågrangsapproximation, i vilket flera intressanta fall av datorseende finns representerade. Applikationerna är närbesläktade de positioneringsproblem som studerades i de första två delarna av avhandlingen, men metoderna är annorlunda och vi tillåter nu även dynamiska objekt. Detta kan vara människor och djur som rör sig eller objekt som deformeras. Återigen finner vi att en och samma bakomliggande matematiska grund leder till många olika tillämpningsområden. I dessa fall ökar komplexiteten och problemen är ofta underbestämda, d.v.s. att det finns väldigt många lösningar som ger ett bra återprojiceringsfel, men av dessa är få fysikaliskt rimliga. Av detta kan man dra slutsatsen att återprojiceringsfelet, d.v.s. skillnaden mellan de uppmätta tvådimensionella bildpunkterna och de skattade (från en 3D-modell), ej ger tillräcklig information för att erhålla de sökta positionerna i rummet. För att kunna utesluta falska lösningar kan man introducera regulariserare, som t.ex. straffar ofysikaliska lösningar. Vårt bidrag utgör ett nytt ramverk för att effektivt kunna använda regulariserare inom flertalet applikationer. Detta ramverk använder andra ordningens approximationer lokalt, för snabbare konvergens, men har även garantier för globala optima i några fall, vilket är ovanligt för denna typ av optimeringsmetoder.

Acknowledgements

I would like to thank my supervisor Anders Heyden for getting me started with what would become my licentiate thesis and the first part of this thesis. His input and guidance were invaluable at this stage of my academic career. This would eventually lead to me exploring neighbouring fields, including the work on UAVs, during which he continued to give feedback and encouragement. Secondly, I would like to thank my co-supervisor Carl Olsson for introducing me to the world of low-rank approximations and how it is connected to computer vision. We have had fruitful discussions and some problems we discussed have consumed me entirely, deprived me of my sleep, yet given me complete satisfaction upon any sign of progress. These years as a PhD candidate would not have been the same without these challenges. I have met a lot of benevolent people at the Centre for Mathematical Sciences at Lund University, with whom I have had both encouraging and interesting discussions. Thank you all for this time!

I would like to take this opportunity to give a special thanks to Mårten Wadenbäck, who has been my co-author on four occasions so far. His insights in the field have strengthened the content of these papers, as well as their presentation. I was lucky to become office neighbours with Patrik Persson, whose practical knowledge in UAVs and UAV navigation led to two interesting papers. Without his competence, the material in those papers would never have got past the drawing board. Last, but not least, I would like to thank José Pedro Iglesias for the discussions we had on the bilinear framework.

This work would not have been possible without the support of my family, especially my wife Savarin, who (at least initially) gave me the strength to pursue my dreams (while, in hindsight, has regretted it repeatedly). I cannot end this page without mentioning my biggest source of inspiration—my favourite superhero, the spark in my eye—Björn, my son, who is the sole reason why this thesis can never become more than my second largest achievement.

Funding I am grateful for the financial support I have received during my time in Lund. This funding has come from the Swedish Research Council through grant no. 2015-05639 ‘Visual SLAM based on Planar Homographies’, and the strategic research project ELLIIT.

Contents

Abstract	v
Populärvetenskaplig sammanfattning	vii
Acknowledgements	ix
Contents	xi
1 Thesis Overview	1
2 Modelling Cameras	5
2.1 The Pinhole Camera Model	5
2.2 Physical Cameras	7
2.3 Correcting Lens Distortion	8
2.4 Finding Point Correspondences	9
3 Homographies	13
3.1 Properties of Homographies	13
3.2 Direct Linear Transformation	14
4 Epipolar Geometry	17
4.1 The Fundamental Matrix	17
4.2 Why Homographies?	20
4.3 Calibrated Cameras and the Essential Matrix	21
4.4 The Eight-Point Algorithm	22
5 Simultaneous Localisation and Mapping	25
5.1 Motivation and History	25
5.2 State-of-the-Art	26

xi

5.3	Constrained Motion	26
5.3.1	The General Planar Motion Model	26
5.3.2	Utilising IMU Data	32
6	Minimal Problems in Computer Vision	35
6.1	Random Sampling Consensus	35
6.2	Minimal Solvers	36
6.3	Solving Polynomial Systems of Equations	36
6.4	The Action Matrix Method	37
7	Bundle Adjustment	39
7.1	The Geometric Reprojection Error	39
7.2	The Levenberg–Marquardt Algorithm	40
7.3	Separable Nonlinear Least-Squares Problems	41
7.4	Schur Complement	42
7.5	Alternative Approaches	44
8	Low-Rank Matrix Completion and Applications to Computer Vision	45
8.1	Convexification	46
8.2	The Nuclear Norm Heuristic	48
8.3	Non-Convex Relaxations	49
8.4	Bilinear Parameterisation	51
9	Closing Words	53
10	Overview of Papers	55
	References	59

Papers

I	Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies	67
1	Introduction	69
2	Related Work	70
3	Theory	70
3.1	Problem Geometry	70
3.2	Camera Parametrisation	71
3.3	Homographies	72
3.4	Parameter Recovery	72
4	Experiments	76
4.1	Synthetic Data	76
4.2	Homography Estimation	76
4.3	Parameter Recovery	77
4.4	Test Cases	77
5	Conclusion	78
	References	84
	Appendix: Proof of Theorem 1	87
II	Generalisation of Parameter Recovery in Binocular Vision	89
1	Introduction	91
2	Related Work	92
3	Theory	92
3.1	Problem Geometry	92
3.2	The Impact of Different Image Planes	94
3.3	Camera Matrices	95
3.4	Homographies	95
4	Parameter Recovery	96
4.1	Pure Translation	96
4.2	General Planar Motion	97
4.3	Solving for the Relative Translation and Unknown Scale	98
5	Experiments	99
5.1	Initial Calibration vs. General Motion	100
5.2	Error vs. Noise	100
5.3	Evaluation on the KITTI Dataset	102
6	Conclusion	104
	References	106

III Fast Non-Minimal Solvers for Planar Motion Compatible Homographies 109

1	Introduction	111
2	Related Work	112
3	Planar Motion	113
3.1	Problem Geometry	113
3.2	Parameter Recovery	115
4	Compatible Homographies	115
4.1	The Action Matrix Method	116
4.2	Necessary Conditions	117
5	Sufficient Conditions	118
6	Non-Minimal Solvers	118
6.1	Parameterizing the Null Space	120
7	Experiments	120
7.1	Noise Sensitivity	120
7.2	Speed Evaluation	121
7.3	Synthetic Image Evaluation	122
7.4	Planar Motion Evaluation	123
7.5	Evaluation on the KITTI Dataset	123
8	Conclusions	125
	References	126
	Appendix	129

IV Efficient Radial Distortion Correction for Planar Motion 133

1	Introduction	135
2	Related Work	137
2.1	Homography Estimation	137
2.2	Modelling Radial Distortion	137
3	The General Planar Motion Model	138
4	Polynomial Solvers	140
4.1	A Non-Minimal Relaxation (4 point)	140
4.2	Minimal Solver with Known Tilt (2 point)	142
5	Experiments	144
5.1	Synthetic Data	144
5.2	Numerical Stability	144
5.3	Noise Sensitivity	145
5.4	Image Stitching	146
5.5	Application to Visual Odometry	147
5.6	Application to Aerial Imagery	150
6	Conclusions	151

References	151
V Enforcing the General Planar Motion Model: Bundle Adjustment for Planar Scenes	155
1 Introduction	157
2 Related Work	159
3 Theory	161
3.1 Problem Geometry	161
3.2 Camera Parameterisation	162
4 Prerequisites	162
4.1 Geometric Reprojection Error	162
4.2 The Levenberg–Marquardt Algorithm	163
4.3 Obtaining an Initial Solution for the Camera Parameters	163
4.4 Obtaining an Initial Solution for the Scene Points	164
5 Planar Motion Bundle Adjustment	165
5.1 Block Structure of the Jacobian	165
5.2 Utilising the Sparse Structure	165
6 Experiments	167
6.1 Initial Solution	167
6.2 Impact of Pre-Processing Steps	168
6.3 Bundle Adjustment Comparison	170
7 Conclusion	172
References	172
VI Minimal Solvers for Indoor UAV Positioning	177
1 Introduction	179
2 Related Work	181
3 Incorporating the IMU Data	182
3.1 Navigation Using the Ground Plane	184
3.2 The Calibrated Case	185
4 Partially Calibrated Cases for Ground Plane Navigation	185
4.1 Equal and Unknown Focal Length (fHf , 2.5-point)	185
4.2 One Unknown Focal Length (Hf , 2.5-point)	187
4.3 Different and Unknown Focal Lengths (f_1Hf_2 , 3-point)	187
4.4 Synthetic Experiments	188
5 Experiments	188
5.1 Real Data	191
6 Conclusions	193
References	193

VII Efficient Real-Time Radial Distortion Correction for UAVs	199
1 Introduction	201
2 Related Work	204
3 Embracing the IMU Drift	204
3.1 New Assumptions on the Homography	205
3.2 Benefits of this Approach	207
4 Polynomial Solvers	208
4.1 Calibrated Case (1.5-point)	208
4.2 Equal and Unknown Focal Length (fHf , 2-point)	208
4.3 Equal and Unknown Focal Length and Radial Distortion Coefficient ($frHfr$, 2.5-point)	209
5 Experiments	211
5.1 Numerical Stability and Noise Sensitivity	211
5.2 Speed Evaluation	212
5.3 Real Data	213
6 Conclusions	217
References	217
VIII Bilinear Parameterization for Differentiable Rank-Regularization	223
1 Introduction	225
1.1 Related Work	229
2 Non-Convex Penalties and Shrinking Bias	230
3 Overparameterization and Optimality	233
4 An Iterative Reweighted VarPro Algorithm	235
5 Experiments	236
5.1 Synthetic Missing Data Problem	237
5.2 pOSE: Pseudo Object Space Error	239
5.3 Non-Rigid Structure from Motion	240
6 Conclusions	241
References	241
Supplementary Material	247
A Proofs	247
B Implementation Details	254
C Additional Experiments on Real Data	255
References	260

IX	Differentiable Fixed-Rank Regularisation using Bilinear Parameterisation	261
1	Introduction	263
2	Related Work	264
3	Differentiable Regularisers	266
3.1	Bilinear Parameterisation and Pseudo-Singular Values	266
3.2	A Bilinear Fixed-Rank Regulariser	266
3.3	Differentiability	267
4	Optimality Conditions	268
5	Implementation	269
6	Why Overparameterisation?	271
7	Non-Rigid Structure from Motion	272
8	Conclusions	274
	References	274
	Supplementary Material	279
A	Proofs	279
B	Implementation Details	283
References	285
X	A Unified Optimization Framework for Low-Rank Inducing Penalties	287
1	Introduction	289
1.1	Related Work	290
2	Problem Formulation and Motivation	291
2.1	Controlled Bias and Rank Selection	292
2.2	The Quadratic Envelope	294
3	A New Family of Functions	294
4	Finding the Maximizing Sequence	295
5	ADMM and the Proximal Operator	297
6	Experiments	299
6.1	Synthetic Missing Data	300
6.2	Non-Rigid Deformation with Missing Data	300
6.3	Motion Capture	301
7	Conclusions	304
	References	306
	Supplementary Material	309
A	Von Neumann’s Trace Theorem	309
B	The Fenchel Conjugate	310
C	The Convex Envelope	311
D	Obtaining the Maximizing Sequences	311

E	Proof of Theorem 1	312
F	Modifying Algorithm 1	313
G	Convergence: Motion Capture	314
	References	314

XI Accurate Optimization of Weighted Nuclear Norm for Non-Rigid Structure from Motion 315

1	Introduction	317
1.1	Related Work and Contributions	319
2	Bilinear Parameterization Penalties	321
2.1	Extreme Points and Optimality	322
3	Non-Square Matrices	323
4	Linear Objectives – Weighted Nuclear Norms	324
5	Experiments	325
5.1	Pseudo Object Space Error (pOSE) and Non-Rigid Structure from Motion	326
5.2	Low-Rank Matrix Recovery with pOSE Errors	327
5.3	Non-Rigid Structure Recovery	328
6	Conclusions	333
	References	333

XII Bilinear Parameterization for Non-Separable Singular Value Penalties 339

1	Introduction	341
1.1	Framework and Contributions	343
1.2	Related Work	345
2	Overview of the Approach	347
2.1	A Continuous Relaxation	347
2.2	Bilinear Parameterization	348
2.3	Quadratic Approximation and Optimization	348
2.4	Overview of Algorithm	350
3	Main Theoretical Result	351
3.1	Theoretical Background	351
3.2	Proof of Theorem 1	352
4	Experiments	355
4.1	Matrix Recovery	355
4.2	Application to Non-Rigid SfM	357
5	Conclusions	358
	References	359
	Supplementary Material	363

A	Proof of Lemma 1	363
B	Proof of the Claim in Section 2.3	364
C	Detailed Algorithm	364
D	Parameters Used in Experiments	365
E	Convergence Plots	365
References	367

Chapter 1

Thesis Overview

Computer vision is a broad field with many applications. Simply put, whenever you are trying to automatically gain understanding from images or videos you have entered the field of computer vision. What kind of knowledge you seek from the input may vary—it can be positioning, building 3D models, tracking and recognising objects, to name a few examples. Many of the applications directly impact society and have done so for a long time. An early example is the invention of the *optical character recognition* (OCR) scanner, which was patented in 1931, and created to aid the visually-impaired. Fast forwarding the story ninety years into the future, to modern society, we see applications of computer vision and machine learning in many diverse industries, including the security sector, the automotive industry, health care, agriculture, and the financial sector. While many problems may be considered to be solved, the demand for new technical solutions incite researchers from both academia and industry to pursue the continued growth of the field, which is expanding rapidly.

This thesis consists of three parts. In the first part (Papers I–V) we study a case of constrained motion, namely the *general planar motion model*. This model is applicable where planar objects dominate the scene geometry, *e.g.* indoor environments, where floors, walls and ceilings are present. Previous authors have investigated this model for monocular vision; however, the case of binocular vision was not studied prior to Papers I and II.

In the second part (Papers VI and VII) we consider another case of constrained motion, namely navigation using unmanned aerial vehicles (UAVs), commonly referred to as drones. Here auxiliary data is available from sensors, specifically an inertial measurement unit (IMU). The IMU typically contains an accelerometer and a gyroscope that measure acceleration and angular velocity, with respect to its frame. From these measurements, it is possible to determine (at least) two degrees of freedom—the roll and pitch

angles. Additionally, some IMUs contain a magnetometer that measures the magnetic field. Modern algorithms can fuse magnetometer data in order to estimate the yaw angle [40]; however, these measurements might be corrupted by indoor environments as well as the magnetic field surrounding the UAV motors [22, 48]. The external magnetic distortions are more common in indoor environments due to the presence of electronic devices, iron reinforcements in buildings etc.

The third part (Papers VIII–XII) treats a different field, namely low-rank approximation. Here, we develop a broad framework, in the sense that it is compatible with many core applications in pattern recognition, computer vision and machine learning. We will focus on applications of non-rigid structure-from-motion, but will also look at applications concerning rigid structure-from-motion, background extraction and photometric stereo.

The following papers form the basis of this thesis:

Main papers:

- I M. Valtonen Örnthag and A. Heyden. “Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Funchal, Madeira, Portugal, Jan. 2018, pp. 568–575 [63],
- II M. Valtonen Örnthag and A. Heyden. “Generalization of Parameter Recovery in Binocular Vision for a Planar Scene”. In: *International Journal of Pattern Recognition and Artificial Intelligence* 33.11 (2019) [62],
- III M. Valtonen Örnthag. “Fast Non-minimal Solvers for Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 40–51 [59],
- IV M. Valtonen Örnthag. “Efficient Radial Distortion Correction for Planar Motion”. In: *Pattern Recognition Applications and Methods. ICPRAM 2020. Lecture Notes in Computer Science*. Springer International Publishing, 2021, pp. 46–63 [72],
- V M. Valtonen Örnthag and M. Wadenbäck. “Enforcing the General Planar Motion Model: Bundle Adjustment for Planar Scenes”. In: *Pattern Recognition Applications and Methods. ICPRAM 2019. Lecture Notes in Computer Science*. Vol. 11996. Springer International Publishing, 2020, pp. 119–135 [70],
- VI M. Valtonen Örnthag, P. Persson, M. Wadenbäck, K. Åström, and A. Heyden. “Minimal Solvers for Indoor UAV Positioning”. In: *International Conference on Pattern Recognition (ICPR)* (2020), pp. 1136–1143 [69],

-
- VII M. Valtonen Örnthag, P. Persson, M. Wadenbäck, K. Åström, and A. Heyden. “Efficient Real-Time Radial Distortion Correction for UAVs”. In: *The IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2021, pp. 1751–1760 [68],
- VIII M. Valtonen Örnthag, C. Olsson, and A. Heyden. “Bilinear Parameterization for Differentiable Rank-Regularization”. In: *In the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2020 [66],
- IX M. Valtonen Örnthag, C. Olsson, and A. Heyden. “Differentiable Fixed-Rank Regularisation using Bilinear Parameterisation”. In: *In the British Machine Vision Conference (BMVC)*. 2019 [67],
- X M. Valtonen Örnthag and C. Olsson. “A Unified Optimization Framework for Low-Rank Inducing Penalties”. In: *In the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020, pp. 8474–8493 [65],
- XI J. P. Iglesias, C. Olsson, and M. Valtonen Örnthag. “Accurate Optimization of Weighted Nuclear Norm for Non-Rigid Structure from Motion”. In: *European Conference on Computer Vision (ECCV)*. Aug. 2020, pp. 21–37 [32],
- XII M. Valtonen Örnthag, J. P. Iglesias, and C. Olsson. “Bilinear Parameterization for Non-Separable Singular Value Penalties”. Submitted. To appear in the *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021 [64],

Subsidiary papers:

Papers II, IV and V are extended versions of the following conference papers (in the order given), which are left out due to overlapping material.

- M. Valtonen Örnthag and A. Heyden. “Generalisation of Parameter Recovery in Binocular Vision for a Planar Scene”. In: *International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI)*. Montréal, Canada, May 2018, pp. 37–42 [61],
- M. Valtonen Örnthag. “Radially Distorted Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Valletta, Malta, Feb. 2020, pp. 568–575 [60],
- M. Valtonen Örnthag and M. Wadenbäck. “Planar Motion Bundle Adjustment”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 24–31 [71],

Chapter 2

Modelling Cameras

2.1 The Pinhole Camera Model

The *pinhole camera model* can be thought of as a light ray passing through a small hole in a box. The light rays meet in the pinhole, or as it is otherwise known, the *camera centre*. The light ray orthogonal to the image plane $z = f$ and passing through the camera centre will intersect the image plane in the *principal point*, which we denote (u_0, v_0) . Now, consider a light ray passing through a scene point $(x, y, z) \in \mathbb{R}^3$. Where will it intersect the image plane? In Figure 2.1 we note two triangles, sharing parts of the projection line as their hypotenuses; hence, they are similar. Therefore, the sought intersection point in 3D space is given by $(fx/z + u_0, fy/z + v_0, f)$, which is naturally mapped to image coordinates by $(u, v) = (fx/z + u_0, fy/z + v_0)$.

In computer vision, it is convenient to work with *homogeneous coordinates*, since projective transformations can be represented by a single matrix multiplication. Furthermore, points at infinity can be represented using finite coordinates. Given a non-zero vector $\mathbf{x} \in \mathbb{R}^{n+1}$, the set of all vectors $\mathcal{R}_{\mathbf{x}} = \{\lambda\mathbf{x} \mid \lambda \in \mathbb{R} \setminus \{0\}\}$ is called a *ray*. The n -dimensional real projective space \mathbb{P}^n is a set of equivalence classes of vectors, identified with the rays $\mathcal{R}_{\mathbf{x}}$, for all non-zero vectors $\mathbf{x} \in \mathbb{R}^{n+1}$. In this setting, equivalent representations \mathbf{x} and \mathbf{y} of the same *point* must be in the same ray, *i.e.* $\exists \lambda \in \mathbb{R} \setminus \{0\}$ such that $\mathbf{x} = \lambda\mathbf{y}$, in which case we will write $\mathbf{x} \sim \mathbf{y}$. The theory of projective spaces goes far beyond the need of regular computer vision applications, and we shall settle with this brief introduction, and recommend the work of Cox *et al.* [12] for further details.

More generally, the complete pinhole camera model can in homogeneous coordinates

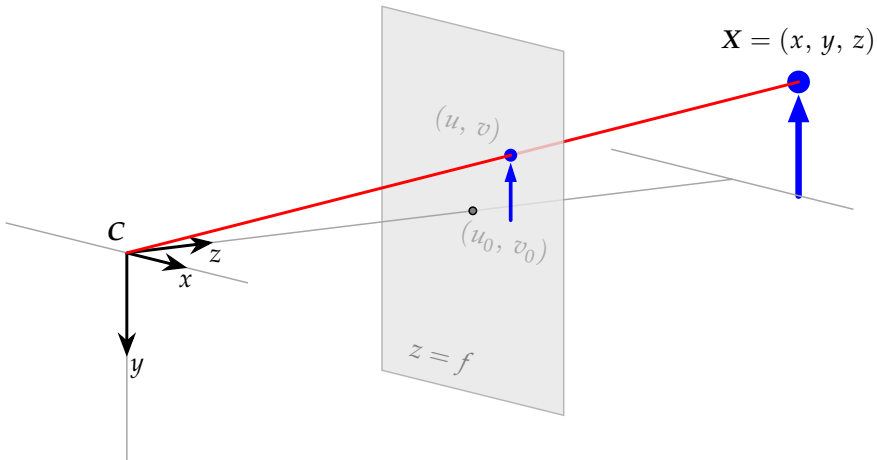


Figure 2.1: The pinhole camera model. A light ray passes through the camera centre C and a scene point \mathbf{X} . The corresponding image point is located at the intersection with the focal plane $z = f$. The principal point (u_0, v_0) determines the location in the local coordinate system used to represent the image.

be written as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim \begin{bmatrix} f & \gamma & u_0 & 0 \\ 0 & \alpha f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f & \gamma & u_0 \\ 0 & \alpha f & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{:=\mathbf{K}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \quad (2.1)$$

where f is the focal length and α is the aspect ratio. Normally $\alpha \approx 1$, but may differ due to a variety of optical and digital techniques, such as using anamorphic formats, flaws in sensor manufacturing or digital post-processing. The skew parameter γ corrects for non-rectangular pixels. Together these parameters are known as the *intrinsic parameters*, and form the *calibration matrix* \mathbf{K} . The intrinsic parameters affect the camera itself and act in synergy with the *extrinsic parameters* which affect the relative pose of the camera with respect to other cameras (or objects) in a global coordinate system.

Working with homogeneous coordinates, let $\mathbf{X} \in \mathbb{P}^3$ denote a scene point and $\mathbf{x} \in \mathbb{P}^2$ the corresponding image point, then $\mathbf{x} \sim \mathbf{P}\mathbf{X}$, where \mathbf{P} is the camera matrix. In general, there are six degrees of freedom for the extrinsic parameters—three translational components, determining the position of the camera in 3D space, and three rotational components determining the orientation. The translation is normally modelled by a translation vector \mathbf{t} and the rotation by a matrix \mathbf{R} . Consequently, the

camera matrix can then be decomposed into two parts $\mathbf{P} = \mathbf{KR}[\mathbf{I} \mid -\mathbf{t}]$, where we recognise the calibration matrix \mathbf{K} , and $\mathbf{R}[\mathbf{I} \mid -\mathbf{t}]$ constituting the extrinsic parameters.

The pinhole camera is an idealised camera model, that does not model more complex phenomena arising from imperfections in lenses and sensors, *e.g.* geometric distortion (which is discussed in Section 2.2). It is, however, a very useful model, as many artefacts can be compensated for; hence, it is of great theoretical importance.

In the general case, a projective camera \mathbf{P} is of rank three and has eleven degrees of freedom (as there is a scale ambiguity). It maps scene points \mathbf{X} to image points $\mathbf{x} \sim \mathbf{PX}$. If we let $\mathbf{P} = [\mathbf{A} \mid \mathbf{t}]$, it can be shown that \mathbf{P} is a finite camera if and only if \mathbf{A} is non-singular. Here a finite camera means that the camera centre is not located in a point at infinity. For a more detailed discussion, the reader is referred to the standard reference in computer vision by Hartley and Zisserman [27].

2.2 Physical Cameras

Physical cameras never obey the pinhole camera model exactly. In practice, the aperture needs to be sufficiently large to allow enough light to reach the sensor, thus deviates from the infinitesimal opening of the theoretical pinhole camera model. However, simply enlarging the aperture would cause the image to blur, as an image point would have multiple rays intersecting it, each corresponding to different 3D points. This is mitigated by using lenses, which efficiently focus points at a specific distance to the focal plane. Note that this is a trade-off, as rays emitted at other points do not converge to a single point on the focal plane.

When working with a complete image system with sensors and optics, there are many unwanted phenomena that arise. Limiting the discussion to the optics, we have issues with (monochromatic) aberrations: defocus, spherical aberration, coma, astigmatism, and field curvature, all of which cause the image to appear blurry. Professional lens manufacturers must make trade-offs in design to suppress these artefacts while making sure the optical elements are able to be mass-produced and sold at a reasonable price. Even if these issues are reduced, the resulting image may still be distorted compared to the pinhole camera model, *e.g.* physically straight lines may appear to be curved in the image.

Furthermore, there are two types of chromatic aberrations—longitudinal and lateral—which are caused by dispersion; that is, different wavelengths of natural light are reflected differently. The blue channel is thus perceived as being augmented, causing a blue (or purple) fringe around the object.



Figure 2.2: Calibration setup for one of the sequences in the famous KITTI dataset, where OpenCV’s `findChessboardCorners` has been used to find the leftmost chequerboard. The radial distortion is noticeable on the sides of the image, where physically straight lines appear to be curved. This phenomenon is not compatible with the pinhole camera model, and thus must be compensated for prior to estimating the relative pose between images. Image credit: KITTI dataset [23].

2.3 Correcting Lens Distortion

Some of the above-mentioned aberrations can be corrected for in software. For computer vision applications, it has been observed that lens distortions can affect the overall performance quite severely, and thus much research goes into correcting for it. The main idea is to take a distorted image and rectify it, such that the rectified image is approximated well by the pinhole camera model. The most common way to approach this is to assume a functional form to model the distortion profile, *i.e.* given distorted image points \mathbf{x}_d the corresponding undistorted points \mathbf{x}_u are obtained using a function g , such that $\mathbf{x}_u = g(\mathbf{x}_d)$. Typically, these functions are assumed to be of a certain type, *e.g.* polynomials with unknown coefficients, thus the problem of determining the distortion profile has been reduced to finding these unknowns.

For radial distortion, the distortion at a point $\mathbf{x} = (u, v)$ is assumed to depend on the Euclidean distance $r = \|\mathbf{x} - \mathbf{x}_c\|$ to the distortion centre \mathbf{x}_c . Although the distortion centre and principal point are not necessarily the same, this is a common assumption. The oldest approach, stemming from the Brown–Conrady model [6], assumes a polynomial model where $\mathbf{x}_u = h(r)(\mathbf{x}_d - \mathbf{x}_c)$, with

$$h(r) = 1 + \sum_{i=1}^n c_i r^i. \quad (2.2)$$

Originally, only even polynomials were used. Furthermore, tangential distortion is also

taken into consideration, *i.e.* when the optics and focal plane are not parallel.

So, how does one calibrate a physical camera in practice? A popular method, that estimates the calibration matrix and the distortion coefficients, is *Zhang's method* [80], in which images of a planar surface (a chequerboard) are captured from different views (either the target or the camera moves) and then a series of homographies (see Chapter 3) are computed, from which the relative poses are estimated. An initial solution for the distortion parameters is obtained by a linear least-squares approximation, and, finally, all parameters are refined by nonlinear optimisation (see Chapter 7). Although the underlying theory is non-trivial, it is easy for non-mathematicians to practically apply this method, as it is available in frameworks such as OpenCV [5], which only require the input images in order to produce the wanted output, see Figure 2.2.

Polynomial models are not the only ones available. Another popular method is the division model [21], where the function is given by

$$h(r) = \frac{1}{1 + \sum_{i=1}^n c_i r^i}. \quad (2.3)$$

The major benefit of using the division model is that fewer parameters are necessary in order to model the distortion profile [52], though this may not be true when allowing higher order polynomials [57]. Even with a single distortion parameter qualitative results can be obtained, which opens up the possibility to use them in minimal solvers, which we will discuss in Chapter 6.

Of course, by now we can think of more exotic models, such as combining the above two models to yield a rational model, as in [34]. What is perhaps interesting to note is a completely different approach, namely generic camera models, where gridded data is used. Some recent papers, such as [54], suggest that these models will become prevalent in the future, as parametric models tend to incur bias to the reconstruction.

2.4 Finding Point Correspondences

In this section, we will briefly discuss how to automatically find point correspondences in images, which is needed in many computer vision applications. Early work, such as *Canny edge detection* or the *Sobel operator*, typically involves computing gradients of the image and using a Gaussian filter to reduce the impact of noise. Another well-known detector is the *Harris detector*, which builds on the fact that the image intensity will vary in more than one direction at a corner point—making it suitable as a feature point, as it is translationally invariant. Due to the Gaussian filter causing the edge to smudge out, it may be wise to use *non-maximum suppression* to get a thinner, and more accurate, edge. This, however, comes at the cost of increased computational power.

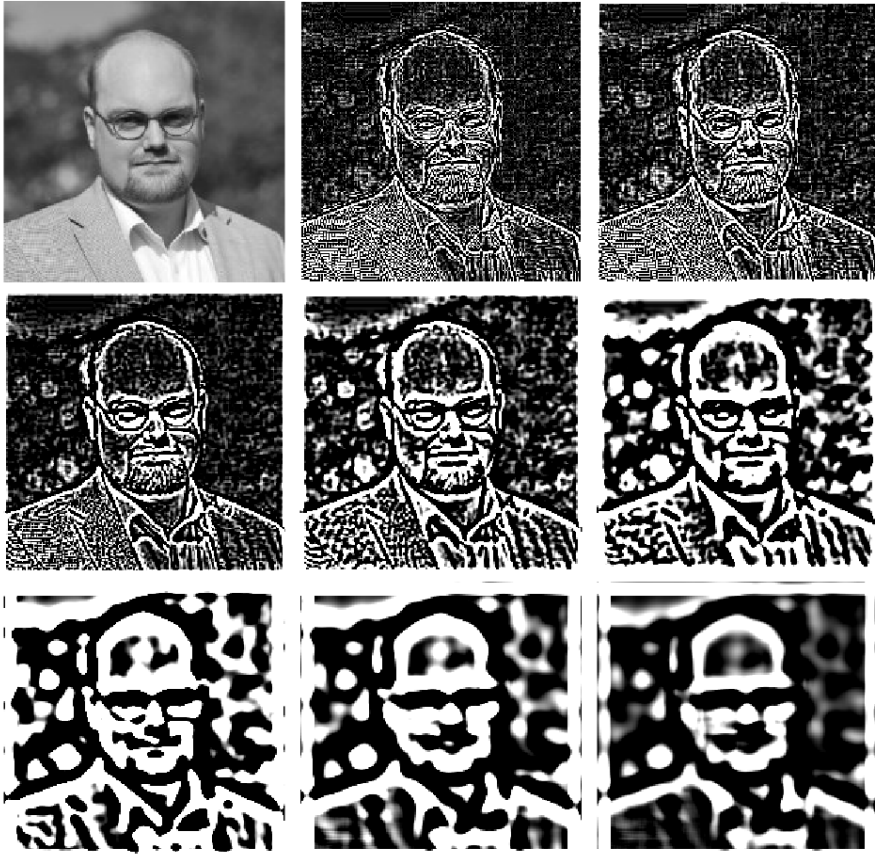


Figure 2.3: Picture of the author and difference of Gaussians (DoG) applied to the image. Note that the first images highlight features on small scale (top row) and, as the standard deviation grows, coarser and coarser features are found (left to right, row by row).

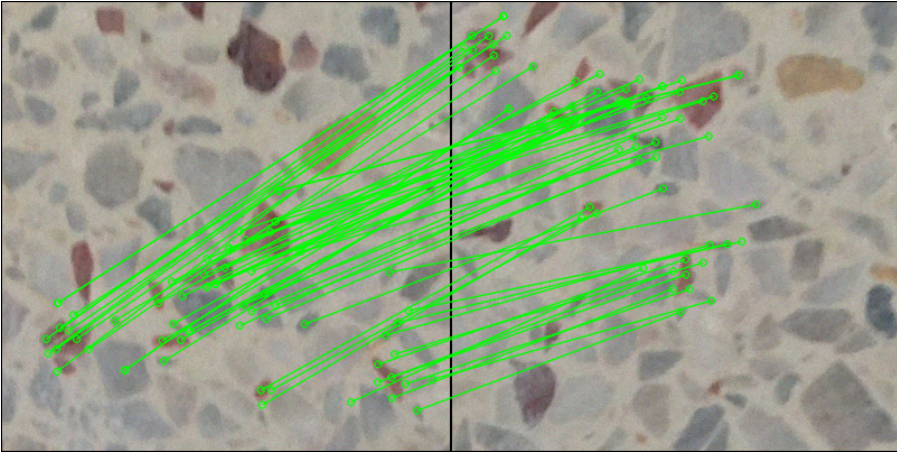


Figure 2.4: SIFT keypoints of an industrial floor matched by a Nearest Neighbour matcher.

The procedure of applying Gaussians is popular, even in the absence of gradients, and is formalised in what is known as *scale-space theory*. In fact, it can be shown that the Gaussian kernel is the only choice fulfilling certain desirable properties, such as linearity, shift-invariance and semigroup property [38]. Intuitively, by increasing the standard deviation of the Gaussian a more blurred version of the image is obtained. By taking the difference of blurred images, features of varying scale may be extracted. This is known as *difference of Gaussians* (DoG) and is illustrated in Figure 2.3. Keypoints that are extrema of the DoG over several scales are good candidates for *scale-invariant features*, and is the key idea behind the popular feature descriptor SIFT [39]. Among more recent descriptors are SURF [3], FAST [49], BRIEF [8] and the combination of the latter two, known as ORB [50], which is used in the popular SLAM system ORB-SLAM2 [45]. When choosing a feature descriptor it is often a trade-off between speed and accuracy [10], and as long as one is able to discard incorrect matches it is not clear which is preferable. In Section 6.1 we will introduce a method capable of separating inliers from outliers.

Generating feature points is only a part of the pipeline—the next step is to match features, and there are a variety of algorithms to facilitate this process. A classic method is the Nearest Neighbour algorithm (NN) which takes one feature in the first image and compares it to all other features in the second image, then selects the closest descriptor with respect to some metric. Such an algorithm is also known as a brute-force matcher, as it matches all possibilities, and is computationally intensive if the number of features is large. To speed up the matching, one may use approximate algorithms, such as Fast Library for Approximate Nearest Neighbours (FLANN) [44] which limits the number

of matches to be compared.

In Figure 2.4 an example of SIFT keypoints (green circles) and their matches (lines connecting the circles) obtained by the Nearest Neighbour matcher is shown. The image depicts what could be a possible input to the algorithms developed in the first part of this thesis—an industrial floor. Note that it appears that most of the points are transformed in a consistent way from the left view to the right, indicating that they were correctly matched. In the next chapter, we will see how these matches can be used in practice.

Chapter 3

Homographies

In this chapter, we will discuss planar motion, as well as introduce the invertible mapping between projective spaces, known as a *homography* or *projective transformation*. For the purpose of this thesis, the projective spaces are typically \mathbb{P}^2 and the corresponding homography $\mathbf{H} : \mathbb{P}^2 \rightarrow \mathbb{P}^2$, *i.e.* for two corresponding image points $\mathbf{x} \leftrightarrow \mathbf{x}'$, a homography \mathbf{H} , if it exists, must fulfil $\mathbf{x}' \sim \mathbf{H}\mathbf{x}$. This is a key concept throughout the first two parts of the thesis (Papers I–VII), and will be the input or output to many of the proposed algorithms.

3.1 Properties of Homographies

We will use the *Sherman–Morrison formula* [55] to prove that points in a plane are related by a homography.

Result 1 (Sherman–Morrison). *Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be non-singular, and $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$. Then $\mathbf{A} + \mathbf{u}\mathbf{v}^T$ is invertible if and only if $1 + \mathbf{v}^T \mathbf{A}^{-1} \mathbf{u} \neq 0$.*

In addition, there is a closed-form expression for the inverse, but it is omitted here as we do not use it explicitly. Instead, we focus on the following important result.

Result 2. *If all scene points lie in a plane not intersecting the camera centres, there exists a homography \mathbf{H} between the corresponding image points.*

Proof. The statement holds true for general projective cameras, but we show it for our special case $\mathbf{P} = [\mathbf{I} \mid \mathbf{0}]$ and $\mathbf{P}' = \mathbf{R}[\mathbf{I} \mid -\mathbf{t}]$. Let the unknown plane be given by $\boldsymbol{\pi} = (\mathbf{n}, 1)$. The parameterisation of $\boldsymbol{\pi}$ is justified by the plane not intersecting any of the camera centres. Given a scene point \mathbf{X} on the plane $\boldsymbol{\pi}$, it is imaged in the first camera as $\mathbf{x} \sim \mathbf{P}\mathbf{X}$, hence $\mathbf{X} \sim (\mathbf{x}, \lambda)$, where λ is determined by the plane constraint

$\boldsymbol{\pi}^T \mathbf{X} = 0$, giving $\lambda = -\mathbf{n}^T \mathbf{x}$. Similarly, for the second camera

$$\mathbf{x}' \sim \mathbf{P}' \mathbf{X} = \mathbf{R} (\mathbf{I} + \mathbf{t} \mathbf{n}^T) \mathbf{x}, \quad (3.1)$$

where we define $\mathbf{H} = \mathbf{R} (\mathbf{I} + \mathbf{t} \mathbf{n}^T)$. The matrix \mathbf{R} is a rotation matrix, hence invertible. By Result 1 the matrix $\mathbf{I} + \mathbf{t} \mathbf{n}^T$ is invertible, as $\boldsymbol{\pi}$ does not intersect the second camera centre \mathbf{t} . Lastly, as the product of two invertible matrices is invertible, so is \mathbf{H} , proving that it is in fact a homography. \square

In the remaining parts of this thesis, we will use the notation $[\mathbf{v}]_{\times}$ to denote the linear operation corresponding to the cross product of \mathbf{v} and another vector,

$$[\mathbf{v}]_{\times} \mathbf{x} = \mathbf{v} \times \mathbf{x}, \quad \forall \mathbf{x} \in \mathbb{R}^3. \quad (3.2)$$

Furthermore, we note that $[\mathbf{v}]_{\times}$ is skew-symmetric, hence $\det([\mathbf{v}]_{\times}) = 0, \forall \mathbf{v} \in \mathbb{R}^3$.

3.2 Direct Linear Transformation

Consider N scene points in a common plane, and project them in two views, forming the image correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$, for $i = 1, \dots, N$. We now know that $\mathbf{x}'_i \sim \mathbf{H} \mathbf{x}_i$ for some homography \mathbf{H} , or, equivalently,

$$\mathbf{x}'_i \times \mathbf{H} \mathbf{x}_i = (\mathbf{x}'_i{}^T \otimes [\mathbf{x}'_i]_{\times}) \text{vec}(\mathbf{H}) = \mathbf{0}, \quad (3.3)$$

where $\text{vec}(\mathbf{X})$ denotes the (column-wise) vectorisation of the matrix \mathbf{X} and \otimes denotes the Kronecker product. The matrix $\mathbf{x}'_i{}^T \otimes [\mathbf{x}'_i]_{\times}$ has rank two, and therefore it is only necessary to include two linearly independent rows; we shall denote this matrix by ζ_i . Since the matrix \mathbf{H} has eight degrees of freedom¹, we need at least four point correspondences to estimate \mathbf{H} . With $N \geq 4$ point correspondences, we are left with the following problem

$$\underbrace{\begin{bmatrix} \zeta_1 \\ \vdots \\ \zeta_N \end{bmatrix}}_{:= \mathbf{Z}} \text{vec}(\mathbf{H}) = \mathbf{0}, \quad (3.4)$$

or, in other words, to find the one-dimensional null space of \mathbf{Z} , which can be found using *singular value decomposition* (SVD). This assumes that the matches are correct and

¹The matrix \mathbf{H} is a 3×3 matrix with nine elements; however, the degrees of freedom are reduced to eight due to the scale ambiguity.

noise-free which is rarely the case in real-life applications. With real data such solutions typically do not exist if $N > 4$, but the SVD can still be used to find the desired null space (or an approximation thereof). This algorithm is known as *direct linear transformation* (DLT). In Chapter 6, we will discuss problems arising due to noisy data and how to determine which point correspondences that are correctly matched.

When the homography is found, it can be used to rectify the first image, which is shown in Figure 3.1. In order to create this image, the matched SIFT keypoints from Figure 2.4 are used together with the DLT equations in a RANSAC framework (see Section 6.1). Creating panoramas from homographies, however, is only one of many applications. Since the homography contains information about how the point correspondences relate to each other, one may extract the relative camera pose between subsequent images. By doing so, a method to track the camera trajectory can be devised, and, therefore, the motion of the mobile platform on which it is mounted.

As discussed in Section 2.1, assuming known calibration, the relative pose consists of three rotational components and three translational components. Therefore, it is possible to represent it using five degrees of freedom. By constraining the motion further, the degrees of freedom will be reduced, and as a consequence, one would not need as many as four point correspondences to fit such a model. In Section 5.3.1 an alternative method for computing the inter-image homography is proposed, where a specific problem geometry, namely the general planar motion model, is enforced.

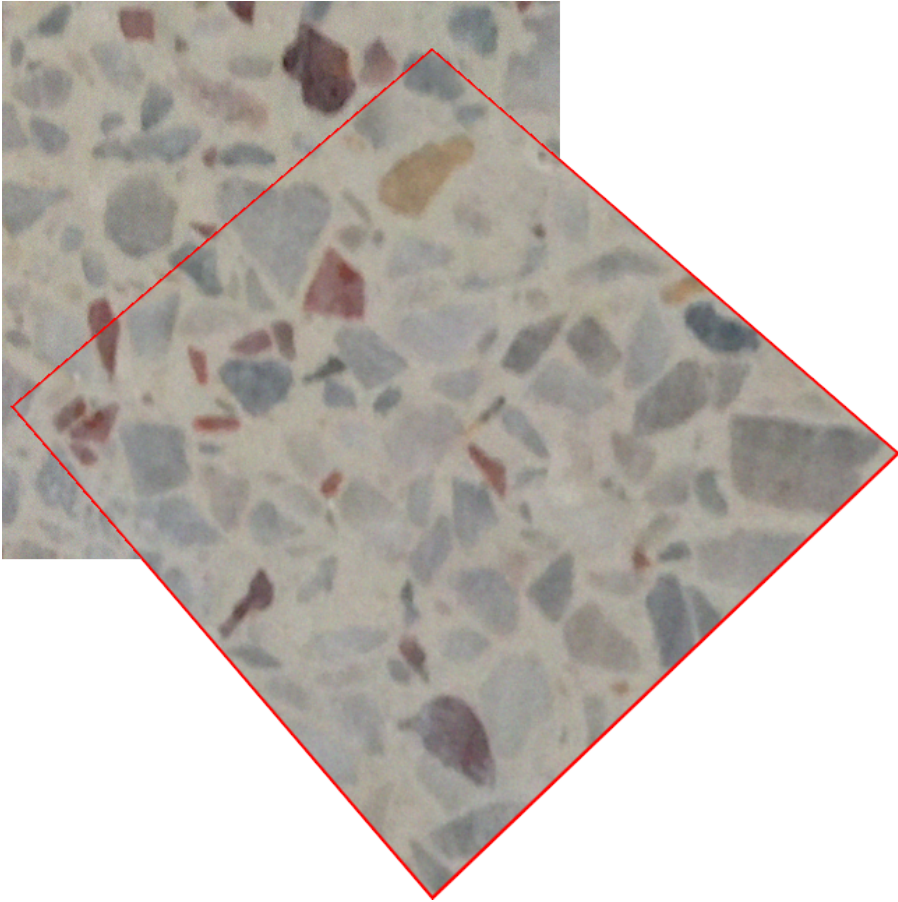


Figure 3.1: Stitched image obtained by computing the inter-image homography from the SIFT keypoints found in Figure 2.4 using direct linear transform (DLT) in a RANSAC framework (Section 6.1). The red frame marks the border of the second image.

Chapter 4

Epipolar Geometry

Consider a scene point X imaged in x and x' by two cameras C and C' , respectively. The back-projection of x is the ray joining the first camera centre and the scene point. It is imaged as a line in the second camera, which is known as the *epipolar line* for x , upon which x' must lie, as is illustrated in Figure 4.1. The geometry is completely scene-independent and captures the projective relation between two cameras defined by their intrinsic and extrinsic parameters. This is known as *epipolar geometry*.

In fact, any image point x corresponds to an epipolar line ℓ' in the other view, given the camera centre C . The ray joining the camera centres is known as the *baseline* and the intersections with the image planes are known as the *epipoles*. The map $x \mapsto \ell'$ is represented by the *fundamental matrix* F , i.e. $\ell' = Fx$, which we will derive in the next section; however, we will need some basic results from epipolar geometry first, in order to do so.

4.1 The Fundamental Matrix

The following result is important regarding how well a scene can be reconstructed from uncalibrated cameras.

Result 3. *If the calibration is unknown and possibly varying, the reconstruction can only be uniquely determined up to a projective transformation.*

Proof. Assume $x \sim PX$, and let $M : \mathbb{P}^3 \rightarrow \mathbb{P}^3$ be a projective transformation. Then

$$x \sim PX = PMM^{-1}X = P'X', \quad (4.1)$$

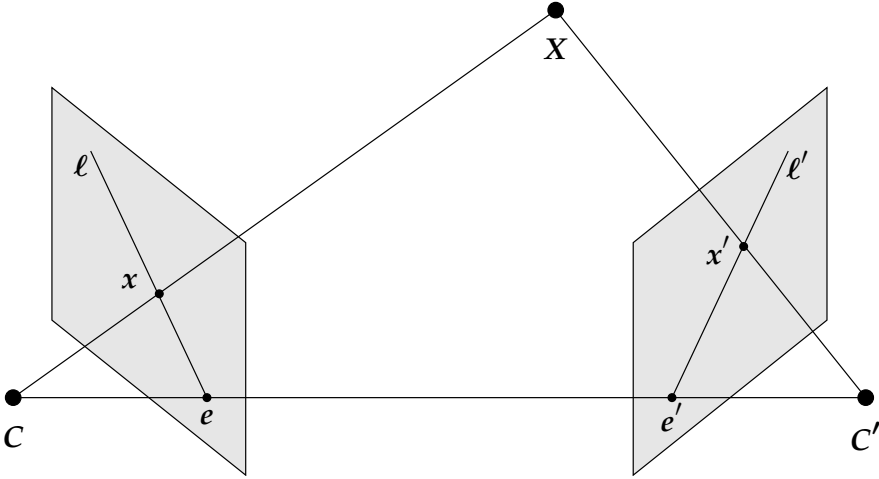


Figure 4.1: Epipolar geometry. A scene point X is imaged in two cameras centred at C and C' ; the corresponding image points are given by x and x' , respectively. The baseline is the ray joining the camera centres, and the intersections of the image planes with the baseline are the epipoles e and e' . The epipolar lines ℓ and ℓ' are the intersections of the corresponding image planes and the plane spanned by the camera centres and the scene point.

where $P' = PM$ and $X' = M^{-1}X$. The existence of M^{-1} is guaranteed since M is a projective transformation. Consequently, a new solution can be constructed by applying a projective transformation to an existing one. \square

When considering the relative pose between cameras it is convenient to define the *canonical form* for a camera pair, which is possible due to Result 3.

Result 4. *A pair of cameras can always be expressed as $P = [I \mid \mathbf{0}]$ and $P' = [A \mid \mathbf{t}]$.*

Proof. Let P_1 and P_2 be a pair of general projective cameras, and let C denote the camera centre of the first camera expressed in homogeneous coordinates. Then $M = [P_1^+ \ C]$, where P_1^+ is the Moore–Penrose pseudo-inverse of P_1 , defines a projective transformation. To see this, assume M is singular, and let p_i denote the columns of P_1^+ . Then there exists λ_i , not all zero, such that $C = \sum_i \lambda_i p_i$. Since C is the camera centre $P_1 C = \mathbf{0}$, implying that $\lambda_1 = \lambda_2 = \lambda_3 = 0$, which is a contradiction. It is easy to verify that $P = P_1 M = [I \mid \mathbf{0}]$ and, together with $P' = P_2 M$, forms an equivalent representation of the original camera pair, according to Result 3. \square

Result 4 holds for all projective cameras; however, for practical purposes it is often convenient to only consider a subset of this class—namely, those where \mathbf{A} is a rotation matrix, or at least invertible. In the remaining part of this thesis, we will assume that this is the case, and explicitly write the camera matrix as $\mathbf{KR}[\mathbf{I} \mid -\mathbf{t}]$, where \mathbf{R} is a rotation matrix. When the calibration matrix \mathbf{K} is unknown we will clearly separate the intrinsic and extrinsic parameters, *i.e.* we may assume that two uncalibrated cameras can be written as $\mathbf{P} \sim \mathbf{K}[\mathbf{I} \mid \mathbf{0}]$ and $\mathbf{P}' \sim \mathbf{K}'\mathbf{R}[\mathbf{I} \mid -\mathbf{t}]$, respectively, and refer to this as the canonical representation. Again, note that this is a restriction of the general case and not all projective cameras are captured by this parameterisation. A benefit of this parameterisation, however, is that the camera centres are located in the origin and \mathbf{t} , respectively, which is convenient for our applications. This follows from the camera centres \mathbf{C} and \mathbf{C}' fulfilling the relations

$$\mathbf{P} \begin{bmatrix} \mathbf{C} \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{C} = \mathbf{0}, \quad \mathbf{P}' \begin{bmatrix} \mathbf{C}' \\ 1 \end{bmatrix} = \mathbf{K}'\mathbf{R}(\mathbf{C}' - \mathbf{t}) = \mathbf{0}, \quad (4.2)$$

giving $\mathbf{C} = \mathbf{0}$ and $\mathbf{C}' = \mathbf{t}$, for the canonical representation of a camera pair. The epipoles are thus given by

$$\mathbf{e} \sim \mathbf{P} \begin{bmatrix} \mathbf{C}' \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{t}, \quad \mathbf{e}' \sim \mathbf{P}' \begin{bmatrix} \mathbf{C} \\ 1 \end{bmatrix} = -\mathbf{K}'\mathbf{R}\mathbf{t}. \quad (4.3)$$

We are now ready to state the *epipolar constraint* for the fundamental matrix and a pair of corresponding points.

Result 5. *For any pair of corresponding points $\mathbf{x} \leftrightarrow \mathbf{x}'$ the fundamental matrix \mathbf{F} satisfies the condition*

$$\mathbf{x}'^T \mathbf{F} \mathbf{x} = 0. \quad (4.4)$$

Furthermore, for a pair of camera matrices in canonical form, the fundamental matrix is given by $\mathbf{F} = [\mathbf{e}']_{\times} \mathbf{K}'\mathbf{R}\mathbf{K}^{-1}$.

Proof. Let \mathbf{P} and \mathbf{P}' be a canonical representation of two camera matrices, and let \mathbf{X} denote an arbitrary scene point. If \mathbf{x} is the image point corresponding to \mathbf{X} in the first camera, then $\mathbf{x} \sim \mathbf{P}\mathbf{X}$ and the corresponding viewing ray is given by $\mathbf{X}(\lambda) = (\lambda\mathbf{K}^{-1}\mathbf{x}, 1)$. The projection of the viewing ray in the second camera is the corresponding epipolar line $\ell' = \mathbf{P}'\mathbf{X}(\lambda) = \mathbf{K}'\mathbf{R}(\lambda\mathbf{K}^{-1}\mathbf{x} - \mathbf{t})$. Since a line in \mathbb{P}^2 can be expressed as $\ell'^T \mathbf{x} = 0$, and the line is defined by two points, we may pick $\lambda = 0$ and $\lambda = 1$ from the parameterisation of ℓ' which yields $-\mathbf{K}'\mathbf{R}\mathbf{t}$ and $\mathbf{K}'\mathbf{R}(\mathbf{K}^{-1}\mathbf{x} - \mathbf{t})$, respectively. We note that $\mathbf{e}' \sim -\mathbf{K}'\mathbf{R}\mathbf{t}$, hence

$$\ell' = \mathbf{e}' \times \mathbf{K}'\mathbf{R}(\mathbf{K}^{-1}\mathbf{x} - \mathbf{t}) = \mathbf{e}' \times \mathbf{K}'\mathbf{R}\mathbf{K}^{-1}\mathbf{x} = [\mathbf{e}']_{\times} \mathbf{K}'\mathbf{R}\mathbf{K}^{-1}\mathbf{x}, \quad (4.5)$$

where $F = [e']_{\times} K' R K^{-1}$ is the fundamental matrix. Lastly, if x' corresponds to x then

$$x'^T \ell' = x'^T F x = 0. \quad (4.6)$$

□

Note that the converse statement is not generally true—two points x and x' , fulfilling the epipolar constraint for some fundamental matrix, need not stem from the same scene point. They do, however, need to be in the same plane spanned by the camera centres and the respective scene points, *i.e.* they must be mapped to the respective epipolar lines, as can be seen in Figure 4.1.

In Chapter 3 we concluded that the scale ambiguity reduced the number of degrees of freedom to eight when considering homographies. The very same argument applies to the fundamental matrix. But is it a homography? The answer is no, as we shall see in the following result.

Result 6. *Any fundamental matrix F satisfies $\det F = 0$.*

Proof. Given a pair of camera matrices in canonical form, the fundamental matrix is given by $F = [e']_{\times} K' R K^{-1}$, hence $\det F = 0$, due to the determinant being a multiplicative map and since the determinant of a 3×3 skew-symmetric matrix is zero. □

Consequently, the fundamental matrix is singular, hence not a homography. We will discuss the differences between the two in the next section and show how this constraint can be used to estimate the fundamental matrix more accurately.

4.2 Why Homographies?

As previously mentioned, we will exclusively use homographies in the first part of the thesis (Papers I–V), and the following result is a strong argument for this approach. Assuming a set of point correspondences is noise-free and no outliers are present, we say that it is geometrically degenerate (w.r.t. the fundamental matrix) if a unique epipolar transformation does not exist. Specifically, if all point correspondences lie in a common plane, the fundamental matrix does not have a unique representation, which we formulate in the following result. We also refer to scene points in a common plane as being *coplanar*. The proof is inspired by the compact version given by Wadenbäck [73]; however, the result is well-known since earlier, see *e.g.* Torr *et al.* [58].

Result 7. *The estimation of the fundamental matrix degenerates when all scene points lie in a common plane.*

Proof. Assume there are N scene points. Since they all lie in a common plane, we may assume the point correspondences are related by a homography, *i.e.* $\mathbf{x}'_i \sim \mathbf{H}\mathbf{x}_i$, for all $i = 1, \dots, N$. It follows that

$$\mathbf{x}'_i{}^T \mathbf{F} \mathbf{x}_i = 0 \Leftrightarrow \mathbf{x}'_i{}^T \mathbf{H}^T \mathbf{F} \mathbf{x}_i = 0 \Leftrightarrow (\mathbf{x}_i \otimes \mathbf{x}_i)^T \text{vec}(\mathbf{H}^T \mathbf{F}) = 0. \quad (4.7)$$

The vector $\mathbf{x}_i \otimes \mathbf{x}_i$ contains only six unique elements, as it consists of the product of all elements of \mathbf{x}_i with itself, hence

$$\text{rank} \begin{bmatrix} (\mathbf{x}_1 \otimes \mathbf{x}_1)^T \\ \vdots \\ (\mathbf{x}_N \otimes \mathbf{x}_N)^T \end{bmatrix} \leq 6. \quad (4.8)$$

By the rank-nullity theorem, there is a three-dimensional family of fundamental matrices compatible with all of the epipolar constraints. \square

4.3 Calibrated Cameras and the Essential Matrix

Let $\bar{\mathbf{X}} \in \mathbb{R}^3$ denote the inhomogeneous representation of $\mathbf{X} \in \mathbb{P}^3$. Assuming known calibration, one may apply the inverse transformation of the calibration matrix to an image point, *i.e.* given $\mathbf{x} \sim \mathbf{P}\mathbf{X} = \mathbf{K}\mathbf{R}(\bar{\mathbf{X}} - \mathbf{t})$, we obtain the *normalised image point* $\tilde{\mathbf{x}} = \mathbf{K}^{-1}\mathbf{x} \sim \mathbf{R}(\bar{\mathbf{X}} - \mathbf{t})$. Furthermore, this can be thought of as applying the inverse transformation to the camera matrix itself, thus producing a *normalised* or *calibrated camera* $\tilde{\mathbf{P}} = \mathbf{K}^{-1}\mathbf{P}$, or, equivalently, that the calibration matrix \mathbf{K} is the identity. When using normalised image points, the corresponding fundamental matrix is known as the *essential matrix*, often denoted \mathbf{E} . The epipolar constraint for a pair of calibrated cameras is thus given by

$$\tilde{\mathbf{x}}'^T \mathbf{E} \tilde{\mathbf{x}} = 0, \quad (4.9)$$

where $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}}'$ denote the normalised image points for the corresponding points $\mathbf{x} \leftrightarrow \mathbf{x}'$. From (4.9), the relation between the fundamental matrix and the corresponding essential matrix can be established; namely, by substituting $\tilde{\mathbf{x}} = \mathbf{K}^{-1}\mathbf{x}$ and $\tilde{\mathbf{x}}' = \mathbf{K}'^{-1}\mathbf{x}'$, one obtains $\mathbf{x}'^T \mathbf{K}'^{-T} \mathbf{E} \mathbf{K}^{-1} \mathbf{x} = 0$, from which it follows that $\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}$. Since the number of unknowns for the essential matrix is only dependent on the relative pose between the cameras there are five degrees of freedom—three rotational components and three translational components; however, as in the case of the fundamental matrix,

the scale ambiguity is present. We leave the discussion on calibrated cameras for now, and again refer to Hartley and Zisserman [27] for further analysis. We will, however, note the following constraint that is necessary for an essential matrix.

Result 8 (Demazure [15]). *Any essential matrix E satisfies*

$$2EE^T E - \text{tr}(EE^T)E = \mathbf{0} . \quad (4.10)$$

This result is commonly referred to as the *trace constraint* and together with $\det E = 0$ (a simple modification of Result 6) it is possible to determine the essential matrix from five point correspondences [36, 46]. We shall discuss such methods in more detail in Chapter 6. Furthermore, it can be shown that an equivalent statement to Result 8 is that either $E \equiv 0$, or E has two positive, identical singular values and a third singular value equal to zero.

Since the intrinsic calibration only depends on the camera itself, it is often favourable to calibrate it in order to reduce the number of unknown parameters to recover. Furthermore, physical cameras suffer from tangential and radial distortions to some degree, as discussed in Section 2.2, which are not modelled by the pinhole camera model, nor are wanted in the image¹. In the first part of this thesis (Papers I–V), we will assume that the intrinsic parameters are known, which in practice requires that the cameras used in the experiments are calibrated.

4.4 The Eight-Point Algorithm

In this section we will describe a method for estimating the fundamental matrix, known as the *eight-point algorithm*, which is similar to the DLT method described in Section 3.2. Let $x_i \leftrightarrow x'_i$ denote point correspondences between two cameras and let F be the corresponding fundamental matrix. Each point pair must satisfy the epipolar constraint, and therefore $x_i'^T F x_i = 0$ for all $i = 1, \dots, N$. Then

$$x_i'^T F x_i = (x_i \otimes x'_i)^T \text{vec}(F) = 0 . \quad (4.11)$$

By considering all point correspondences, the following equation must be satisfied

$$\underbrace{\begin{bmatrix} (x_1 \otimes x'_1)^T \\ \vdots \\ (x_N \otimes x'_N)^T \end{bmatrix}}_{:=A} \text{vec}(F) = \mathbf{0} . \quad (4.12)$$

¹In certain cases these discrepancies are in fact wanted for aesthetic and artistic appearance; however, for the topic of this thesis they are considered as defects.

By computing the right null space of \mathbf{A} , the fundamental matrix can be estimated; in order to do so, one would typically use *singular value decomposition* (SVD) where the matrix \mathbf{A} decomposes into $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. Assuming the point correspondences are correctly matched, the column of \mathbf{V} corresponding to the zero (or vanishing) singular value is the null space of \mathbf{A} . This requires a minimum of eight point correspondences; hence the name. In practice, one would also normalise the image points for numerical stability², and enforce the constraint $\det \mathbf{F} = 0$. The latter can be accomplished by nullifying the smallest singular value of the fundamental matrix, thus finding the closest rank-2 matrix in the Frobenius norm. Another method would be to use but seven point correspondences and add the cubic constraint emanating from the rank-2 constraint; however, solving such equations requires different techniques, some of which we will discuss in more detail in Chapter 6.

²As we treat the uncalibrated case we do not have access to the intrinsic parameters, and therefore do not know the calibration matrix. When we refer to *normalisation* in this context, we mean *Hartley normalisation* [28], which scales the coordinates so that the average distance of an image point to the origin is $\sqrt{2}$.

Chapter 5

Simultaneous Localisation and Mapping

5.1 Motivation and History

Imagine yourself waking up in an unknown environment—where are you and how should you move in order to get to where you want to be? With the technological advances of today, similar questions arise for how to successfully manoeuvre autonomous vehicles. One of the most fundamental questions is: where is the vehicle located with respect to its surroundings? The question becomes harder to answer if the terrain, in which the vehicle travels, is unknown.

In the second part of the 1980s, many papers were published concerning the reconstruction of three-dimensional structures, including the uncertainty in measurements, which was formalised by Smith and Cheeseman [56] and Durrant–Whyte [17]. At that moment in time, the first algorithms dealing with the question of how to track a vehicle in an unknown environment emerged. Today they can be grouped together under what is known as Simultaneous Localisation and Mapping (SLAM). During its youth, robotics researchers favoured sensors such as lasers and odometers; however, cameras are a popular choice today. This is partly due to the many algorithms available from the computer vision community, but also the decreasing price of imaging sensors and increased computational capacity of commercial electronic devices. In the first two parts of this thesis (Papers I–VII) the input is typically assumed to be sequences of images, and the proposed algorithms will categorise as *Visual Odometry* (VO). One of the first VO algorithms is due to Harris and Pike [26] in 1988. They used Kalman filtering, a technique that has been proven successful when fusing data from multiple sensors and for dealing with uncertainties in measurements [18].

5.2 State-of-the-Art

During the three decades that have passed since the pioneering work of Harris and Pike, a multitude of methods have surfaced. Among the latest and more popular ones are ORB-SLAM2 [45], which supports monocular, stereo and RGB-D cameras.

In order to successfully deploy SLAM algorithms, it is often necessary to tailor them after what is known about the vehicle and the expected surroundings, *e.g.* is it an autonomous bus meant to travel in urban areas, or a drone monitoring the growth of crops in a rural area, or perhaps an underwater vehicle? In the first part of this thesis (Papers I–V) we will look at specific algorithms suitable for indoor environments, particularly such where it is known that the mobile platform is travelling parallel to a floor, which could, *e.g.* be a moving manufacturing robot on a factory floor. In the second part (Papers VI and VII), we look into models where auxiliary IMU data is merged with the image data available from an onboard camera.

There is also a current trend concerning autonomous navigation using deep learning techniques where the classic SLAM system is replaced with an end-to-end network. Networks such as DeepVO [78] and LS-VO [11] are examples of well-performing networks comparable with state-of-the-art methods using geometric computer vision models. Typically, a convolutional neural network (CNN) replaces the feature extraction process (see Section 2.4) and the output is used for sequential learning. More recent methods explore self-supervised learning, taking advantage of adversarial learning [37].

5.3 Constrained Motion

In this section, we discuss how the SLAM problem changes when the motion is constrained. We treat two special cases in detail—the general planar motion model and the motion of a UAV with known IMU data. These cases are quite different mechanically. In the first case, the vehicle cannot move in all six degrees of freedom, nor the camera mounted on it, whereas, in the latter case, the IMU data provides us with information that constrains the remaining motion parameters; however, the mechanical motion of the UAV is not constrained.

Furthermore, we include derivations of the homographies and other trivial parts, which were otherwise omitted due to the lack of space provided for the conference papers.

5.3.1 The General Planar Motion Model

Consider a mobile platform, *e.g.* an industrial robot, travelling along a planar surface. Two cameras are assumed to be rigidly mounted on the platform, and we allow the

cameras to be tilted; however, the tilt is fixed (and possibly unknown) throughout the entire movement of the mobile platform. Furthermore, assume that the cameras are moving in the planes $z = a$ and $z = b$, respectively. In this setting, there is a fixed rigid body motion connecting the two camera centres, and, without loss of generality, we may assume the centre of rotation of the mobile platform is located in the first camera centre.

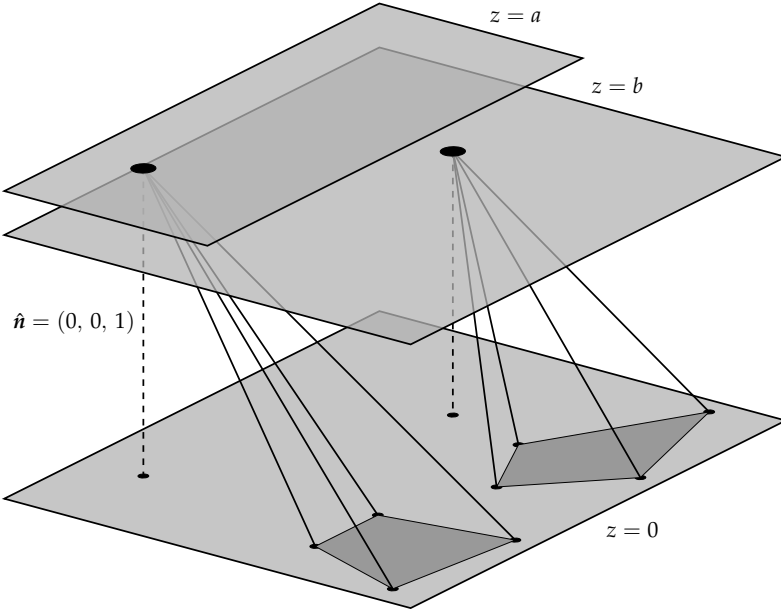


Figure 5.1: The problem geometry considered in the first part of this thesis. The cameras are assumed to move in the planes $z = a$ and $z = b$, respectively, and the relative pose between them, as well as the tilt towards the floor normal, is assumed to be fixed as the mobile platform moves freely.

The camera matrices for the first camera centre in two consecutive views, A and B , can be parameterised as

$$\begin{aligned} P_A &= \mathbf{R}_{\psi\theta}[\mathbf{I} \mid \mathbf{0}], \\ P_B &= \mathbf{R}_{\psi\theta}\mathbf{R}_\varphi[\mathbf{I} \mid -\mathbf{t}], \end{aligned} \quad (5.1)$$

where $\mathbf{R}_{\psi\theta}$ is a rotation θ about the y -axis followed by a rotation ψ about the x -axis, which are assumed to be fixed. The non-fixed parameters are the translation \mathbf{t} and the rotation φ , which depend on the motion of the mobile platform. Similarly, by accounting for the relative translation $\boldsymbol{\tau}$ and orientation $\boldsymbol{\eta}$, the corresponding poses for the second

camera are given by

$$\begin{aligned} P'_A &= \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b) [I \mid \mathbf{0}], \\ P'_B &= \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b) \mathbf{R}_\varphi [I \mid -\mathbf{t}], \end{aligned} \quad (5.2)$$

where $\mathbf{T}_\tau(b)$ is a translation matrix with respect to the vertical positioning of the camera. In Paper II we discuss in detail the consequences of having different planes and show how this affects the translation matrix.

The homography relating the image correspondences for the first camera is derived in [73], and is given by

$$\mathbf{H} \sim \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi \mathbf{T}_t \mathbf{R}_{\psi\theta}^T. \quad (5.3)$$

For the second camera, the derivation follows analogously; namely, consider a scene point \mathbf{X} imaged in \mathbf{x}'_A and \mathbf{x}'_B , respectively. Then the homography \mathbf{H}' describes the relation between the image correspondences as follows

$$\begin{aligned} \mathbf{x}'_B \sim \mathbf{H}' \mathbf{x}'_A &\Leftrightarrow P'_B \mathbf{X} \sim \mathbf{H}' P'_A \mathbf{X}, \\ &\Leftrightarrow \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b) \mathbf{R}_\varphi \mathbf{T}_t(b) \bar{\mathbf{X}} \sim \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b) \bar{\mathbf{X}}, \end{aligned} \quad (5.4)$$

where $\bar{\mathbf{X}} \in \mathbb{R}^3$ is the inhomogeneous representation of \mathbf{X} . Since \mathbf{X} was chosen arbitrarily this must hold for all $\mathbf{X} \in \mathbb{P}^3$, hence

$$\mathbf{H}' \sim \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b) \mathbf{R}_\varphi \mathbf{T}_t(b) \mathbf{T}_\tau(b)^{-1} \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T. \quad (5.5)$$

The rigid body motion constraint, due to the relative translation $\boldsymbol{\tau}$ joining the camera centres, affects the translational component of the second camera centre. We will discuss this geometric interpretation briefly, as it is necessary to understand in order to convert a sequence of relative poses to absolute poses. Assume the mobile platform moves in a direction given by the translation vector \mathbf{t} without rotating about the floor normal. Then, the second camera centre should move the same physical distance; however, due to the relative orientation η the local representation of the translation may differ. Since the world coordinate system is chosen such that the first camera centre is located in the centre of rotation of the mobile platform, the second camera centre is subject to a translation depending on the rotation by φ , as well as the relative translation. This is illustrated in Figure 5.2.

Absolute and Relative Poses

As we now know, a homography maps image correspondences between two images. Given a sequence of images, a homography can be estimated between every pair of images, assuming there are enough image correspondences. The main idea in the first part

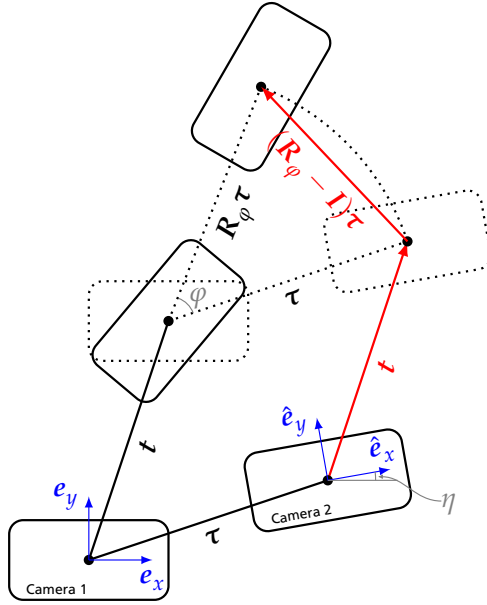


Figure 5.2: The translational component, when treated as a monocular system, is dependent on the relative translation from the origin and the angles φ and η . The dotted rectangular boxes illustrate the intermediate step without the impact of the rotation by φ of the mobile platform.

of this thesis (Papers I–V) is to decompose the inter-image homographies in order to extract information about the relative pose between the images, and thus the position of the mobile platform.

Assume that a number of homographies are given, and that one is able to extract the relative pose between the camera positions from them. How to estimate the trajectory of the mobile platform from these estimates? As discussed in the previous section, the local coordinate system has to be taken into account. Let $\Delta \mathbf{t}_i$ denote the relative translation at a given time i , in a local coordinate system, and $\Delta \varphi_i$ the corresponding angle. The absolute position \mathbf{t} and angle φ is then given by

$$\mathbf{t}_{i+1} = \mathbf{R}_{\varphi_i}^T \Delta \mathbf{t}_{i+1} + \mathbf{t}_i = \sum_{k=0}^i \mathbf{R}_{\varphi_k}^T \Delta \mathbf{t}_{k+1}, \quad (5.6)$$

where $\varphi_i = \sum_{k=0}^i \Delta \varphi_k$, with $\mathbf{t}_0 = (0, 0, 0)$ and $\varphi_0 = 0$. The inverse problem, *i.e.* given absolute poses, compute the corresponding relative poses, leads to the relations $\Delta \mathbf{t}_{i+1} = \mathbf{R}_{\varphi_i} (\mathbf{t}_{i+1} - \mathbf{t}_i)$ and $\Delta \varphi_{i+1} = \varphi_{i+1} - \varphi_i$.

Parameter Recovery of the Monocular Case

Trying to estimate the fundamental matrix from the traditional eight-point algorithm fails since all points lie in a plane, *cf.* Result 7. One possible solution is to use the inter-image homography and decompose it to extract information about the camera positions. During the last twenty years, several papers have considered this or similar approaches, and many of them will be discussed in the papers as related work. We must, however, discuss the work by Wadenbäck and Heyden [75, 76] in more detail to understand the pipeline of the intended VO system, as some of the algorithms presented in this thesis are an extension to their work for monocular systems.

In the first paper [76] the camera matrices and the homography $\mathbf{H} \sim \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi \mathbf{T}_t \mathbf{R}_{\psi\theta}^T$ were derived, as in Section 5.3.1. The algorithm proceeds to estimate $\mathbf{R}_{\psi\theta}$ by an iterative method, where one of the angles is fixed, while the other is solved for, as a coordinate-descent optimisation scheme. This is possible by observing, after some algebraic manipulation of (5.3), that

$$\mathbf{R}_{\psi\theta}^T \mathbf{M} \mathbf{R}_{\psi\theta} \sim \mathbf{T}^T \mathbf{T} = \begin{bmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \\ -t_x & -t_y & 1 + |\mathbf{t}|^2 \end{bmatrix}, \quad (5.7)$$

where $\mathbf{M} = \mathbf{H}^T \mathbf{H}$. The top-left 2×2 submatrix of the right-hand side is independent of the translation \mathbf{t} , whereas the left-hand side of the same submatrix only depends on the unknown tilt angles and the known homography. By denoting the left-hand side of (5.7) by \mathcal{L} , this is equivalent to

$$\begin{cases} \mathcal{L}_{11} - \mathcal{L}_{22} = 0, \\ \mathcal{L}_{12} = 0. \end{cases} \quad (5.8)$$

From this observation we wish to extract the unknown tilt angles, but instead of solving for both simultaneously, we fix one and solve for the other iteratively. Thus, fix θ and form the matrix $\hat{\mathbf{M}}_\theta = \mathbf{R}_\theta^T \mathbf{M} \mathbf{R}_\theta$, where \mathbf{R}_θ is a rotation about the y -axis by an angle θ , which together with (5.7) and some algebraic manipulation lead to the following relation

$$\begin{bmatrix} \hat{m}_{11} - \hat{m}_{22} & -2\hat{m}_{23} & \hat{m}_{11} - \hat{m}_{33} \\ \hat{m}_{12} & \hat{m}_{13} & 0 \\ 0 & \hat{m}_{12} & \hat{m}_{13} \end{bmatrix} \begin{bmatrix} \cos^2 \psi \\ \cos \psi \sin \psi \\ \sin^2 \psi \end{bmatrix} = \mathbf{0}. \quad (5.9)$$

The solution to (5.9) is equivalent to finding the null space of the coefficient matrix in (5.9), which can be found using SVD. Given the null space vector $\mathbf{v} = (v_1, v_2, v_3)$

the estimated value for ψ may be obtained as

$$\psi = \frac{1}{2} \arcsin \frac{2v_2}{v_1 + v_3}. \quad (5.10)$$

Once an estimated value of ψ has been found the matrix $\hat{\mathbf{M}}_\psi = \mathbf{R}_\psi^T \mathbf{M} \mathbf{R}_\psi$, can be used to solve a similar null space problem. This process is repeated until convergence, and when the overhead tilt $\mathbf{R}_{\psi\theta}$ is estimated the angle φ and translation \mathbf{t} can be extracted by QR-decomposition of $\mathbf{R}_{\psi\theta}^T \mathbf{H} \mathbf{R}_{\psi\theta}$.

The main contribution of the second paper [75] is to note that the null space of $\hat{\mathbf{M}}_\psi$ and $\hat{\mathbf{M}}_\theta$ should be the same for all homographies, regardless of the motion parameters, *i.e.* given homographies $\mathbf{H}_1, \dots, \mathbf{H}_N$, the corresponding coefficient matrices in (5.9), denoted Ψ_i , can be used to estimate ψ from all homographies simultaneously by finding the null space of

$$\Psi = \begin{bmatrix} \Psi_1 \\ \vdots \\ \Psi_N \end{bmatrix}. \quad (5.11)$$

Similarly, the same approach can be used for estimating θ from multiple homographies. By simultaneously using more than one homography the robustness of the overhead tilt estimation is increased as well as the accuracy.

In Wadenbäck *et al.* [77] the method using multiple homographies was used to initially estimate the tilt of the mobile platform. It was found that after approximately 15–20 frames (or 2 seconds of motion) the tilt estimation reached convergence, thus being able to translate the subsequent problem to a two-dimensional rigid body motion problem, for which the homography estimation step is not necessary.

A Minimal 2.5-Point Solver for Planar Motion

An alternative to the DLT equations for homography estimation (Section 3.2) is the method by Wadenbäck *et al.* [74], which is compatible with the general planar motion model. Since the homography (5.3) has five degrees of freedom the minimal problem is given by 2.5-point correspondences. A solver to such a system can be obtained by considering the system of polynomial equations defined by $\lambda \mathbf{H} - \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi \mathbf{T}_t \mathbf{R}_{\psi\theta}^T = \mathbf{0}$, which gives eleven quartic constraints on \mathbf{H} . The DLT system consisting of the constraints generated by 2.5-point correspondences has a four-dimensional null space, which allows a parameterisation of the homography in terms of the basis vectors of the null space, *i.e.* $\mathbf{H} = \sum_{i=1}^4 k_i \mathbf{H}_i$, where one may assume $k_1 = 1$ as it is only possible to recover

the homography up to scale. Inserting this parameterisation of \mathbf{H} into the quartic constraints gives eleven new quartic constraints in three variables, and by using the method described in Section 6.4 a compatible homography is obtained.

In Paper III we will explore alternative, non-minimal solvers, which are shown to be significantly faster than the existing 2.5-point solver, and show that they remain robust in cases where the planar motion model is invalid, whereas the minimal 2.5-point solver does not. Furthermore, we incorporate the division model (from Section 2.3) into the general planar motion model in Paper IV. This enables us to compensate for radial distortion, without the need for pre-calibration procedures.

5.3.2 Utilising IMU Data

In this section, we look at how to incorporate IMU data in our methods. We will assume that the IMU data is pre-processed—this can be achieved using *e.g.* filtering techniques—which gives us two angles of the corresponding rotation matrix. Since a UAV can move in all six degrees of freedom, the relative pose problem consists of finding a translation vector $\mathbf{t} \in \mathbb{R}^3$, as well as the unknown rotational angle, *i.e.* there are in total four degrees of freedom in the calibrated case.

Camera Models for UAVs

Let us assume that the gravity direction is aligned with the y -axis, which is the common reference direction given by the IMU. Furthermore, the pre-processed IMU data for the first pose is encoded in \mathbf{R}_1 and, similarly in \mathbf{R}_2 , for the second pose. The resulting camera matrices are now given by¹

$$\begin{aligned} \mathbf{P}_1 &= \mathbf{K}_1[\mathbf{R}_1 \mid \mathbf{0}], \\ \mathbf{P}_2 &= \mathbf{K}_2[\mathbf{R}_2 \mathbf{R}_y \mid \mathbf{t}], \end{aligned} \tag{5.12}$$

where \mathbf{R}_y is the unknown rotation about the yaw angle (y -axis). In the second part of the thesis (Papers VI and VII) we will explore indoor positioning methods for UAVs, and thus continue using homographies for tracking. Therefore, consider a scene point $\mathbf{X} \in \mathbb{P}^3$ on a plane π . Projecting the first point $\mathbf{x}_1 \sim \mathbf{P}_1 \mathbf{X} = \mathbf{K}_1 \mathbf{R}_1 \bar{\mathbf{X}}$, where $\bar{\mathbf{X}} \in \mathbb{R}^3$ is the inhomogeneous representation of \mathbf{X} , and applying the plane constraint $\boldsymbol{\pi}^T \mathbf{X} = 0$, where $\boldsymbol{\pi} = (\mathbf{n}, 1)$, gives $\mathbf{X} = (\mathbf{R}_1^T \mathbf{K}_1^{-1} \mathbf{x}_1, -\mathbf{n}^T \mathbf{R}_1^T \mathbf{K}_1^{-1} \mathbf{x}_1)$. Now, projecting the second point yields

$$\mathbf{R}_2^T \mathbf{K}_2^{-1} \mathbf{x}_2 \sim (\mathbf{R}_y - \mathbf{R}_2^T \mathbf{t} \mathbf{n}^T) \mathbf{R}_1^T \mathbf{K}_1^{-1} \mathbf{x}_1, \tag{5.13}$$

¹Here we deviate from our standard canonical form in order to conform with other authors.

where we recognise the homography (in normalised and rectified space)

$$\mathbf{H}_y \sim \mathbf{R}_y - \hat{\mathbf{t}}\mathbf{n}^T, \quad (5.14)$$

where $\hat{\mathbf{t}} = \mathbf{R}_2^T \mathbf{t}$. In the calibrated case, the vectors

$$\mathbf{y}_j^{(i)} := \mathbf{R}_j^T \mathbf{K}_j^{-1} \mathbf{x}_i, \quad (5.15)$$

can be pre-computed, and as such, the only unknowns are embedded in \mathbf{H}_y from (5.14). The relation to the full homography \mathbf{H} , such that $\mathbf{x}_2 \sim \mathbf{H}\mathbf{x}_1$, is given by

$$\mathbf{H}_y \sim \mathbf{R}_2^T \mathbf{K}_2^{-1} \mathbf{H} \mathbf{K}_1 \mathbf{R}_1. \quad (5.16)$$

This, and other geometrical properties, was used by Ding *et al.* in [16]. In Papers VI and VII we leverage this further by considering the floor normal only—this allows us to construct significantly faster solvers, suitable for real-time applications. Furthermore, we show that by incorporating IMU data using filtering techniques, we are able to do further intrinsic calibration, such as correcting for radial distortion.

Chapter 6

Minimal Problems in Computer Vision

In previous chapters, we have seen a method for estimating the fundamental matrix and another similar method for estimating a homography between image correspondences, when the scene points are coplanar. In most systems such image correspondences will be automatically generated by *e.g.* matching SIFT descriptors, as described in Section 2.4. This is, however, not a foolproof method, and mismatched features will occur, and must therefore be recognised and discarded by the framework in order to get a robust method. A common way of achieving this is by using RANSAC or one of the many derivations thereof.

6.1 Random Sampling Consensus

Assuming that the number of outliers is small, the probability of selecting a subset of point correspondences that are inliers is relatively high. From this simple heuristic, the essence of *random sampling consensus* (RANSAC) is captured. Without going into detail, the algorithm can be outlined as follows

1. Select a small subset of point correspondences, and solve the problem,
2. Evaluate the residuals for all point correspondences (for a suitable cost function),
3. Repeat and select the solution with the largest *consensus set*.

Here, the set of measurements with residuals smaller than a specified threshold is called the consensus set. By choosing the minimal subset required to obtain the solution the number of iterations necessary to pick a subset containing only inliers, by a certain probability, is as low as possible. This could, at least theoretically, decrease the total time of

determining the largest consensus set, and thus result in a fast algorithm. Examples of cost functions will be discussed in Chapter 7.

6.2 Minimal Solvers

A solver that uses the minimal number of points needed for a specific problem is known as a *minimal solver*. Estimating the homography from the DLT system requires four point correspondences as the homography has eight degrees of freedom. Furthermore, we need at least eight point correspondences for the eight-point algorithm in order to estimate the fundamental matrix; however, this is not a minimal solver. In fact, due to Result 6, the fundamental matrix has seven degrees of freedom, but the determinant constraint is cubic. One cannot apply the method of estimating the null space using SVD, if trying to incorporate such a constraint. Similarly, we saw the trace constraint from Result 8, that arises for the case of calibrated cameras, which is also nonlinear.

Polynomial systems of equations frequently arise in computer vision, and other methods must be used in order to solve these. We shall briefly explain the theory behind one such method.

6.3 Solving Polynomial Systems of Equations

We will introduce some notation from algebraic geometry in order to outline the theory behind the *action matrix method*. Using multi-index notation, let \mathbf{x}^α denote a *monomial* of degree $|\alpha|$. Our main goal is to solve a polynomial system of equations

$$\begin{aligned} f_1(\mathbf{x}) &= 0, \\ &\vdots \\ f_s(\mathbf{x}) &= 0, \end{aligned} \tag{6.1}$$

where each polynomial equation can be expressed as $f = \sum_{\alpha} c_{\alpha} \mathbf{x}^{\alpha}$. The set of all solutions to (6.1) is called an *affine variety*, and is denoted $\mathbf{V}(f_1, \dots, f_s) \subset \mathbb{C}$. Let $\mathbb{C}[\mathbf{x}]$ denote the set of polynomials in \mathbf{x} with coefficients in \mathbb{C} . An *ideal* $I \subset \mathbb{C}[\mathbf{x}]$ is an additive group satisfying the *absorption property*, i.e. if $f \in I$ and $h \in \mathbb{C}[\mathbf{x}]$ then $hf \in I$. Moreover, given a set of polynomials $f_1, \dots, f_s \in \mathbb{C}[\mathbf{x}]$ we consider the ideal

$$\langle f_1, \dots, f_s \rangle = \left\{ \sum_{i=1}^s h_i f_i : h_1, \dots, h_s \in \mathbb{C}[\mathbf{x}] \right\}, \tag{6.2}$$

which we will refer to as the *ideal generated by* f_1, \dots, f_s . The *Hilbert Basis Theorem* states that every ideal of $\mathbb{C}[\mathbf{x}]$ is finitely generated, *i.e.* given an ideal $I \subset \mathbb{C}[\mathbf{x}]$ there exist $f_1, \dots, f_s \in \mathbb{C}[\mathbf{x}]$ such that $I = \langle f_1, \dots, f_s \rangle$. Thus, the polynomial system of equations (6.1) is defined by the generated ideal; however, the basis is in general not unique.

To be able to represent the system of equations uniquely, one must impose a *monomial ordering*, *e.g.* the lexicographic order, the graded lex order or the graded reverse lex order. Regardless of which monomial ordering is chosen the *leading term* (w.r.t. the monomial ordering) is uniquely defined, and we shall denote it $\text{LT}(f)$. Furthermore, for an ideal I define $\text{LT}(I) := \{\text{LT}(f) : f \in I\}$. Then a finite subset $G = \{g_1, \dots, g_t\} \subset I$ is a *Gröbner basis* if $\langle \text{LT}(g_1), \dots, \text{LT}(g_t) \rangle = \text{LT}(I)$.

After this exposition one may ask: how does this relate to solving polynomial systems of equations? The answer lies in the fact that we have efficient ways of computing a Gröbner basis, by means of *Buchberger's algorithm*. For more details regarding the algorithm, and to get a deeper understanding of the subject, the work of Cox *et al.* [12, 13] is highly recommended.

6.4 The Action Matrix Method

Let us return to the polynomial system of equations (6.1). Under the assumption that the system has finitely many solutions, *i.e.* when $V(I)$ is finite, it follows by the Finiteness Theorem (see *e.g.* [13]) that I is zero-dimensional and the quotient space $A = \mathbb{C}[\mathbf{x}]/I$ is finite dimensional. Given the *coset* $[f] = \{f + h : h \in I\}$, consider the operator $T_f : A \rightarrow A$, defined by $T_f([g]) = [fg]$. It is easily seen that T_f is linear with the property that $T_f = T_g$ if and only if $f - g \in I$. Since the quotient space A is finite-dimensional this operation can be represented by a matrix M_f , which is known as the *action matrix*. Furthermore, we may select a basis for A , *e.g.* a monomial basis $\mathcal{B} = \{[\mathbf{x}^{\alpha_j}]\}_{j \in J}$, typically obtained by (an improved version of) Buchberger's algorithm. When the action matrix $M_f = (m_{ij})$ acts on the basis elements we obtain a linear combination of the monomials forming the basis, namely

$$T_f([\mathbf{x}^{\alpha_j}]) = [f\mathbf{x}^{\alpha_j}] = \sum_{i \in J} m_{ij}[\mathbf{x}^{\alpha_i}]. \quad (6.3)$$

This implies that, for $\mathbf{x} \in V(I)$,

$$f(\mathbf{x})\mathbf{x}^{\alpha_j} = \sum_{i \in J} m_{ij}\mathbf{x}^{\alpha_i}. \quad (6.4)$$

By representing the basis \mathcal{B} with a vector \mathbf{b} , and using the fact that (6.4) must hold for all basis elements, the problem can be reduced to

$$f(\mathbf{x})\mathbf{b}(\mathbf{x}) = \mathbf{M}_f^T \mathbf{b}(\mathbf{x}), \quad (6.5)$$

which we recognise as an eigenvalue problem. Let us recapitulate: given a polynomial system of equations, use Buchberger's algorithm to obtain a Gröbner basis. Create the action matrix and compute the eigenvalues and eigenvectors to extract the solutions.

Chapter 7

Bundle Adjustment

The theory so far has dealt with minimising the error locally between two consecutive images; however, small errors propagate over time and the global prediction of the trajectory suffers. Furthermore, the DLT equations, and similar methods, measure an algebraic error, which is not an ideal estimation of the geometric error. As a consequence of these observations, most modern SLAM systems do at some point minimise the *geometric reprojection error* globally; however, exceptions exist and are discussed in Section 7.5. This is known as *bundle adjustment*, as the bundle of rays passing through the camera centres is adjusted to the scene points. In this section, we will try to convey the general idea, and leave the details to Paper V regarding how to efficiently enforce the planar motion model in a bundle adjustment algorithm.

7.1 The Geometric Reprojection Error

Let $\mathbf{x}_i^{(j)} \in \mathbb{R}^2$ denote the measured image point in camera $P^{(j)}$ of the scene point X_i , which, due to noise, may differ from the estimated measurement, *i.e.* the corresponding reprojected image point $\hat{\mathbf{x}}_i^{(j)} \sim P^{(j)} X_i$. A popular cost function for bundle adjustment is the sum of squared distances between the reprojected points and the measured points, known as the *geometric reprojection error*. There is a strong argument for this; namely, it is the maximum likelihood estimation under the assumption of Gaussian noise. Assuming all image points are visible in all views, we may formulate it as a minimisation problem

$$\min_{P^{(j)}, X_i} \sum_{i=1}^M \sum_{j=1}^N \|\mathbf{x}_i^{(j)} - \hat{\mathbf{x}}_i^{(j)}\|^2, \quad (7.1)$$

where $\hat{\mathbf{x}}_i^{(j)}$ is the inhomogeneous representation of $\hat{\mathbf{x}}_i^{(j)}$. This is an example of a nonlinear least-squares problem, which in general can be written as

$$\min_{\mathbf{x}} \|\boldsymbol{\varepsilon}(\mathbf{x})\|_2^2, \quad (7.2)$$

where $\mathbf{x} \in \mathbb{R}^m$ and $\boldsymbol{\varepsilon} : \mathbb{R}^m \rightarrow \mathbb{R}^n$ is the residual vector.

7.2 The Levenberg–Marquardt Algorithm

The objective function (7.2) is generally non-convex and methods constructed with the purposes of solving such problems are usually iterative. If they converge, one cannot, in general, guarantee global optimality, but can only hope to find a local optimum. Therefore, it is often considered beneficial to have a good initial guess, although recent advances in the field suggest that this may not be necessary if other objectives are considered instead. We will discuss such methods in Section 7.5, and return to them in the last part of this thesis (Papers VIII–XII). In the first part of this thesis (Papers I–V), we will develop a method for extracting a good initial guess (Papers I–II), and use it to globally minimise the geometric reprojection error (Paper V), as defined in (7.1). In order to solve (7.1) we will use the *Levenberg–Marquardt algorithm* (LM) [35, 42], which we will discuss now.

Using the general form (7.2), consider a neighbourhood $\mathcal{N}_{\Delta\mathbf{x}}(\mathbf{x})$ in which the Taylor expansion $\boldsymbol{\varepsilon}(\mathbf{x} + \Delta\mathbf{x}) \approx \boldsymbol{\varepsilon}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\Delta\mathbf{x}$ holds. Applying this to each subsequent step in the iteration yields $\boldsymbol{\varepsilon}(\mathbf{x}_{i+1}) = \boldsymbol{\varepsilon}(\mathbf{x}_i + \Delta\mathbf{x}_i) \approx \boldsymbol{\varepsilon}(\mathbf{x}_i) + \mathbf{J}(\mathbf{x}_i)\Delta\mathbf{x}_i$. Hence, each step should minimise

$$\min_{\Delta\mathbf{x}_i} \|\boldsymbol{\varepsilon}_i + \mathbf{J}_i\Delta\mathbf{x}_i\|_2^2, \quad (7.3)$$

where $\boldsymbol{\varepsilon}_i := \boldsymbol{\varepsilon}(\mathbf{x}_i)$ and $\mathbf{J}_i := \mathbf{J}(\mathbf{x}_i)$. The approximated expression (7.3) is the linear least-squares problem, which is known to have a closed-form solution that coincides with solution to the *normal equations*, given by

$$\mathbf{J}^T \mathbf{J} \Delta\mathbf{x}_i = \mathbf{J}^T \boldsymbol{\varepsilon}_i. \quad (7.4)$$

In general, there is no guarantee that the solution to (7.4) gives an error reduction, as it assumes that $\boldsymbol{\varepsilon}$ can be approximated well by the first order Taylor expansion in $\mathcal{N}_{\Delta\mathbf{x}}(\mathbf{x})$; however, this is not known *a priori*. One possible solution is to use the Levenberg–Marquardt algorithm which solves the *augmented normal equations*

$$(\mathbf{J}^T \mathbf{J} + \mu \mathbf{I}) \Delta\mathbf{x}_i = \mathbf{J}^T \boldsymbol{\varepsilon}_i. \quad (7.5)$$

The damping factor $\mu > 0$ can be tuned to ensure that the error is reduced in every step. Note that if μ is made large enough the sum $\mathbf{J}^T \mathbf{J} + \mu \mathbf{I}$ is diagonally dominant, hence positive definite, which implies that the update moves in the direction of the gradient, as in the *steepest descent algorithm*. On the other hand, assuming that the linearisation approximates the objective well in an increasing neighbourhood around the current point, we may choose decreasing values of μ . Secondly, note that the eigenvalues are shifted along the real axis by the damping factor μ , and therefore convergence can be guaranteed if the right-hand side is positive definite, *i.e.* when all eigenvalues are positive. In practice, however, the damping factor μ is controlled by some simple heuristics, as an eigendecomposition is too expensive to compute. If such an approach is used one would pick the first value of μ that reduces the reprojection error, update the solution accordingly, and repeat the process until the value of the cost function does not decrease by more than a predefined threshold, or some other suitable stopping criteria.

7.3 Separable Nonlinear Least-Squares Problems

Certain objectives that arise naturally in computer vision are separable least-squares problems, meaning that they can be written as

$$\|\boldsymbol{\varepsilon}(\mathbf{u}, \mathbf{v})\|_2^2 = \|\mathbf{A}(\mathbf{u})\mathbf{v} - \mathbf{b}(\mathbf{u})\|_2^2, \quad (7.6)$$

for variables $\mathbf{u} \in \mathbb{R}^p$ and $\mathbf{v} \in \mathbb{R}^q$, and (possibly) nonlinear operators $\mathbf{A} : \mathbb{R}^p \rightarrow \mathbb{R}^{t \times q}$ and $\mathbf{b} : \mathbb{R}^p \rightarrow \mathbb{R}^t$.

One could, of course, treat this objective as a general nonlinear least-squares problem of the form (7.2) and simply stack the variables \mathbf{u} and \mathbf{v} , which is known as *joint optimisation*. We will look at another method known as the variable projection method (VarPro), originally proposed by Golub and Pereyra [24]. While the method has been around for many decades, the benefits of applying it to computer vision problems just recently emerged in a series of publications by Hong *et al.* [29–31]. Essentially, it has been empirically established that the basin of convergence is significantly larger when applying VarPro compared to LM—it is even possible to converge to the (best known) optima for several difficult real-life datasets using only random initial starting points, thus making complex initialisation schemes unnecessary.

We use the nomenclature introduced in [31]. The method makes use of the following observation: Fix \mathbf{u} and solve for the second variable

$$\mathbf{v}^*(\mathbf{u}) := \arg \min_{\mathbf{v}} \|\mathbf{A}(\mathbf{u})\mathbf{v} - \mathbf{b}(\mathbf{u})\|_2^2 = \mathbf{A}(\mathbf{u})^+ \mathbf{b}(\mathbf{u}). \quad (7.7)$$

Now, inserting (7.7) back into the objectives, gives a problem in \mathbf{u}

$$\min_{\mathbf{u}} \|\boldsymbol{\varepsilon}^*(\mathbf{u})\|_2^2 := \min_{\mathbf{u}} \|\boldsymbol{\varepsilon}(\mathbf{u}, \mathbf{v}^*(\mathbf{u}))\|_2^2 = \min_{\mathbf{u}} \|\left(\mathbf{A}(\mathbf{u})\mathbf{A}(\mathbf{u})^+ - \mathbf{I}\right)\mathbf{b}(\mathbf{u})\|_2^2, \quad (7.8)$$

where the objective can be simplified to (see *e.g.* [31] for details)

$$\boldsymbol{\varepsilon}^*(\mathbf{u}) = \left(\mathbf{I} - \mathbf{J}_v(\mathbf{u})\mathbf{J}_v(\mathbf{u})^+\right)\boldsymbol{\varepsilon}(\mathbf{u}, \mathbf{v}). \quad (7.9)$$

Now, consider solving (7.8) using LM, in which case we seek the Jacobian $\frac{d\boldsymbol{\varepsilon}^*(\mathbf{u})}{d\mathbf{u}}$. Applying the chain rule, we need to evaluate $\frac{d\mathbf{v}^*(\mathbf{u})}{d\mathbf{u}}$, which requires one to differentiate a pseudo-inverse. While this can be done analytically, an approximation known as RW2 (Ruhe and Wedin Algorithm 2, after the second algorithm proposed by the original authors in [51]), is often used. The approximation gives the update

$$\frac{d\boldsymbol{\varepsilon}^*(\mathbf{u})}{d\mathbf{u}} = \left(\mathbf{I} - \mathbf{J}_v(\mathbf{u})\mathbf{J}_v(\mathbf{u})^+\right)\mathbf{J}_u(\mathbf{u}, \mathbf{v}^*(\mathbf{u})), \quad (7.10)$$

which has been shown to have similar convergence properties as the full pseudo-inverse update, while enjoying reduced computational complexity.

We will use VarPro in the third part of this thesis (Papers VIII–XII), as our second order method of choice, for a variety of different objectives found in computer vision literature.

7.4 Schur Complement

In Paper V, it is shown that one does not need to compute and store the complete Jacobian, but only the non-zero elements. This speeds up performance significantly and makes it tractable for real-time applications. The main idea is to decompose the Jacobian using the *Schur complement*. Assume that the Jacobian \mathbf{J} may be decomposed into the following block structure

$$\mathbf{J} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}, \quad (7.11)$$

and define the Schur complement \mathbf{S} of block \mathbf{D} as $\mathbf{S} := \mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}$. It follows that

$$\begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{D}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}. \quad (7.12)$$

The form (7.12) can be used to boost performance if the size of \mathbf{A} , and thus \mathbf{S} , is significantly smaller than \mathbf{D} , which is typically the case for bundle adjustment. Due to

the top-right zero block in (7.12), the parameters related to \mathbf{A} can be solved for by back-substitution. Furthermore, as noted in Section 7.2, positive definiteness is an attractive property, that can guarantee convergence in certain cases, which is why the following result is of interest to us. We use the notation $\mathbf{X} \succ 0$ to denote that \mathbf{X} is positive definite.

Result 9. *Let $\mathbf{J} \succ 0$ be a symmetric matrix, of the form*

$$\mathbf{J} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}, \quad (7.13)$$

where \mathbf{C} is invertible. Then $\mathbf{C} \succ 0$ and the Schur complement $\mathbf{S} = \mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^T \succ 0$.

Proof. Define

$$\mathbf{X} = \begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{C}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (7.14)$$

and note that

$$\mathbf{X}\mathbf{J}\mathbf{X}^T = \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{C} \end{bmatrix}. \quad (7.15)$$

Congruence transformations preserve positive definiteness, *i.e.* for any square matrices \mathbf{Y} and \mathbf{Z} , where \mathbf{Z} is invertible, the matrix $\mathbf{Y} \succ 0$ if and only if $\mathbf{Z}\mathbf{Y}\mathbf{Z}^T \succ 0$. Furthermore, a block diagonal matrix is positive definite if and only if each block is positive definite, which concludes the proof. \square

If $\mathbf{S} \succ 0$, the *Cholesky decomposition* can be used to solve the system

$$\mathbf{S}\mathbf{X} = \mathbf{B}. \quad (7.16)$$

This is done by decomposing $\mathbf{S} = \mathbf{L}\mathbf{L}^T$ where \mathbf{L} is a lower triangular matrix. The solution to (7.16) can thus be obtained by solving $\mathbf{L}\mathbf{Y} = \mathbf{B}$, by forward-substitution with $\mathbf{Y} = \mathbf{L}^T\mathbf{X}$, followed by back-substitution of $\mathbf{L}^T\mathbf{X} = \mathbf{Y}$. In general, the Cholesky decomposition is of order $O(n^3)$ for a dense $n \times n$ matrix; however, it can be shown that for sparse matrices the computational complexity is dependent on the number of non-zero elements in the *Cholesky factor* \mathbf{L} , which is related to the sparsity pattern of \mathbf{S} . In some applications of SLAM, where the sparsity pattern is often very structured, the Cholesky decomposition can be obtained in $O(n)$ computations [41].

7.5 Alternative Approaches

Bundle adjustment can be time consuming, and the classic approach has been to establish an initial solution from pairwise views. If the initial reconstruction is good it increases the chances for the bundle adjustment algorithm to reach a good reconstruction result. Furthermore, the number of necessary iterations to reach a local minimum might decrease.

One way of boosting the performance of the initial solution was proposed in [43] for the case of 3D point-cloud registration. They show that computing the initial poses of cameras in small loops, *i.e.* solving the pose problem for groups of three, four, or five cameras, can achieve superior performance compared to pairwise methods. This is done using minimal solvers.

Another attractive way is to completely skip the initialisation step and rely on the bundle adjustment algorithm. It has been shown in a number of recent publications that the VarPro algorithm exhibits a large basin of convergence, which makes this a feasible approach [30, 31].

Minimising the reprojection error is not the only way to enforce consistency and feasibility of the 3D reconstruction. Another approach, that has been used successfully with dense reconstructions, is minimising the photometric error [1, 14, 79]. When minimising the reprojection error we rely on feature-based detectors to extract keypoints, which introduce errors, partly originating from pixel noise. The photometric error comes closer to the source by directly working with the pixels, and minimises the consistency of neighbouring image patches. In [79] a low memory VarPro algorithm was proposed, making it feasible to consider large scale models for photometric bundle adjustment.

Another important question to be raised is: “Are large-scale 3D models really necessary for accurate visual localisation?” which is also the title of the paper [53], where the authors consider geo-tagged 2D images, as an alternative to 3D models. The main argument for doing so is that large scale 3D models are hard to acquire and maintain, whereas the 2D models are significantly easier. They show that it is possible to use visual navigation with enough relevant databases, and get results comparable to the state-of-the-art.

Chapter 8

Low-Rank Matrix Completion and Applications to Computer Vision

Many problems in computer vision can be treated as low-rank approximation, *i.e.* a minimisation problem where one seeks to approximate a given data matrix while encouraging solutions with low rank. The Eckart–Young theorem is one of the earliest results in this field.

Result 10 (Eckart–Young [19]). *Let $\mathbf{X}_0 \in \mathbb{R}^{m \times n}$, where $m \geq n$, and consider finding the best rank- k approximation*

$$\min_{\text{rank}(\mathbf{X}) \leq k} \|\mathbf{X} - \mathbf{X}_0\|_F^2. \quad (8.1)$$

Let $\mathbf{X}_0 = \sum_{i=1}^n \sigma_i(\mathbf{X}_0) \mathbf{u}_i \mathbf{v}_i^T$ be an SVD of \mathbf{X}_0 , then

$$\mathbf{X}^* = \sum_{i=1}^k \sigma_i(\mathbf{X}_0) \mathbf{u}_i \mathbf{v}_i^T, \quad (8.2)$$

is an optimal solution to (8.1).

The Eckart–Young theorem is well-known in the community, and the fact that it has a closed-form solution is far from obvious. This can be illustrated through the following, seemingly trivial generalisation, that does not behave as well. Consider

$$\min_{\text{rank}(\mathbf{X}) \leq k} \|\mathbf{X} - \mathbf{X}_0\|_F^2 + c(\mathbf{X}), \quad (8.3)$$

where c is a “nice” convex function. In general, there is no closed-form solution to this problem, nor is it differentiable, or even convex. All of this together makes it difficult to use local optimisation, in order to find a global optimum.

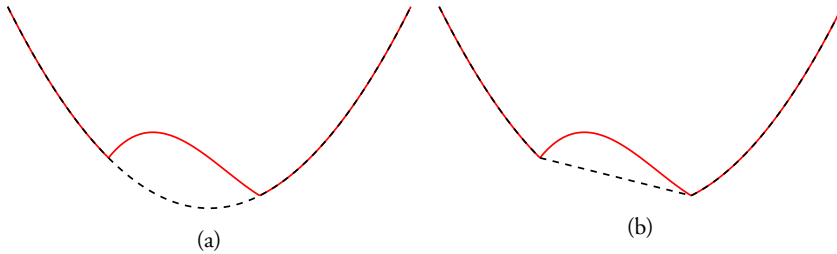


Figure 8.1: Examples of convexification. The function corresponding to the red curve is non-convex and has two local minima, where only one of them is globally optimal. If one would convexify the objective (black dashed line) as in (a), the global optimum is not preserved. When convexification is done with care, as in (b), the global optima of the convexified function and the original function coincide. In this case, the convex function is the convex envelope of the original function.

8.1 Convexification

One way of tackling problems of the form (8.3) is to use convexification, *i.e.* to replace the cost function with a convex counterpart. The benefit of this approach is that you can find the global minimum of your new cost function; however, this does not necessarily mean that it is the global minimum of the original problem formulation, which is illustrated in Figure 8.1.

It is natural to ask: how does one obtain a convexification such that the global optima are preserved? The answer is the *l.s.c. convex envelope* of $f : X \rightarrow \mathbb{R}$, denoted $\overline{\text{co}} f$, which is defined as

$$\overline{\text{co}} f(x) := \sup \{g(x) \mid g \text{ convex and } g \leq f \text{ on } X\} . \quad (8.4)$$

In fact, the convexification shown in Figure 8.1b is the convex envelope of the original function. Unfortunately, it can be hard to compute the convex envelope of a function; hence, this definition is of less practical importance. One way of explicitly computing the convex envelope of a function f is by using the *Fenchel conjugate*, denoted by f^* ,

$$f^*(\mathbf{y}) := \sup_x \langle \mathbf{x}, \mathbf{y} \rangle - f(\mathbf{x}) . \quad (8.5)$$

Note that for any choice $\mathbf{x} \in X$, the function $h(\mathbf{x}) = \langle \mathbf{x}, \mathbf{y} \rangle - f^*(\mathbf{y}) \leq f(\mathbf{x})$ is a hyperplane. Therefore, the conjugate can be interpreted as a supporting hyperplane with slope \mathbf{y} , see Figure 8.2a. Now, in order to construct the convex envelope, we simply need to consider the conjugate of the conjugate function, known as the biconjugate.

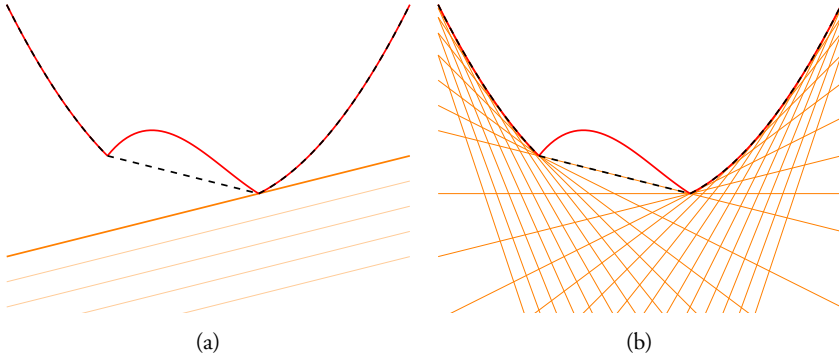


Figure 8.2: (a) Example of hyperplanes, $h(x) = xy - f(y)$, with the supremum marked (thicker line). (b) The pointwise maximum of the hyperplanes form the biconjugate, which is the l.s.c. convex envelope.

This is the pointwise supremum of all supporting hyperplanes, which is illustrated in Figure 8.2b.

Result 11. *The l.s.c. convex envelope is the biconjugate, i.e. for a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$*

$$\overline{\text{co}}(f) = (f^*)^* := f^{**}. \quad (8.6)$$

Computing the biconjugate is a useful tool; however, for more complex cost functions, one might not be able to compute it explicitly. For such problems, *convex relaxations* are an alternative, i.e. convex cost functions mimicking the original cost functions as much as possible, but perhaps without theoretical guarantees—such as ensuring that the global optima coincide.

In the following sections, we will focus on low-rank approximations. This means that we are interested in solving problems such as

$$\min_{\text{rank}(\mathbf{X}) \leq k} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (8.7)$$

where $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ is a linear operator and $\mathbf{b} \in \mathbb{R}^p$ is a data term. We can write this as

$$\min_{\mathbf{X}} \iota_k(\mathbf{X}) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (8.8)$$

where ι_k is the indicator function

$$\iota_k(\mathbf{X}) = \begin{cases} 0, & \text{if } \text{rank}(\mathbf{X}) \leq k, \\ \infty, & \text{else.} \end{cases} \quad (8.9)$$

When the rank of the sought matrix is not known *a priori*, one may also be interested in studying objectives of the form

$$\min_{\mathbf{X}} \mu \operatorname{rank}(\mathbf{X}) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (8.10)$$

where the parameter μ controls the impact of the rank—larger values of μ encourages solutions with lower rank. Sometimes we refer to (8.7) as the *hard rank problem*, whereas (8.10) is referred to as the *soft rank problem*. In both cases, however, for a general operator \mathcal{A} , one cannot find explicit formulations for the convex envelope, and is forced into convex relaxations (if possible).

8.2 The Nuclear Norm Heuristic

The rank function is non-convex and discontinuous, hence hard to optimise numerically. A common convex relaxation is therefore to use the nuclear norm, defined as the sum of the singular values

$$\|\mathbf{X}\|_* := \sum_{i=1}^n \sigma_i(\mathbf{X}). \quad (8.11)$$

This relaxation is theoretically justified, through the following result.

Result 12 (Fazel [20]). *The convex envelope of the rank function, restricted to the set $\{\mathbf{X} \in \mathbb{R}^{m \times n} \mid \sigma_1(\mathbf{X}) \leq 1\}$, is the nuclear norm.*

Consequently, instead of optimising (8.7) or (8.10) the convex surrogate

$$\min_{\mathbf{X}} \mu \|\mathbf{X}\|_* + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (8.12)$$

has been used in a multitude of applications, some of which exhibited state-of-the-art performance at the time. One issue, however, concerning this objective, is revealed through a simple example. Consider

$$\min_{\mathbf{X}} \mu \|\mathbf{X}\|_* + \|\mathbf{X} - \mathbf{X}_0\|_F^2 = \min_{\mathbf{X}} \sum_{i=1}^n \mu \sigma_i(\mathbf{X}) + (\sigma_i(\mathbf{X}) - \sigma_i(\mathbf{X}_0))^2, \quad (8.13)$$

where we restrict the optimisation to the singular values. By completing squares, we note that the sum is minimised if

$$\sigma_i(\mathbf{X}) = \left[\sigma_i(\mathbf{X}_0) - \frac{1}{2}\mu \right]_+, \quad (8.14)$$

where $[\cdot]_+ = \max\{0, \cdot\}$. In a typical scenario, we are observing a low-rank structure \mathbf{X}_0 , but noise in the measurement data obstructs us from retrieving the true signal.

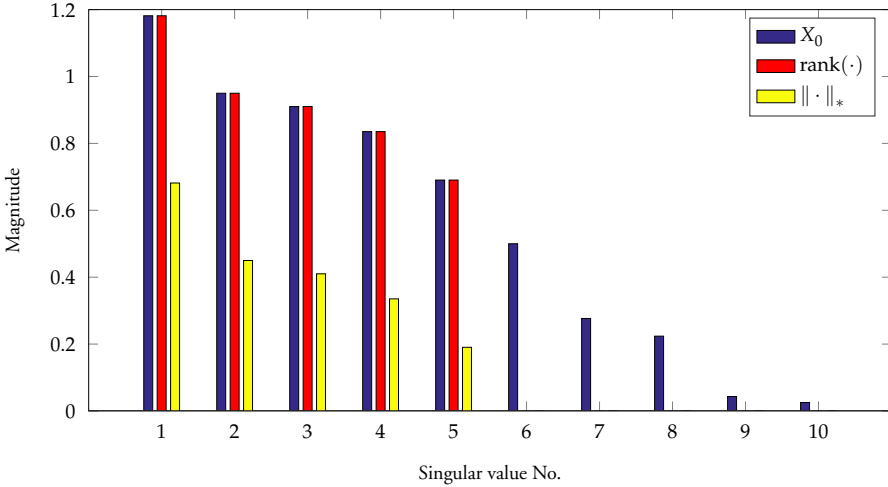


Figure 8.3: Given an initial measurement matrix with noise X_0 of rank ten, we wish to find the best rank-5 matrix. We would like to achieve what the rank penalty does (red), but with the nuclear norm we get a heavily altered solution (yellow).

This causes the smaller singular values to be non-zero. Ideally, as in the case of the Eckart–Young theorem, we would simply force these smaller singular values to be zero, while leaving the true signal (the larger singular values) untouched. This is where the problems of the nuclear norm emerge. Regardless of the size of the singular values, we remove at least $\frac{1}{2}\mu$ from the true signal, *cf.* (8.14), using this heuristic, *i.e.* suppression of noise requires equal or greater suppression of the true signal. This phenomenon is known as *shrinking bias*, and is illustrated in Figure 8.3. This motivates the use of stronger alternatives.

8.3 Non-Convex Relaxations

From the previous section, we see a need to penalise singular values differently—why penalise large singular values, when they are likely to be correlated to the sought structure? We may consider penalties such as

$$p(\mathbf{X}) = \sum_{i=1}^n f(\sigma_i(\mathbf{X})), \quad (8.15)$$

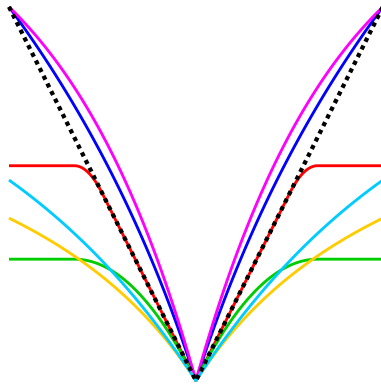


Figure 8.4: Different non-convex regularisers penalising smaller singular values proportionally harder than the larger ones. The nuclear norm (dashed) is used as a reference.

for some function f . In order to penalise small singular values proportionally harder than the larger ones, these functions typically have non-increasing derivatives on $[0, \infty)$, *i.e.* are concave. In Figure 8.4 we show a variety of such penalties acting on the singular values, where the dashed line corresponds to the nuclear norm.

First of all, since we left the convex realm, we are dealing with potentially discontinuous objectives. Furthermore, far from all penalties on the form (8.15) are theoretically justified. We still want to solve problems of the form (8.7) or (8.10), and therefore we must ask ourselves: when do the global minima coincide? A problem with the nuclear norm approach is that it does not use the potential contribution from the convex datafit term $\|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2$. For general \mathcal{A} and \mathbf{b} there are no explicit expressions available, but one may consider the simple cases

$$f_k(\mathbf{X}) = \iota_k(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2, \quad (8.16)$$

and

$$f_\mu(\mathbf{X}) = \mu \operatorname{rank}(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2. \quad (8.17)$$

Now, including the datafit terms, what are the convex envelopes of f_k and f_μ ? By now we know that this can be answered by computing f_k^{**} and f_μ^{**} , respectively. Successfully doing so—hopefully with explicit expressions—can lead to global minima not moving. In [33] these objectives were studied, or, in fact, a family of functions, in which (8.16)

and (8.17) are special cases. For the soft rank problem, it was found that

$$f_{\mu}^{**}(\mathbf{X}) = \mathcal{R}_{\mu}(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2, \quad (8.18)$$

where the regulariser \mathcal{R}_{μ} has an explicit expression

$$\mathcal{R}_{\mu}(\mathbf{X}) = \sum_{i=1}^n \left(\mu - [\sqrt{\mu} - \sigma_i(\mathbf{X})]_+^2 \right). \quad (8.19)$$

Note that (8.19) is a concave function $f(s) = \mu - [\sqrt{\mu} - s]_+^2$, acting on the singular values, but with the added benefit of having the global optima coincide for this particular case. It is natural to ask, when do the global minima of (8.10), coincide with the global minimisers of

$$\min_{\mathbf{X}} \mathcal{R}_{\mu}(\mathbf{X}) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2? \quad (8.20)$$

It was shown in [9] that they do when $\|\mathcal{A}\| < 1$. Compared to the nuclear norm, this is a great improvement, but there are still some gaps to fill.

In order to handle non-convex objectives, one often turns to first order methods, such as splittings schemes and subgradient based methods. These are known to be initially fast in convergence but can be slow in the neighbourhood of a minimum [4]. This is rarely an issue with low-level imaging objectives but is becoming clear when dealing with more complex objectives, such as those encountered in rigid and non-rigid structure-from-motion. We will demonstrate this in Paper X, and show how shrinking bias manifests itself in various computer vision applications.

Typically, state-of-the-art methods in structure-from-motion rely on second order methods. These methods are known to be robust—at least empirically—but theoretical optimality guarantees are often only available for simple objectives. For interesting objectives, there is always the risk of ending up in a local minimum.

8.4 Bilinear Parameterisation

So far we have only discussed how one can encourage a matrix to have low rank by penalising the singular values. A completely different approach is to use bilinear parameterisation. Note that a rank- k matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ can be written as $\mathbf{X} = \mathbf{B}\mathbf{C}^T$, where $\mathbf{B} \in \mathbb{R}^{m \times k}$ and $\mathbf{C} \in \mathbb{R}^{n \times k}$. The original problem formulation (8.7) can now, equivalently, be transformed into

$$\min_{\mathbf{B}, \mathbf{C}} \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2, \quad (8.21)$$

which is differentiable, and thus second order methods, such as LM and VarPro, are applicable.

Note that the decomposition $\mathbf{X} = \mathbf{B}\mathbf{C}^T$ is not unique; indeed, given an invertible matrix $\mathbf{P} \in \mathbb{R}^{k \times k}$, we have $\hat{\mathbf{B}} = \mathbf{B}\mathbf{P} \in \mathbb{R}^{m \times k}$ and $\hat{\mathbf{C}} = \mathbf{C}\mathbf{P}^{-T} \in \mathbb{R}^{n \times k}$, with $\hat{\mathbf{B}}\hat{\mathbf{C}}^T = \mathbf{X}$.

Now, this formulation makes sense when the sought rank is known. But what if it is not? One approach is to bring back a type of penalty, which encourages such solutions. Since the nuclear norm heuristic has been popular in the linear setting, it is natural to consider it in the bilinear setting as well. It turns out that there is a variational form of the nuclear norm [47],

$$\|\mathbf{X}\|_* = \min_{\mathbf{X}=\mathbf{B}\mathbf{C}^T} \frac{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2}{2}, \quad (8.22)$$

hence, a bilinear soft rank equivalent to (8.10) could be formulated as

$$\min_{\mathbf{B}, \mathbf{C}} \frac{\mu}{2} \left(\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2 \right) + \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|_F^2. \quad (8.23)$$

Let us look at the differences between (8.10) and (8.23). In the linear case we have a convex, but non-differential objective. In the bilinear case, however, we have a twice differential objective, but we may have introduced additional stationary points, *e.g.* the point $(\mathbf{B}, \mathbf{C}) = (0, 0)$.

To some relief, it was shown in [2, 25], that any local minimiser (\mathbf{B}, \mathbf{C}) is globally optimal, as long as $\text{rank}(\mathbf{B}\mathbf{C}^T) < k$, where k is the number of columns. This, however, does not compensate for the shrinking bias, and, even in this setting, the nuclear norm penalty has been shown to be too weak for more complex objectives. Nevertheless, it has not stopped people from using it to achieve good performance for certain tasks, *e.g.* by reducing the parameterisation, as in [7]. While doing this, however, many of the initial theoretical optimality guarantees are lost.

The main accomplishment in the third part of this thesis (Papers VIII–XII) is the unification of non-convex penalties and bilinear parameterisation—by doing so we are leveraging the best of two worlds: the theoretical justification and guarantees of optimality from the linear framework and the convergence speed from the bilinear counterpart.

Chapter 9

Closing Words

The main contributions of this thesis are threefold. First is the extension of previous work for monocular cameras to include a binocular vision system, and the extension to minimise the geometric reprojection error, instead of a purely algebraic error, in a bundle adjustment algorithm. The latter was not well-researched, although outlined in [73], prior to Paper V. Furthermore, the radial distortion auto-calibration algorithm presented in Paper IV was a novel addition to the existing literature. This part of the thesis has focused on the tracking problem of SLAM, both locally and globally, but there are other aspects, such as mapping and practical implementation on real-time systems that are left out. These parts would be interesting to explore in the future. When having access to the mobile platform, other sensory data can be utilised and incorporated to increase performance. Perhaps one could incorporate the IMU data for this setup as well, as was done in the second part of the thesis (Papers VI and VII) concerning UAV navigation. Using an IMU would also allow us to fix the scale in metric units, and rid ourselves of the global scale ambiguity.

In the second part, we analysed another constrained motion problem, namely that of a UAV with IMU data available. Here we proposed a method surpassing the state-of-the-art in terms of speed and accuracy for indoor navigation, with partially calibrated cameras. More specifically, we proposed the first-ever minimal solver capable of simultaneous estimation of motion parameters, focal length and distortion profile while being tractable for real-time applications. While the assumption of having a ground floor available at all times worked well in the datasets used in these papers, it is easy to think of scenarios where they would fail. It would be interesting to explore other methods by relaxing this constraint.

In the final part, we built a bilinear framework, in which many famous low-rank objectives are compatible. This allowed us to achieve faster convergence speed, while still

having theoretical guarantees. In Paper VIII we discussed some situations in which global optimality can be achieved; however, with missing data, such conditions are not met. It would be interesting to analyse under which conditions global optimality can be achieved with missing data present. Furthermore, one of the benefits of using bilinear optimisation is that we have access to the bilinear factors explicitly. In *e.g.* SLAM, this is useful as the factors can be interpreted as the camera matrices and scene points, respectively. Adding constraints to the bilinear factors would improve the practical aspects of the framework; however, one would lose the optimality guarantees.

As an example consider the structure-from-motion problem without missing data, for orthographic cameras, *i.e.* the (stacked) scene points $\mathbf{X} \in \mathbb{R}^{3 \times M}$ are projected in the i :th frame as $\mathbf{x}_i = \mathbf{R}_i \mathbf{X}$. Here $\mathbf{R}_i \in \mathbb{R}^{2 \times 3}$ is the first two rows of the i :th camera rotation, thus $\mathbf{R}_i \mathbf{R}_i^T = \mathbf{I}_2$. Assuming N cameras, we may stack all rotation matrices \mathbf{R}_i into $\mathbf{R} \in \mathbb{R}^{2N \times 3}$, and the resulting objective is given by

$$\min_{\mathbf{R} \in \mathbb{R}^{2N \times 3}, \mathbf{X} \in \mathbb{R}^{3 \times M}} \|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F^2, \quad \text{subject to } \mathbf{R}_i \mathbf{R}_i^T = \mathbf{I}_2, \text{ for } i = 1, \dots, N, \quad (9.1)$$

where \mathbf{M} is the measured image points. Such, and many other interesting objectives involving constraints on the bilinear factors, would benefit the framework. Therefore, it would be of interest to analyse optimality conditions given these constraints.

Chapter 10

Overview of Papers

Paper I In this paper, we show that it is possible to retrieve the relative pose between two cameras positioned on a mobile platform under the assumption that the camera centres are located at the same height above the floor. This is done by decomposing the inter-image homographies and separating the translational and rotational components by a series of algebraic observations. Finally, the problem can be formulated as a non-convex optimisation problem in the translational components, and a method for solving this problem robustly and efficiently is proposed. The rotational component can later be obtained by rearranging the elements of the inter-image homographies leading to a minimisation problem, which, it is shown, has a closed-form solution. By using synthetically generated image sequences, the method for estimating the relative parameters is analysed. It is proven to be comparable in accuracy and robustness to the method proposed by Wadenbäck and Heyden [75], which estimates the motion parameters of the monocular case.

Authors contributions: AH came up with the problem. MVÖ outlined the theory, conducted the experiments and wrote the paper.

Paper II In this paper, we generalise the work conducted in Paper I by allowing the cameras to be positioned at different heights. This makes the algorithm more tractable for real-life situations, as it is hard to mount the cameras exactly at the same height. It is shown that much of the theory from Paper I can be reused by adding an extra parameter to the non-convex optimisation problem. Furthermore, by imposing a constraint on the motion, simulating a calibration sequence of the mobile platform, it is shown that the height can be estimated more robustly.

Authors contributions: AH came up with the problem. MVÖ outlined the theory, conducted the experiments and wrote the paper.

Paper III In this paper, several novel polynomial solvers for homographies compatible with the general planar motion model was presented. The sensitivity to noise is compared to the existing minimal solver (see Section 5.3.1) and the traditional 4-point solver (see Section 3.2), and is demonstrated to have a median error between the corresponding values of the other solvers. For real data, where the general planar motion model is only a crude approximation of the actual scene, the proposed solvers have an advantage. This has applications to autonomous driving, where the mechanical construction of vehicles often allow them to perform motions not valid in the general planar motion model. Furthermore, it is significantly faster than the existing minimal solver, and comparable in speed to the traditional 4-point solver.

Authors contributions: MVÖ came up with the idea, outlined the theory, conducted the experiments and wrote the paper. Magnus Oskarsson suggested using a sampled basis instead of a Gröbner basis to reduce the elimination template.

Paper IV In this paper, we propose polynomial solvers capable of rectifying images taken with cameras suffering from radial distortion (see Section 2.2), while enforcing the general planar motion model. Furthermore, we propose a fast minimal solver for the case when the overhead tilt is known, which has applications to aerial imagery.

Authors contributions: MVÖ came up with the idea, outlined the theory, conducted the experiments and wrote the paper.

Paper V In this paper, a specific bundle adjustment method is devised, which refines the local estimates and the scene points obtained by the method developed in the Papers I and II. The assumption of constant tilt and rigid body motion gives rise to a specific block structure of the Jacobian, and standard methods for sparse bundle adjustment are not directly applicable; however, it is shown that by nesting Schur complements, the method can take advantage of the sparsity. Furthermore, experiments show a physically correct solution, with an advantage compared to generic bundle adjustment solvers.

Authors contributions: MW outlined the monocular case in his PhD thesis, but it did not lead to a publication. MVÖ revived the idea after extending it to the binocular case, and outlined the theory and conducted the experiments. MVÖ and MW jointly wrote the paper.

Paper VI In this paper, we propose a minimal solver for indoor UAV navigation, capable of incorporating the IMU data. We treat the case of a partially calibrated setup, where the focal length is unknown. In this setting, we show that our method is significantly faster than state-of-the-art methods, while also returning superior positioning estimates.

Authors contributions: MVÖ came up with the idea, and outlined the theory, and constructed the minimal solvers. PP collected the UAV sequences with ground truth. MVÖ and PP conducted the experiments together. MVÖ, MW, and PP wrote most of the paper, with some sections contributed from KÅ and AH.

Paper VII In this paper, we once again consider UAV navigation; however, this time we propose the first-ever minimal solver for simultaneous radial distortion correction, focal length and motion parameter estimation, incorporating IMU data. This is accomplished by utilising existing filtering techniques of IMU data, allowing for accurate estimation of all angles—including the yaw angle. Since our method is capable of calibrating the distortion profile and intrinsic parameters, we have made calibration procedures obsolete. Furthermore, we perform on par or better than state-of-the-art methods that rely on pre-calibration procedures, while also being able to run in real-time.

Authors contributions: PP came up with the idea to use filtering techniques to avoid IMU drift. MVÖ outlined the theory and implemented the solvers. PP and MVÖ made the experiments together. The paper was written by MVÖ, PP, and MW with feedback from KÅ and AH.

Paper VIII In this paper, we propose using bilinear parameterisation where splitting methods are commonly used, to make objectives smooth. This allows for using second order methods, which are known to exhibit good convergence properties. We show on a variety of different pattern recognition and computer vision problems, that our method performs as good or better than the state-of-the-art.

Authors contributions: CO had the initial idea. MVÖ proved Theorem 1, and CO proved the theorems concerning low-rank optima. MVÖ conducted most of the experiments, with a VarPro implementation by CO. MVÖ and CO wrote the paper jointly.

Paper IX In this paper, we devise a fixed rank version of the bilinear method proposed in Paper VIII, and again show similar performance boosts. A toy example illustrates how one may avoid local minima by overparameterisation, and for real data, we experiment on a motion capture dataset. The results indicate a more robust performance than regular VarPro implementations when using random initialisations.

Authors contributions: The idea was developed jointly by CO and MVÖ. MVÖ extended the theory from Paper VIII to this case. The paper was written by MVÖ with feedback from CO.

Paper X In this paper, we introduce a new regulariser, fusing two popular methods. This way we are able to combine the best of two worlds by encouraging bias reduction while enforcing low-rank solutions. Furthermore, we investigate the effect of shrinking bias, and how it manifests itself differently in various computer vision applications.

Authors contributions: CO came up with the idea. MVÖ developed the theory and conducted the experiments. MVÖ and CO wrote the paper jointly.

Paper XI In this paper, we propose a bilinear framework for efficient optimisation of weighted nuclear norm objectives. Compared to standard ADMM implementations the proposed method performs superior, which is validated on several rigid and non-rigid structure-from-motion test sets.

Authors contributions: CO came up with the idea. The theory was developed by JPI, CO, and MVÖ. More specifically, CO did the initial theory and identified the space \mathcal{S} , as well as the theory for how to extend the problem to non-square matrices. MVÖ contributed with relaxing the space \mathcal{S} to the space of superstochastic matrices, which is essential to prove the main theorem. JPI did all the experiments and wrote most of the paper.

Paper XII In this paper, the regulariser from Paper X is modified to fit the bilinear framework and additional experiments confirm that using the proposed framework greatly increases the performance compared to using splitting schemes such as ADMM, for more demanding applications, such as non-rigid structure-from-motion. We also motivate this theoretically by showing a novel variational formulation proving equivalence between the bilinear and the linear objectives of the fused regulariser.

Authors contributions: The idea and theory were jointly developed by MVÖ and CO. JPI conducted the experiments, and MVÖ, CO, and JPI wrote the paper together.

All papers are reproduced with permission of their respective publishers.

References

- [1] H. Alismail, B. Browning, and S. Lucey. “Photometric Bundle Adjustment for Vision-Based SLAM”. In: *Asian Conference of Computer Vision (ACCV)*. 2016, pp. 324–341.
- [2] F. R. Bach. “Convex relaxations of structured matrix factorizations”. In: *CoRR* abs/1309.3117 (2013).
- [3] H. Bay, T. Tuytelaars, and L. Van Gool. “SURF: Speeded Up Robust Features”. In: *European Conference on Computer Vision (ECCV)*. Graz, Austria, May 2006, pp. 404–417.
- [4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Found. Trends Mach. Learn.* 3.1 (Jan. 2011), pp. 1–122.
- [5] G. Bradski. “The OpenCV Library”. In: *Dr. Dobb’s Journal of Software Tools* (2000).
- [6] D. C. Brown. “Decentering Distortion of Lenses”. In: *Photogrammetric Engineering* 32 (Mar. 1966), pp. 444–462.
- [7] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [8] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. “BRIEF: Binary Robust Independent Elementary Features”. In: *European Conference on Computer Vision (ECCV)*. Heraklion, Crete, Greece, Sept. 2010, pp. 778–792.
- [9] M. Carlsson. “On Convex Envelopes and Regularization of Non-Convex Functionals without moving Global Minima”. In: *Journal of Optimization Theory and Applications* 183 (2019), pp. 66–84.

- [10] H. J. Chien, C. C. Chuang, C. Y. Chen, and R. Klette. “When to use what feature? SIFT, SURF, ORB, or A-KAZE features for monocular visual odometry”. In: *International Conference on Image and Vision Computing New Zealand (IVCNZ)*. Nov. 2016, pp. 1–6.
- [11] G. Costante and T. A. Ciarfuglia. “LS-VO: Learning Dense Optical Subspace for Robust Visual Odometry Estimation”. In: *IEEE Robotics and Automation Letters* 3.3 (July 2018), pp. 1735–1742.
- [12] D. A. Cox, J. Little, and D. O’Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Third edition. Undergraduate Texts in Mathematics. Springer-Verlag New York, 2007.
- [13] D. A. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Second edition. Graduate Texts in Mathematics. Springer-Verlag New York, 2005.
- [14] A. Delaunoy and M. Pollefeys. “Photometric Bundle Adjustment for Dense Multi-view 3D Modeling”. In: *The IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1486–1493.
- [15] M. Demazure. *Sur deux problemes de reconstruction*. Tech. rep. RR-0882. INRIA, 1988.
- [16] Y. Ding, J. Yang, J. Ponce, and H. Kong. “An Efficient Solution to the Homography-Based Relative Pose Problem With a Common Reference Direction”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2019, pp. 1655–1664.
- [17] H. F. Durrant–Whyte. “Uncertain geometry in robotics”. In: *IEEE Trans. Robotics and Automation* 4.1 (1988), pp. 23–31.
- [18] H. Durrant–Whyte and T. Bailey. “Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms”. In: *IEEE Robotics and Automation Magazine* 2 (2006), pp. 99–110.
- [19] C. Eckart and G. Young. “The approximation of one matrix by another of lower rank”. In: *Psychometrika* 1.3 (1936), pp. 211–218.
- [20] M. Fazel. “Matrix Rank Minimization with Applications”. PhD thesis. Stanford University, 2002.
- [21] A. W. Fitzgibbon. “Simultaneous linear estimation of multiple view geometry and lens distortion”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Dec. 2001.
- [22] D. Gebre-Egziabher, G. H. Elkaim, J. D. Powell, and B. W. Parkinson. “Calibration of Strapdown Magnetometers in Magnetic Field Domain”. In: *Journal of Aerospace Engineering* 19.2 (2006), pp. 87–102.

-
- [23] A. Geiger, P. Lenz, and R. Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Providence, RI, USA, June 2012, pp. 3354–3361.
- [24] G. H. Golub and V. Pereyra. “The Differentiation of Pseudo-Inverses and Non-linear Least Squares Problems Whose Variables Separate”. In: *SIAM Journal on Numerical Analysis* 10.2 (1973), pp. 413–432.
- [25] B. D. Haeffele and R. Vidal. “Structured Low-Rank Matrix Factorization: Global Optimality, Algorithms, and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.6 (2020), pp. 1468–1482.
- [26] C. Harris and J. Pike. “3D positional integration from image sequences.” In: *Image and Vision Computing* 6.2 (1988), pp. 87–90.
- [27] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
- [28] R. I. Hartley. “In Defense of the Eight-Point Algorithm”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.6 (June 1997), pp. 580–593.
- [29] J. H. Hong and A. Fitzgibbon. “Secrets of Matrix Factorization: Approximations, Numerics, Manifold Optimization and Random Restarts”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.
- [30] J. H. Hong and C. Zach. “pOSE: Pseudo Object Space Error for Initialization-Free Bundle Adjustment”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018, pp. 1876–1885.
- [31] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.
- [32] J. P. Iglesias, C. Olsson, and M. Valtonen Örnåhag. “Accurate Optimization of Weighted Nuclear Norm for Non-Rigid Structure from Motion”. In: *European Conference on Computer Vision (ECCV)*. Aug. 2020, pp. 21–37.
- [33] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [34] V. Larsson, T. Sattler, Z. Kukulova, and M. Pollefeys. “Revisiting Radial Distortion Absolute Pose”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019, pp. 1062–1071.
- [35] K. Levenberg. “A Method for the Solution of Certain Non-Linear Problems In Least Squares”. In: *Quarterly of Applied Mathematics* 2.2 (1944), pp. 164–168.

- [36] H. Li and R. Hartley. “Five-Point Motion Estimation Made Easy”. In: *International Conference on Pattern Recognition (ICPR)*. Jan. 2006, pp. 630–633.
- [37] S. Li, F. Xue, X. Wang, Z. Yan, and H. Zha. “Sequential Adversarial Learning for Self-Supervised Deep Visual Odometry”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019, pp. 2851–2860.
- [38] T. Lindeberg. *Scale-space theory in Computer Vision*. The Kluwer international series in engineering and computer science: Robotics. Kluwer Academic, 1994.
- [39] D. G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision (IJCV)* 60.2 (Nov. 2004), pp. 91–110.
- [40] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan. “Estimation of IMU and MARG orientation using a gradient descent algorithm”. In: *IEEE International Conference on Rehabilitation Robotics*. 2011, pp. 1–7.
- [41] I. Mahon, S. Williams, O. Pizarro, and M. Johnson-Roberson. “Efficient View-Based SLAM Using Visual Loop Closures.” In: *IEEE Transactions on Robotics* 24.5 (Oct. 2008), pp. 1002–1014.
- [42] D. W. Marquardt. “An Algorithm for Least-Squares Estimation of Nonlinear Parameters”. In: *Journal of the Society for Industrial and Applied Mathematics* 11.2 (1963), pp. 431–441.
- [43] P. Miraldo, S. Saha, and S. Ramalingam. “Minimal Solvers for Mini-Loop Closures in 3D Multi-Scan Alignment”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 9691–9700.
- [44] M. Muja and D. G. Lowe. “Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration”. In: *International Conference on Computer Vision Theory and Application (VISSAPP)*. Lisboa, Portugal, 2009, pp. 331–340.
- [45] R. Mur-Artal and J. D. Tardós. “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras”. In: *IEEE Transactions on Robotics* 33.5 (Oct. 2017), pp. 1255–1262.
- [46] D. Nister. “An efficient solution to the five-point relative pose problem”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.6 (June 2004), pp. 756–770.
- [47] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *SIAM Rev.* 52.3 (Aug. 2010), pp. 471–501.
- [48] V. Renaudin, M. H. Afzal, and G. Lachapelle. “Complete Triaxis Magnetometer Calibration in the Magnetic Domain”. In: *Journal of Sensors* (2010).

-
- [49] E. Rosten and T. Drummond. “Machine Learning for High-Speed Corner Detection”. In: *European Conference on Computer Vision (ECCV)*. Graz, Austria, May 2006, pp. 430–443.
- [50] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. “ORB: An efficient alternative to SIFT or SURF”. In: *International Conference on Computer Vision (ICCV)*. Barcelona, Spain, Nov. 2011, pp. 2564–2571.
- [51] A. Ruhe and P. A. Wedin. “Algorithms for Separable Nonlinear Least Squares Problems”. In: *SIAM Review* 22.3 (1980), pp. 318–337.
- [52] D. Santana-Cedr s, L. G mez, M. Alem n-Flores, A. Salgado, J. Esclar n, L. Mazorra, and L.  lvarez. “An Iterative Optimization Algorithm for Lens Distortion Correction Using Two-Parameter Models”. In: *Image Processing On Line* 6 (2016), pp. 326–364.
- [53] T. Sattler, A. Torii, J. Sivic, M. Pollefeys, H. Taira, M. Okutomi, and T. Pajdla. “Are Large-Scale 3D Models Really Necessary for Accurate Visual Localization?”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 6175–6184.
- [54] T. Schops, V. Larsson, M. Pollefeys, and T. Sattler. “Why Having 10,000 Parameters in Your Camera Model Is Better Than Twelve”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020, pp. 2535–2544.
- [55] J. Sherman and W. J. Morrison. “Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix”. In: *The Annals of Mathematical Statistics* 21.1 (1950), pp. 124–127.
- [56] R. C. Smith and P. Cheeseman. “On the representation of spatial uncertainty”. In: *International Journal of Robotics Research* 5.4 (1987), pp. 56–68.
- [57] Z. Tang, R. Grompone von Gioi, P. Monasse, and J. Morel. “A Precision Analysis of Camera Distortion Models”. In: *IEEE Transactions on Image Processing* 26.6 (2017), pp. 2694–2704.
- [58] P. H. S. Torr, A. Zisserman, and S. J. Maybank. “Robust Detection of Degenerate Configurations while Estimating the Fundamental Matrix”. In: *Computer Vision and Image Understanding* 71.3 (1998), pp. 312–333.
- [59] M. Valtonen  rnhag. “Fast Non-minimal Solvers for Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 40–51.

- [60] M. Valtonen Örnthag. “Radially Distorted Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Valletta, Malta, Feb. 2020, pp. 568–575.
- [61] M. Valtonen Örnthag and A. Heyden. “Generalisation of Parameter Recovery in Binocular Vision for a Planar Scene”. In: *International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI)*. Montréal, Canada, May 2018, pp. 37–42.
- [62] M. Valtonen Örnthag and A. Heyden. “Generalization of Parameter Recovery in Binocular Vision for a Planar Scene”. In: *International Journal of Pattern Recognition and Artificial Intelligence* 33.11 (2019).
- [63] M. Valtonen Örnthag and A. Heyden. “Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Funchal, Madeira, Portugal, Jan. 2018, pp. 568–575.
- [64] M. Valtonen Örnthag, J. P. Iglesias, and C. Olsson. “Bilinear Parameterization for Non-Separable Singular Value Penalties”. Submitted. To appear in the *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021.
- [65] M. Valtonen Örnthag and C. Olsson. “A Unified Optimization Framework for Low-Rank Inducing Penalties”. In: *In the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020, pp. 8474–8493.
- [66] M. Valtonen Örnthag, C. Olsson, and A. Heyden. “Bilinear Parameterization for Differentiable Rank-Regularization”. In: *In the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2020.
- [67] M. Valtonen Örnthag, C. Olsson, and A. Heyden. “Differentiable Fixed-Rank Regularisation using Bilinear Parameterisation”. In: *In the British Machine Vision Conference (BMVC)*. 2019.
- [68] M. Valtonen Örnthag, P. Persson, M. Wadenbäck, K. Åström, and A. Heyden. “Efficient Real-Time Radial Distortion Correction for UAVs”. In: *The IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2021, pp. 1751–1760.
- [69] M. Valtonen Örnthag, P. Persson, M. Wadenbäck, K. Åström, and A. Heyden. “Minimal Solvers for Indoor UAV Positioning”. In: *International Conference on Pattern Recognition (ICPR)* (2020), pp. 1136–1143.

-
- [70] M. Valtonen Örnthag and M. Wadenbäck. “Enforcing the General Planar Motion Model: Bundle Adjustment for Planar Scenes”. In: *Pattern Recognition Applications and Methods. ICPRAM 2019. Lecture Notes in Computer Science*. Vol. 11996. Springer International Publishing, 2020, pp. 119–135.
- [71] M. Valtonen Örnthag and M. Wadenbäck. “Planar Motion Bundle Adjustment”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 24–31.
- [72] M. Valtonen Örnthag. “Efficient Radial Distortion Correction for Planar Motion”. In: *Pattern Recognition Applications and Methods. ICPRAM 2020. Lecture Notes in Computer Science*. Springer International Publishing, 2021, pp. 46–63.
- [73] M. Wadenbäck. “Homography-Based Positioning and Planar Motion Recovery”. PhD thesis. Lund University, 2017.
- [74] M. Wadenbäck, K. Åström, and A. Heyden. “Recovering Planar Motion from Homographies Obtained using a 2.5-Point Solver for a Polynomial System”. In: *International Conference on Image Processing (ICIP)*. Phoenix, AZ, USA, Sept. 2016, pp. 2966–2970.
- [75] M. Wadenbäck and A. Heyden. “Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Lisbon, Portugal, Jan. 2014, pp. 635–639.
- [76] M. Wadenbäck and A. Heyden. “Planar Motion and Hand-Eye Calibration using Inter-Image Homographies from a Planar Scene”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Barcelona, Spain, 2013, pp. 164–168.
- [77] M. Wadenbäck, M. Karlsson, A. Heyden, A. Robertsson, and R. Johansson. “Visual Odometry from Two Point Correspondences and Initial Automatic Tilt Calibration”. In: *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*. Porto, Portugal, Feb. 2017, pp. 340–346.
- [78] S. Wang, R. Clark, H. Wen, and N. Trigoni. “DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks”. In: *International Conference on Robotics and Automation (ICRA)*. May 2017, pp. 2043–2050.
- [79] O. J. Woodford and E. Rosten. “Large Scale Photometric Bundle Adjustment”. In: *Proceedings of the British Machine Vision Conference (BMVC)*. 2020.

- [80] Z. Zhang. “A flexible new technique for camera calibration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (Nov. 2000), pp. 1330–1334.

Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies

MARCUS VALTONEN ÖRNHAG AND ANDERS HEYDEN
Centre for Mathematical Sciences, Lund University

Abstract: In this paper we consider a mobile platform with two cameras directed towards the floor mounted the same distance from the ground, assuming planar motion and constant internal parameters. Earlier work related to this specific problem geometry has been carried out for monocular systems, and the main contribution of this paper is the generalisation to a binocular system and the recovery of the relative translation and orientation between the cameras. The method is based on previous work on monocular systems, using sequences of inter-image homographies. Experiments are conducted using synthetic data, and the results demonstrate a robust method for determining the relative parameters.

1 Introduction

In robotics research, it is of interest to accurately track the position of a mobile robot relative to its surroundings. The emergence of artificial intelligence and autonomous vehicles in recent years demand robust algorithms to handle such problems. During the years of research in the field many kinds of sensors have been used—LIDAR, rotary encoders, inertial sensors and GPS, to mention a few—and often in combination. The type of sensor one chooses to work with restricts what algorithms that can be used, and how the resulting map of the robot and its surroundings will look.

One sensor of particular interest for the robotics and computer vision community is the image sensor and there are many reasons why it is popular. One important factor is that the wide range of algorithms used in computer vision, *e.g.* visual feature extraction and pose estimation, can be used in this setting; however, from an industrial point of view image sensors are an often considered design choice since they are relatively cheap compared to other sensors. Furthermore, they are often available on consumer products, such as smartphones and tablets, where similar techniques can be used, *e.g.* in Augmented Reality (AR). With image sensors one is not limited to sparse 3D clouds of feature points, but can model the map using dense and textured 3D models.

Visual SLAM systems have been developed for nearly three decades, with [5] being one of the first. Since then, several improvements have been made, and with the

aid of modern computing power, a variety of methods for real-time SLAM are available. Among the more recent ones are MonoSLAM [1], LSD-SLAM [2] and ORB-SLAM2 [10], where the latter includes support for monocular, stereo and RGB-D cameras.

2 Related Work

In epipolar geometry, the fundamental matrix, introduced by [3] and [6], has been a tool for many algorithms concerning scene reconstruction; however, planar motion is known to be ill-conditioned, see *e.g.* [7]. The problem geometry considered in this paper is forced to planar motion, which is common in *e.g.* indoor environments. To overcome this issue algorithms that take advantage of planar homographies have been devised, which by construction are constrained to planar motion and therefore do not suffer from being ill-conditioned. Some early work on planar motion using homographies include that of [8] and [4]. More recent work on ego-motion recovery in a monocular system using inter-image homographies for a planar scene has been covered in [13] for a single homography and by the same authors for several homographies in [12]. In [14] the same methods are used to calibrate the fixed parameters initially, transforming the subsequent problem to a two-dimensional rigid body motion problem.

The stereo rig problem involving two cameras with fixed relative orientation is investigated for auto-calibration in [7]. In [11] a method for multi-camera platform calibration using multi-linear constraints is developed; however, this method does not rely on the inter-image homographies, but rather using the camera matrices.

3 Theory

3.1 Problem Geometry

In this paper we consider a mobile platform with two cameras directed towards the floor mounted the same distance from the ground. By a suitable choice of the world coordinate system the cameras move in the plane $z = 0$ and relative to the ground plane positioned at $z = 1$. Both cameras are assumed to be mounted rigidly onto the platform and no common scene point is assumed to be visible in the cameras simultaneously. Furthermore, the mobile platform's centre of rotation is assumed to be located in the first camera centre. In this setting the second camera centre is connected to the first by a rigid body motion.

The 3D rotations are parametrised using Tait–Bryan angles

$$\mathbf{R}(\psi, \theta, \varphi) = \mathbf{R}_x(\psi)\mathbf{R}_y(\theta)\mathbf{R}_z(\varphi), \quad (1)$$

where \mathbf{R}_x , \mathbf{R}_y and \mathbf{R}_z denote the rotation around the respective coordinate axes with a given angle. The problem geometry is illustrated in Figure 1.

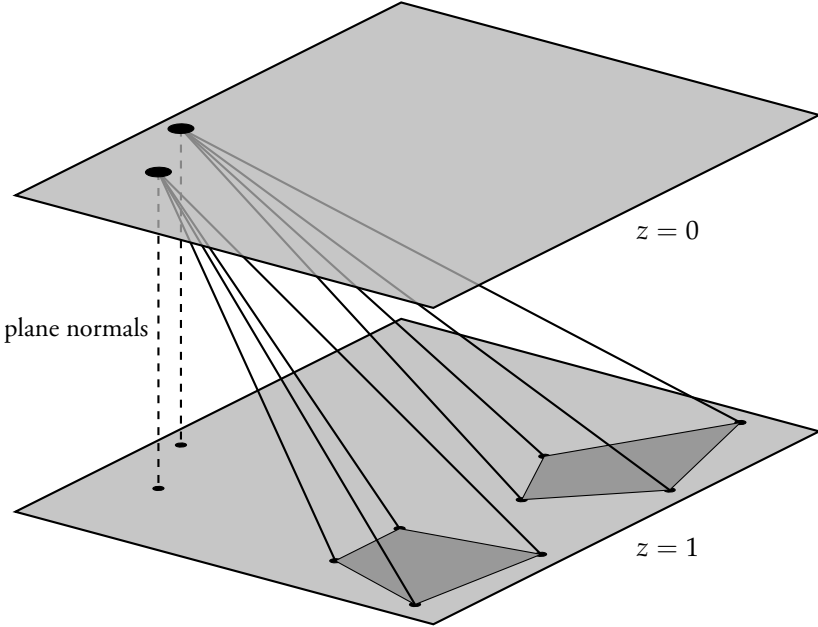


Figure 1: The problem geometry considered in this paper. The cameras are assumed to move in the plane $z = 0$ and the relative orientation between them as well as the tilt towards the floor normal is assumed to be fixed as the mobile platform moves freely.

3.2 Camera Parametrisation

As in [13], consider two consecutive images, A and B , for the first camera. The camera matrices are then

$$\begin{aligned} P_A &= \mathbf{R}_{\psi\theta}[\mathbf{I} \mid \mathbf{0}], \\ P_B &= \mathbf{R}_{\psi\theta}\mathbf{R}_\varphi[\mathbf{I} \mid -\mathbf{t}], \end{aligned} \quad (2)$$

where $\mathbf{R}_{\psi\theta}$ is a rotation θ around the y -axis followed by a rotation of ψ around the x -axis. The movement of the mobile platform is modelled by a rotation φ around the z -axis, corresponding to \mathbf{R}_φ and translation vector \mathbf{t} .

The camera matrices for the second camera can be parametrised as

$$\begin{aligned} P'_A &= \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau [I \mid \mathbf{0}], \\ P'_B &= \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau \mathbf{R}_\varphi [I \mid -\mathbf{t}], \end{aligned} \quad (3)$$

where $\mathbf{R}_{\psi'\theta'}$ is the tilt, defined as for the first camera. Furthermore, \mathbf{R}_η is a fixed rotation by η degrees around the z-axis relative to the first camera, and $\boldsymbol{\tau}$ is the rigid body translation vector between the first and the second camera centre. The matrix \mathbf{T}_τ corresponds to a translation by $\boldsymbol{\tau}$ defined as $\mathbf{T}_\tau = \mathbf{I} - \boldsymbol{\tau} \mathbf{n}^T$, where $\mathbf{n} = (0, 0, 1)^T$ is a floor normal.

3.3 Homographies

Given point correspondences \mathbf{x}_1 and \mathbf{x}_2 , in homogeneous coordinates, the homography \mathbf{H} transforms the points such that $\mathbf{x}_2 = \lambda \mathbf{H} \mathbf{x}_1$, where $\lambda \neq 0$ is due to universal scale ambiguity. In [13] the homography for the first camera is derived and is given by

$$\lambda \mathbf{H} = \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi \mathbf{T}_t \mathbf{R}_{\psi\theta}^T. \quad (4)$$

Similarly, the homography \mathbf{H}' for the second camera is given by

$$\lambda' \mathbf{H}' = \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau \mathbf{R}_\varphi \mathbf{T}_t \mathbf{T}_\tau^{-1} \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T. \quad (5)$$

3.4 Parameter Recovery

By separating the fixed angles from φ and the translation \mathbf{t} the following relation holds

$$\mathbf{R}_\varphi \mathbf{T}_t = \lambda \mathbf{R}_{\psi\theta}^T \mathbf{H} \mathbf{R}_{\psi\theta} = \lambda' \mathbf{T}_\tau^{-1} \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau, \quad (6)$$

It is shown in [13] how to recover the parameters for the monocular case, and by doing so the parameters ψ , θ , φ and \mathbf{t} as well as ψ' and θ' can be recovered; the latter two from treating the second camera as a monocular system. Furthermore, we shall assume that all homographies \mathbf{H} are normalised such that $\det \mathbf{H} = 1$.

Recovering the Relative Translation

The relative translation and rotation can be separated by putting (6) in the form

$$\mathbf{T}_\tau \mathbf{R}_\varphi \mathbf{T}_t \mathbf{T}_\tau^{-1} = \lambda' \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta, \quad (7)$$

and multiplying with the transpose from the left on both sides yield

$$\mathbf{T}_{t-\tau}^T \mathbf{R}_\varphi^T \mathbf{T}_\tau^T \mathbf{T}_\tau \mathbf{R}_\varphi \mathbf{T}_{t-\tau} = \lambda' \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}'^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta. \quad (8)$$

The left hand side of (8) can be simplified to

$$\mathcal{L} = \begin{bmatrix} 1 & 0 & \ell_1 \\ 0 & 1 & \ell_2 \\ \ell_1 & \ell_2 & \ell_3 \end{bmatrix}, \quad (9)$$

where

$$\begin{aligned} \ell_1 &= \tau_x - t_x - \tau_y \sin \varphi - \tau_x \cos \varphi, \\ \ell_2 &= \tau_y - t_y + \tau_x \sin \varphi - \tau_y \cos \varphi, \\ \ell_3 &= k_1 \tau_x + k_2 \tau_y + c \tau_x^2 + c \tau_y^2 + |\mathbf{t}|^2 + 1, \end{aligned} \quad (10)$$

and

$$\begin{aligned} k_1 &= 2(t_x \cos \varphi - t_y \sin \varphi - t_x), \\ k_2 &= 2(t_x \sin \varphi + t_y \cos \varphi - t_y), \\ c &= 2(1 - \cos \varphi). \end{aligned} \quad (11)$$

The eigenvalues of \mathcal{L} from (9) are given by $\lambda_2 = 1$ and λ_1, λ_3 such that $\lambda_1 \lambda_3 = \ell_3 - \ell_1^2 - \ell_2^2 = 1$. Furthermore, the right hand side of (8) has the same eigenvalues as $\mathbf{H}'^T \mathbf{H}'$, as they are similar. Since the sum of the eigenvalues is the trace of the corresponding matrix, the following relation holds

$$\text{tr } \mathbf{H}'^T \mathbf{H}' = 2 + \ell_3, \quad (12)$$

which is independent of η . By letting $h = \text{tr } \mathbf{H}'^T \mathbf{H}' - 3 - |\mathbf{t}|^2$ the relation becomes

$$k_1 \tau_x + k_2 \tau_y + c \tau_x^2 + c \tau_y^2 - h = 0. \quad (13)$$

The other invariants do not give any new relations for $\boldsymbol{\tau}$ since, $\det \mathcal{L} = 1$ and $\frac{1}{2} \left((\text{tr } \mathcal{L})^2 - \text{tr } \mathcal{L}^2 \right) = \text{tr } \mathcal{L}$.

Solving for the Relative Translation

With only one pair of homographies one cannot determine $\boldsymbol{\tau}$ explicitly; however, using multiple pairs one equation on the form (13) for each pair of homography is given, which yields a system of equations

$$\begin{aligned} k_1^{(1)} \tau_x + k_2^{(1)} \tau_y + c^{(1)} (\tau_x^2 + \tau_y^2) - h^{(1)} &= 0, \\ k_1^{(2)} \tau_x + k_2^{(2)} \tau_y + c^{(2)} (\tau_x^2 + \tau_y^2) - h^{(2)} &= 0, \\ &\vdots \\ k_1^{(n)} \tau_x + k_2^{(n)} \tau_y + c^{(n)} (\tau_x^2 + \tau_y^2) - h^{(n)} &= 0. \end{aligned} \quad (14)$$

The system in (14) is over-determined for $n > 2$, hence minimising

$$\min_{\boldsymbol{\tau} \in \mathbb{R}^2} \sum_{i=1}^n \left| k_1^{(i)} \tau_x + k_2^{(i)} \tau_y + c^{(i)} (\tau_x^2 + \tau_y^2) - h^{(i)} \right|^2, \quad (15)$$

gives the desired result. This can be re-formulated as

$$\min_{\boldsymbol{\tau} \in \mathbb{R}^2} \|\mathbf{K}\boldsymbol{\tau} + \mathbf{c}\boldsymbol{\tau}^T \boldsymbol{\tau} - \mathbf{h}\|_2^2, \quad (16)$$

where

$$\mathbf{K} = \begin{bmatrix} k_1^{(1)} & k_2^{(1)} \\ k_1^{(2)} & k_2^{(2)} \\ \vdots & \vdots \\ k_1^{(n)} & k_2^{(n)} \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} c^{(1)} \\ c^{(2)} \\ \vdots \\ c^{(n)} \end{bmatrix} \quad \text{and} \quad \mathbf{h} = \begin{bmatrix} h^{(1)} \\ h^{(2)} \\ \vdots \\ h^{(n)} \end{bmatrix}. \quad (17)$$

By introducing a new variable $r = |\boldsymbol{\tau}|^2$, an equivalent problem is obtained

$$\min_{\boldsymbol{\tau} \in \mathbb{R}^2, r=|\boldsymbol{\tau}|^2} \|\mathbf{K}\boldsymbol{\tau} + \mathbf{c}r - \mathbf{h}\|_2^2 = \min_{\mathbf{x} \in \mathbb{R}^3, r=|\boldsymbol{\tau}|^2} \|\mathbf{M}\mathbf{x} - \mathbf{h}\|_2^2, \quad (18)$$

where $\mathbf{x} = (\tau_x, \tau_y, r)^T$ and $\mathbf{M} = [\mathbf{K} \mid \mathbf{c}]$, where the objective function can be written as

$$\|\mathbf{M}\mathbf{x} - \mathbf{h}\|_2^2 = \mathbf{x}^T \mathbf{Q}\mathbf{x} + \mathbf{d}^T \mathbf{x} + \mathbf{h}^T \mathbf{h}, \quad (19)$$

where $\mathbf{Q} = \mathbf{M}^T \mathbf{M}$ and $\mathbf{d}^T = -2\mathbf{h}^T \mathbf{M}$. In conclusion, one may consider minimising $f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q}\mathbf{x} + \mathbf{d}^T \mathbf{x}$, subject to $x_1^2 + x_2^2 - x_3 = 0$. Note that, the constraint can be written as $\mathbf{x}^T \mathbf{A}\mathbf{x} + \mathbf{b}^T \mathbf{x} = 0$, where

$$\mathbf{A} = \begin{bmatrix} 1 & & \\ & 1 & \\ & & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}. \quad (20)$$

The Lagrangian is given by

$$\mathcal{L}(\mathbf{x}; \lambda) = \mathbf{x}^T \mathbf{Q}\mathbf{x} + \mathbf{d}^T \mathbf{x} + \lambda(\mathbf{x}^T \mathbf{A}\mathbf{x} + \mathbf{b}^T \mathbf{x}), \quad (21)$$

and solving $\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}; \lambda) = \mathbf{0}$ results in

$$\mathbf{x} = -\frac{1}{2}(\mathbf{Q} + \lambda \mathbf{A})^{-1}(\mathbf{d} + \lambda \mathbf{b}). \quad (22)$$

The constraint $\nabla_{\lambda} \mathcal{L}(\mathbf{x}; \lambda) = 0$ yields a rational equation in λ , which can be turned into finding the roots of a fifth degree polynomial. This in turn can be translated into an eigenvalue problem, and solved robustly. Using this approach solving (16) takes $\sim 100 \mu\text{s}$ which is suitable for real-time applications. Furthermore, due to the precision of modern eigenvalue solvers, the error is usually negligible.

Solving for the Relative Orientation

Given τ from the previous step consider (6) and multiply the first homography with the matrix corresponding to the translation τ . This yields

$$T_\tau R_{\psi\theta}^T H R_{\psi\theta} T_\tau^{-1} \sim R_\eta^T R_{\psi'\theta'}^T H' R_{\psi'\theta'} R_\eta, \quad (23)$$

where η is the only unknown parameter. Define $W = T_\tau R_{\psi\theta}^T H R_{\psi\theta} T_\tau^{-1}$ and $W' = R_{\psi'\theta'}^T H' R_{\psi'\theta'}$ and note that W and W' share the same eigenvalues since they are similar and the corresponding eigenvectors are rotated η degrees.

Let us recall that the null space of a matrix is spanned by the right-singular vectors corresponding to zero—or due to noise, vanishing—singular values. Using the same approach as in [12] we conclude that $\text{nulldim } W'^T W' = 1$. Consequently, the eigenvectors spanning the null spaces, $x \in \mathcal{N}(W^T W)$ and $x' \in \mathcal{N}(W'^T W')$, can be obtained using SVD—this ensures that we work with real eigenvectors.

We will use the following theorem to recover η robustly, using all available pairs of homographies.

Theorem 1. *Let $Y, Y' \in \mathbb{R}^{3 \times N}$ and non-zero. Furthermore, let $R_\eta = R_z(\eta)$ be a rotation matrix, corresponding to a rotation of angle η around the third axis. Then*

$$\min_{\substack{\eta \in (-\pi, \pi] \\ \lambda \neq 0}} \|Y' - \lambda R_\eta Y\|_F, \quad (24)$$

is solved when

$$\eta_{opt} = \alpha + \begin{cases} 0, & \text{if } \mathbf{y}_3^T \mathbf{y}'_3 > 0, \\ \pi, & \text{otherwise,} \end{cases} \quad (25)$$

where α may be expressed using the programming friendly `atan2` function,

$$\alpha = \text{atan2}\left(\mathbf{y}_1^T \mathbf{y}'_2 - \mathbf{y}_2^T \mathbf{y}'_1, \mathbf{y}_1^T \mathbf{y}'_1 + \mathbf{y}_2^T \mathbf{y}'_2\right). \quad (26)$$

Here \mathbf{y}_i denotes the column vector of dimension N corresponding to the i :th row of Y . The vectors \mathbf{y}'_i are defined analogously. The angles are considered as equivalence classes, where $\eta \equiv \eta + 2\pi k$, $k \in \mathbb{Z}$, with the class representative being in the interval $(-\pi, \pi]$.

Proof. Using the relation between the Frobenius norm and the trace, the square of the objective function can be simplified

$$\begin{aligned} \|Y' - \lambda R_\eta Y\|_F^2 &= \text{tr}\left[(Y' - \lambda R_\eta Y)(Y' - \lambda R_\eta Y)^T\right] \\ &= \text{tr } Y' Y'^T - \lambda \text{tr } Y' Y^T R_\eta^T \\ &\quad - \lambda \text{tr } R_\eta Y Y'^T + \lambda^2 \text{tr } R_\eta Y Y^T R_\eta^T. \end{aligned} \quad (27)$$

Since the trace is invariant under cyclic permutations it follows that the last term is independent of η . Furthermore,

$$\text{tr } \mathbf{Y}' \mathbf{Y}^T \mathbf{R}_\eta^T = \text{tr} \left(\left(\mathbf{Y}' \mathbf{Y}^T \mathbf{R}_\eta^T \right)^T \right) = \text{tr } \mathbf{R}_\eta \mathbf{Y} \mathbf{Y}'^T. \quad (28)$$

Combining these observations (24) is equivalent to solving

$$\min_{\substack{\eta \in (-\pi, \pi] \\ \lambda \neq 0}} \lambda^2 \|\mathbf{Y}\|_F^2 - 2\lambda \text{tr } \mathbf{R}_\eta \mathbf{Y} \mathbf{Y}'^T. \quad (29)$$

The reader can easily verify that the optimum is reached when η is on the form (25). \square

In conclusion, the angle η may be obtained using Theorem 1 where the i :th column of \mathbf{Y} corresponds to the eigenvector spanning the null space of $\mathbf{W}_i^T \mathbf{W}_i$ —the matrix \mathbf{Y}' is defined analogously.

4 Experiments

4.1 Synthetic Data

In order to validate the theory and evaluate the algorithm synthetic data was generated in form of sequences of images mimicking those taken by a mobile platform as described in Section 3.1. A high-resolution image of a planar scene, in this case a textured floor, was chosen to yield many key-points. Furthermore, different paths simulating the mobile platform was defined. In order to simulate the tilt the original image was transformed around a given point along the predefined path and then cropped, such that the centre point in the cropped image coincide with this point. The parameters used in the transformation serve as ground truth, and the resolution used in each image is 400×400 pixels. The field of view of the simulated camera is normalised to 90 degrees, which affects the impact of the distortion of the images caused by the cameras being tilted.

4.2 Homography Estimation

The homography estimation was done by extracting SIFT keypoints [9] from every frame, keeping the most prominent once as candidates for key-point matching. The remaining key-points are then matched between subsequent images only, using a brute-force matcher based on the K Nearest Neighbour algorithm. From the matched key-points a random subset is chosen iteratively in the RANSAC framework and from these

a homography is estimated. The homography with the highest amount of inliers is chosen, where the maximum allowed reprojection error for a point pair to be considered as an inlier is five pixels.

4.3 Parameter Recovery

The parameters were recovered using the method proposed in [12] for both trajectories, treated as two independent monocular systems, using five homographies to determine the tilt, rotation and translation in each step. The optimal relative translation vector was obtained by solving (16) using the optimisation scheme proposed in Section 3.4.

Using the closed form solution presented in Theorem 1 the relative rotation around the z-axis was estimated for the five pairs of homographies used in the previous step. The computations involves finding the vectors spanning the null spaces in order to compute the matrices used in the closed form expression (24) for η which is computationally inexpensive.

4.4 Test Cases

Elliptic path

This case simulates the mobile platform moving in an elliptic path while rotating between the images. The test case was chosen as it includes general motions which generates many different combinations of values for the nonfixed parameters. The sequence of images for both cameras used in this case is shown in Figure 2. The parameters used in this experiment are $\psi = 3.3^\circ$, $\theta = -1.2^\circ$, $\psi' = 5.1^\circ$, $\theta' = -4.6^\circ$, $\tau = (100, 80)$ and $\eta = 30^\circ$.

The results from analysing the two paths independently are shown in Figure 3 and Figure 4. The estimation of the relative pose is shown in Figure 5.

Rotation around the First Camera Centre

Estimating the tilt in a monocular system with only rotations and no translation is generally hard. A possible benefit of a binocular system is that the rigid body motion between the cameras results in a translational component in the second camera. The generated paths are shown in Figure 6. All fixed angles are set to 0° and the relative translation $\tau = (800, 0)$. Furthermore, the mobile platform is moving with a constant rotation.

Since (8) degenerates, as $k_1^{(i)} = k_2^{(i)} = 0$ for all homographies, one cannot expect to recover the components of τ but it is possible to recover $|\tau|$, as shown in Figure 7.

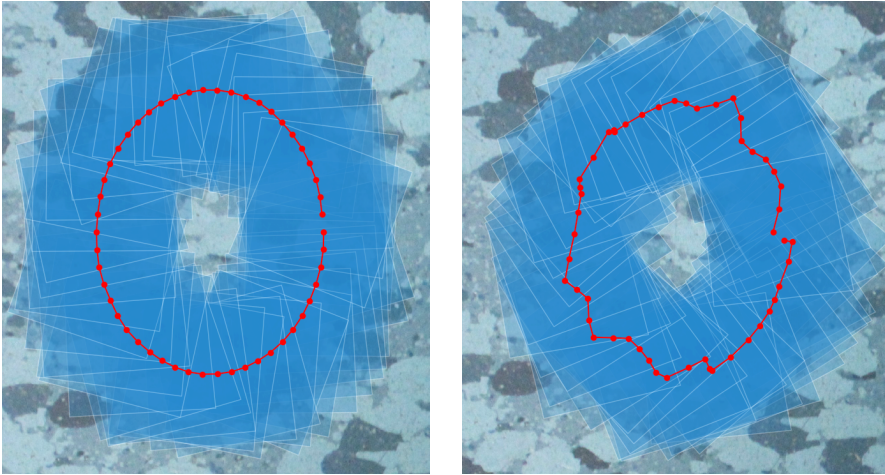


Figure 2: Path of the simulated mobile platform for the first camera (left) and the second camera (right). The red dots represent the absolute position of the camera and the blue squares are the extracted images. The impact of the tilt is illustrated by the frames not being square, but rather slanted. Note that the second camera path is not elliptic as the translational components are affected by the rotation of the mobile platform.

Mean Error vs. Number of Homographies

The relation between the accuracy and the amount of homographies used to estimate the relative pose is shown in Figure 8. The same setup as in Section 4.4 was used but the amount of homographies varied. From the figures one can see that it is not a significant improvement in the parameter estimation of the relative pose after approximately twenty homographies. In practice this means that the calibration could be done initially, and then be used to track the position of the mobile platform, without re-computation of the fixed parameters.

5 Conclusion

This paper has extended the work of [12] to binocular vision. A method has been devised to robustly estimate the relative translation and orientation of the two cameras using several pairs of homographies, by reusing the computations from the cameras treated as two monocular systems. The translational component is recovered by solving a non-convex problem, which can be turned into an eigenvalue problem. The proposed optimisation

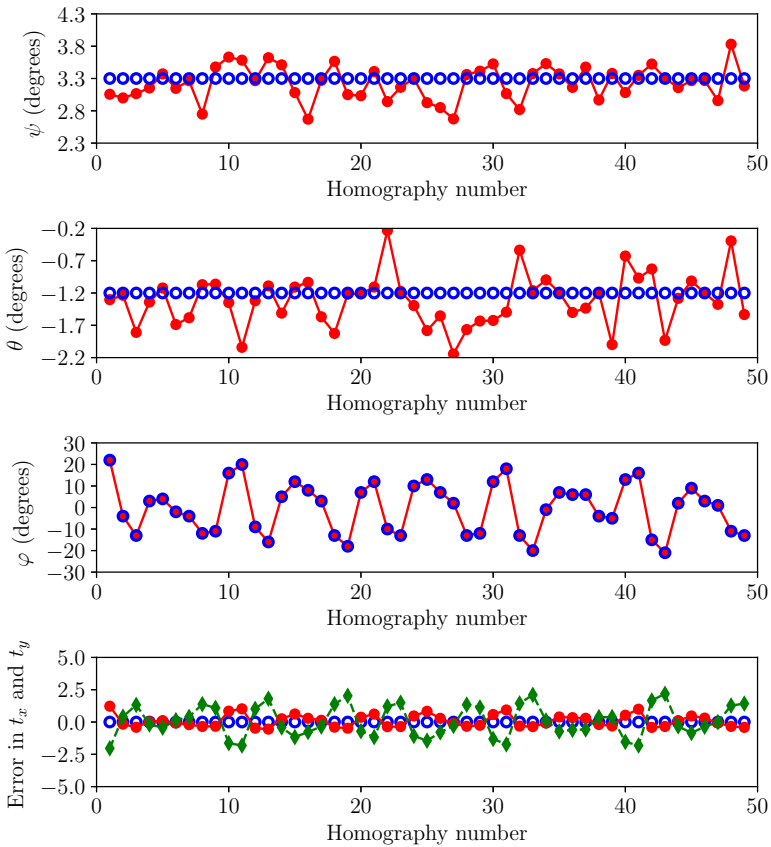


Figure 3: Estimated parameters for the first camera. In all plots the blue circles represent the ground truth. The red dots are the estimated parameters for the angles (first three subplots) and in the last subplot the red dots and the green diamond are the error in t_x and t_y respectively. The error in translation is measured in pixels. The estimates have been calculated using five homographies at each frame.

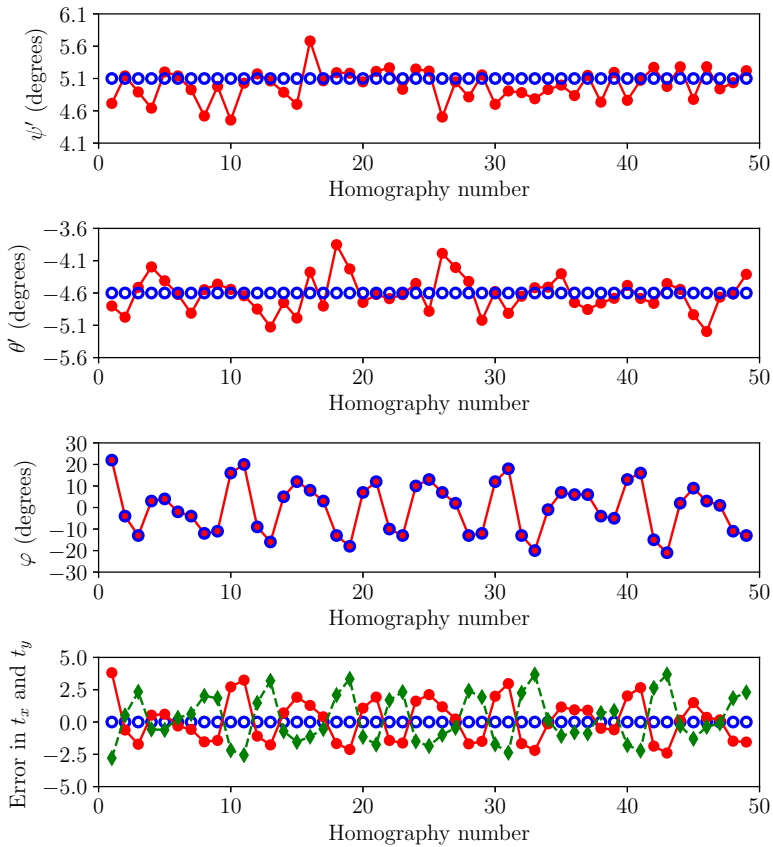


Figure 4: Estimated parameters for the second camera. In all plots the blue circles represent the ground truth. The red dots are the estimated parameters for the angles (first three subplots) and in the last subplot the red dots and the green diamond are the error in t_x and t_y respectively. The error in translation is measured in pixels. The estimates have been calculated using five homographies at each frame.

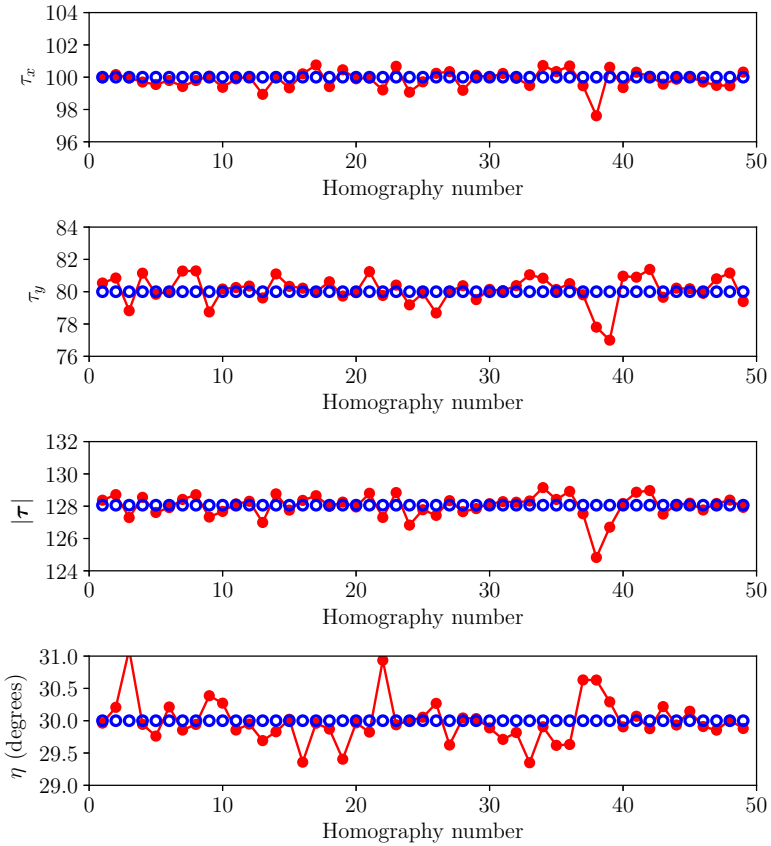


Figure 5: Estimated value of $\tau = (\tau_x, \tau_y)$ and η using five pairs of homographies at each frame. The magnitude of the translational component $|\tau|$ is also shown. The red dots are the estimated parameter and the blue circles represent the ground truth.

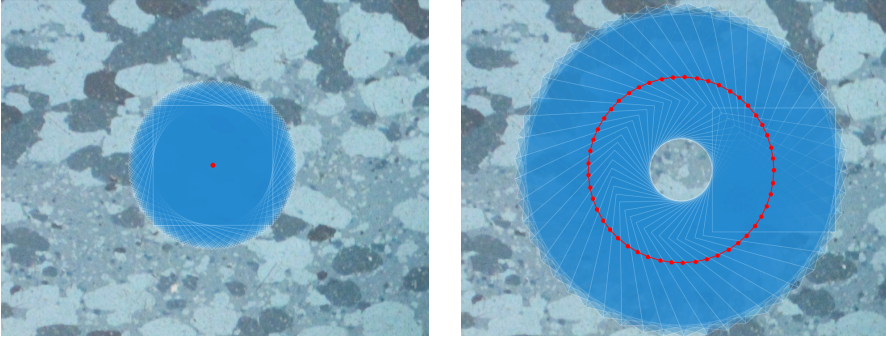


Figure 6: Path of the simulated mobile platform for the first camera (left) in the second test case, and for the second camera (right). The red dots represent the absolute position of the camera and the blue squares are the extracted images. Due to the rigid body motion connecting the cameras the second camera rotates around the first camera centre causing a non-zero translational component.

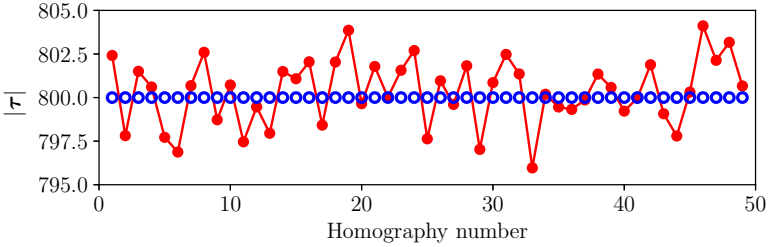


Figure 7: Estimated value of $|\tau|$ using five pairs of homographies at each frame. When considering only rotations for the first camera, the components of τ cannot be obtained by the proposed method. The red dots are the estimated parameter for the magnitude and the blue circles represent the ground truth.

scheme is robust and suitable for real-time applications. Furthermore, when solving for the relative rotation, the closed form solution presented in Theorem 1 is computationally inexpensive. Experimental results from synthetic data have demonstrated that the method has an acceptable accuracy for most problems, and highlighted problems where the method fails to recover both of the translational components; it is also shown that in this case the magnitude can be recovered accurately.

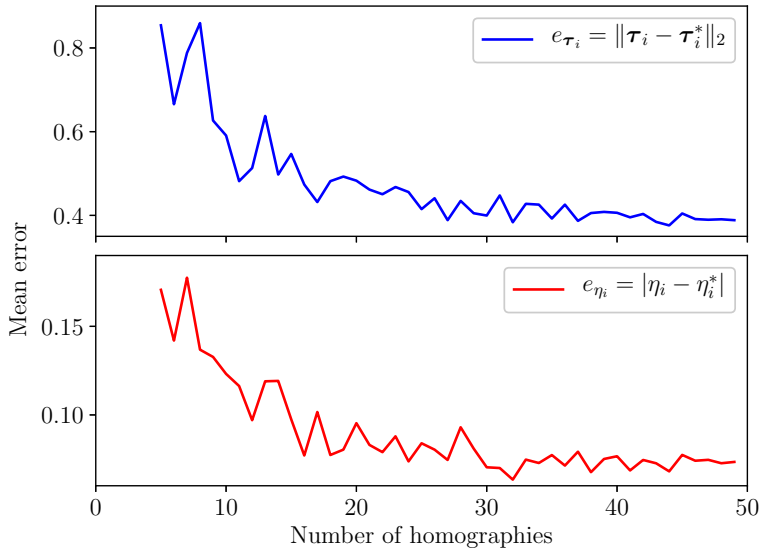


Figure 8: Mean error for the relative translation and rotation as a function of the number of pairs of homographies used in the optimisation step. The error for the translational component is measured in the Euclidean norm. The mean error is computed from 49 pairs of homographies estimated from the sequence described in Section 4.4.

Acknowledgements

This work has been funded by the Swedish Research Council through grant no. 2015-05639 ‘Visual SLAM based on Planar Homographies’.

References

- [1] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. “MonoSLAM: Real-Time Single Camera SLAM”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 29.6 (June 2007), pp. 1052–1067.
- [2] J. Engel, T. Schöps, and D. Cremers. “LSD-SLAM: Large-Scale Direct Monocular SLAM”. In: *European Conference on Computer Vision (ECCV)*. Zurich, Switzerland, Sept. 2014, pp. 834–849.
- [3] O. D. Faugeras. “What Can Be Seen in Three Dimensions with an Uncalibrated Stereo Rig”. In: *European Conference on Computer Vision (ECCV)*. Santa Margherita Ligure, Italy, May 1992, pp. 563–578.
- [4] H. Hajjdiab and R. Laganière. “Vision-based Multi-Robot Simultaneous Localization and Mapping”. In: *Canadian Conference on Computer and Robot Vision (CRV)*. London, ON, Canada, May 2004, pp. 155–162.
- [5] C. Harris and J. Pike. “3D positional integration from image sequences.” In: *Image and Vision Computing* 6.2 (1988), pp. 87–90.
- [6] R. I. Hartley. “Estimation of relative camera positions for uncalibrated cameras”. In: *European Conference on Computer Vision (ECCV)*. Santa Margherita Ligure, Italy, May 1992, pp. 579–587.
- [7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
- [8] B. Liang and N. Pears. “Visual Navigation using Planar Homographies”. In: *International Conference on Robotics and Automation (ICRA)*. Washington, DC, USA, May 2002, pp. 205–210.
- [9] D. G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision (IJCV)* 60.2 (Nov. 2004), pp. 91–110.
- [10] R. Mur-Artal and J. D. Tardós. “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras”. In: *IEEE Transactions on Robotics* 33.5 (Oct. 2017), pp. 1255–1262.
- [11] P. Nyman, A. Heyden, and K. Åström. “Multi-camera Platform Calibration Using Multi-linear Constraints”. In: *International Conference on Pattern Recognition (ICPR)* (2010), p. 53.
- [12] M. Wadenbäck and A. Heyden. “Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Lisbon, Portugal, Jan. 2014, pp. 635–639.

- [13] M. Wadenbäck and A. Heyden. “Planar Motion and Hand-Eye Calibration using Inter-Image Homographies from a Planar Scene”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Barcelona, Spain, 2013, pp. 164–168.
- [14] M. Wadenbäck, M. Karlsson, A. Heyden, A. Robertsson, and R. Johansson. “Visual Odometry from Two Point Correspondences and Initial Automatic Tilt Calibration”. In: *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*. Porto, Portugal, Feb. 2017, pp. 340–346.

Appendix: Proof of Theorem 1

In this appendix we give a complete proof of Theorem 1, which was left out due to space limitations.

Proof of Theorem 1 (cont.). Consider the equivalent formulation (29). The second term can be simplified due to the following observation

$$\mathbf{R}_\eta \mathbf{Y} \mathbf{Y}'^T = \begin{bmatrix} \cos \eta & -\sin \eta & 0 \\ \sin \eta & \cos \eta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sum_i y_{1i} y'_{i1} & \sum_i y_{1i} y'_{i2} & \sum_i y_{1i} y'_{i3} \\ \sum_i y_{2i} y'_{i1} & \sum_i y_{2i} y'_{i2} & \sum_i y_{2i} y'_{i3} \\ \sum_i y_{3i} y'_{i1} & \sum_i y_{3i} y'_{i2} & \sum_i y_{3i} y'_{i3} \end{bmatrix}, \quad (30)$$

and therefore

$$\begin{aligned} \text{tr } \mathbf{R}_\eta \mathbf{Y} \mathbf{Y}'^T &= \left(\sum_i y_{1i} y'_{i1} + y_{2i} y'_{i2} \right) \cos \eta + \left(\sum_i y_{2i} y'_{i1} - y_{1i} y'_{i2} \right) \sin \eta + \sum_i y_{3i} y'_{i3} \\ &= A (\cos \alpha \cos \eta + \sin \alpha \sin \eta) + \gamma = A \cos(\eta - \alpha) + \gamma, \end{aligned} \quad (31)$$

where A and α are obtained by the cosine subtraction formula and γ is the constant term. Using (31) consider the objective function

$$f(\eta, \lambda) = k\lambda^2 - 2\lambda[A \cos(\eta - \alpha) + \gamma], \quad (32)$$

where $k = \|\mathbf{Y}\|_F^2 > 0$, since $\mathbf{Y} \neq \mathbf{0}$. The partial derivatives are

$$\begin{aligned} \frac{\partial f}{\partial \eta} &= 2\lambda A \sin(\eta - \alpha), \\ \frac{\partial f}{\partial \lambda} &= 2k\lambda - 2[A \cos(\eta - \alpha) + \gamma], \end{aligned} \quad (33)$$

and since $\lambda \neq 0$ it follows that $\sin(\eta - \alpha) = 0$, at a critical point, *i.e.* $\eta = \alpha + \pi n$, $n \in \mathcal{Z}$. Furthermore,

$$k\lambda = A \cos(\eta - \alpha) + \gamma = A(-1)^n + \gamma. \quad (34)$$

The determinant of the Hessian \mathbf{H} , evaluated at the critical points, are given by $\det \mathbf{H}(\alpha + \pi n, \hat{\lambda}) = 4k\hat{\lambda}(-1)^n$. Thus, a local minimum can only occur when $\hat{\lambda} > 0$ and $n = 0$ or $\hat{\lambda} < 0$ and $n = 1$, as we consider the angles as equivalence classes, with the representative in the interval $(-\pi, \pi]$. It follows that the objective function has two potential minimiser,

$$f\left(\alpha, \frac{\gamma + A}{k}\right) = -\frac{(\gamma + A)^2}{k}, \quad (35)$$

and

$$f\left(\alpha + \pi, \frac{\gamma - A}{k}\right) = -\frac{(\gamma - A)^2}{k}. \quad (36)$$

We now derive the constraint necessary for deducing whether $\eta = \alpha$ or $\eta = \alpha + \pi$ is the correct solution for the angle

$$f\left(\alpha, \frac{\gamma + A}{k}\right) < f\left(\alpha + \pi, \frac{\gamma - A}{k}\right) \Leftrightarrow (\gamma + A)^2 > (\gamma - A)^2 \Leftrightarrow \gamma > 0, \quad (37)$$

where we use that $k > 0$ and $A > 0$. In conclusion,

$$\eta_{\text{opt}} = \alpha + \begin{cases} 0, & \text{if } \sum_i y_{3i}y'_{i3} > 0, \\ \pi, & \text{otherwise,} \end{cases}$$

where α may be expressed using the programming friendly `atan2` function,

$$\alpha = \text{atan2}\left(\sum_i y_{1i}y'_{i2} - y_{2i}y'_{i1}, \sum_i y_{1i}y'_{i1} + y_{2i}y'_{i2}\right),$$

and the desired form may be obtained by vectorisation. □

Generalisation of Parameter Recover in Binocular Vision for a Planar Scene

MARCUS VALTONEN ÖRNHAG AND ANDERS HEYDEN
Centre for Mathematical Sciences, Lund University

Abstract: In this paper we consider a mobile platform with two cameras directed towards the floor. In earlier work this specific problem geometry has been considered under the assumption that the cameras have been mounted at the same height. This paper extends the previous work by removing the height constraint, as it is hard to realize in real-life applications.

We develop a method based on an equivalent problem geometry, and show that much of previous work can be re-used with small modification to account for the height difference. A fast solver for the resulting non-convex optimization problem is devised. Furthermore, we propose a second method for estimating the height difference by constraining the mobile platform to pure translations. This is intended to simulate a calibration sequence, which is not uncommon to impose. Experiments are conducted using synthetic data, and the results demonstrate a robust method for determining the relative parameters comparable to previous work.

1 Introduction

The past decades Simultaneous Localization and Mapping (SLAM) has been studied by the robotics community and the computer vision community alike. The importance of correctly mapping and navigating through unknown terrain is crucial as we enter the age of autonomous vehicles, which is why SLAM is still an active field of research. Furthermore, as the cost for well-performing image sensors is low compared to other alternatives they are an understandable design choice for many applications.

Taking the known properties of the physical environment into account when constructing a SLAM system can increase performance in all stages—from navigation and localization to consistent map building—and many SLAM systems have been developed in different domains, ranging from outdoor environments, to underwater and airborne systems. Modern SLAM systems can, due to the increasing computational power available on consumer products and carefully developed algorithms, handle all of the stages in real-time. Not only monocular systems are available, but a variety of different setups including stereo and RGB-D cameras are supported by the latest frameworks.

In this paper we will continue to investigate methods suitable for planar motion, a problem which is typical for indoor environments. A common approach to determine the correlation between scene points is to estimate the fundamental matrix [1, 4]. This approach fails for planar motion, as it is known to be ill-conditioned, see *e.g.* [5], and other algorithms have been devised to take the intended environment into account. One way of doing so is by using inter-image homographies, which was proposed by Liang and Pears [6] and Hajjdiab and Laganière [3] among others.

2 Related Work

Recent work on ego-motion recovery has been conducted by Wadenbäck and Heyden [10] for a monocular system using inter-image homographies for a planar scene. The work was later generalized to use more than one homography in each frame in order to make the estimations more accurate and robust [9]. In [12] the authors demonstrate that the parameter recovery can be used as an initial stage in order to transform the problem into a two-dimensional rigid body motion problem, thus being able to track the motion by point correspondences alone, by first correcting for the fixed parameters.

It was shown in Valtonen Örnthag and Heyden [8] how to recover the parameters from a binocular system by extending the methods developed by Wadenbäck and Heyden [9]; however, the method assumed the camera centers to be positioned at the same height above the ground floor, which is hard to achieve in real-life applications.

In [8] the procedure for recovering all parameters, *i.e.* the tilt angles, the motion of the mobile platform, as well as the relative translation and orientation between the cameras, can be summarized by the following: *Step 1:* Compute the translation and rotation for the mobile platform using images from the first camera. The tilt angles are obtained for both cameras by treating each camera trajectory as being monocular. *Step 2:* The relative translation is estimated by minimizing the error of a function relating the homographies from the second camera to the motion of the mobile platform. *Step 3:* The relative rotation is estimated using the homographies and the estimated fixed angles as well as the relative translation.

3 Theory

3.1 Problem Geometry

Consider a mobile platform with two cameras directed towards the floor. By a suitable choice of the world coordinate system the cameras move in the plane $z = a$ and $z = b$ respectively relative to the ground plane positioned at $z = 0$. Both cameras are assumed

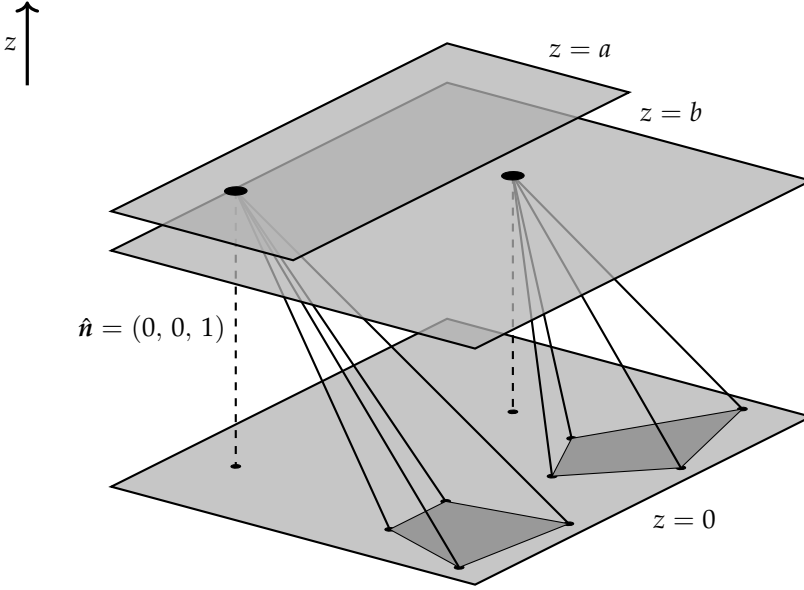


Figure 1: The problem geometry considered in this paper. The cameras are assumed to move in the planes $z = a$ and $z = b$ the relative orientation between them as well as the tilt towards the floor normal is assumed to be fixed as the mobile platform moves freely.

to be mounted rigidly onto the platform and no common scene point is assumed to be visible in the cameras simultaneously. In this setting the second camera center is connected to the first by a rigid body motion.

It is convenient to work with the camera centers at $z = 0$ and therefore, we consider the equivalent problem where both camera centers are positioned at $z = 0$, but with different ground planes, as is illustrated in Figure 1 and Figure 2.

These two models are equivalent; however, in order to re-use results from previous work one may consider the relative translation $\boldsymbol{\tau} = (\tau_x, \tau_y, \tau_z)$ decomposed into two components $\boldsymbol{\tau}_{xy} = (\tau_x, \tau_y, 0)$ and $\boldsymbol{\tau}_z = (0, 0, \tau_z)$. By doing so, $\boldsymbol{\tau}_{xy}$ from the physical problem geometry is identical to $\boldsymbol{\tau}$ in the theoretical model with two ground planes. The z -component from the original problem, τ_z , corresponds to the difference between the image planes in the theoretical model. Without loss of generality, one may assume that the origin is located in the first camera center.

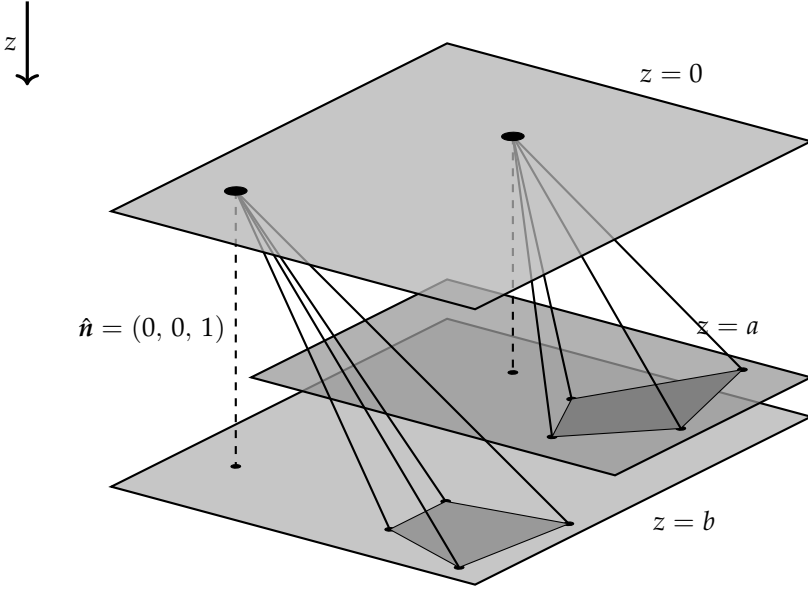


Figure 2: An equivalent problem is to assume that the cameras move in the same plane but the image planes are different. This is, of course, only of theoretical value, but simplifies some of the relations.

3.2 The Impact of Different Image Planes

In general, a translation by $\mathbf{t} = (t_x, t_y, 0)^T$ in a plane $z = z_0$ is given by $(x, y, z_0) \mapsto (x - t_x, y - t_y, z_0)$, which in homogeneous coordinates correspond to

$$\hat{\mathbf{X}} \mapsto [\mathbf{I} \mid -\mathbf{t}]\mathbf{X} = \left(\mathbf{I} - \frac{\mathbf{t}\mathbf{n}^T}{z_0} \right) \hat{\mathbf{X}}, \quad (1)$$

where $\mathbf{X} = (x, y, z_0, 1)$, $\hat{\mathbf{X}} = (x, y, z_0)$ is the corresponding image point and $\mathbf{n} = (0, 0, 1)^T$ is a floor normal. We introduce the notation $\mathbf{T}_{\mathbf{t}}(z_0) = \mathbf{I} - \mathbf{t}\mathbf{n}^T/z_0$, which is justified by (1). For convenience, let $\mathbf{T}_{\mathbf{t}}(1) = \mathbf{T}_{\mathbf{t}}$, thereby leading to the relation $\mathbf{T}_{\mathbf{t}}(b) = \mathbf{T}_{\mathbf{t}/b}$.

The difference, compared to the simplified case where $a = b = 1$, is that the translations occur in different image planes. This physical property can be incorporated in the translation matrices $\mathbf{T}_{\mathbf{r}}$ and $\mathbf{T}_{\mathbf{t}}$ by a scaling factor equal to the distance to the image plane. The monocular case is not affected by introducing different image planes more

than up to scale; however, this issue is always present due to global scale ambiguity. The method proposed in [8] for estimating the relative orientation can be re-used with the modified translation matrix without changing the theoretical aspects or the performance. Therefore, the major concern in the generalized problem is to estimate the position of the different image planes relative to each other.

3.3 Camera Matrices

It was shown in [10] that the camera matrices for the first camera, assuming two consecutive images A and B , can be parametrised as

$$\begin{aligned} P_A &= \mathbf{R}_{\psi\theta}[\mathbf{I} \mid \mathbf{0}], \\ P_B &= \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi[\mathbf{I} \mid -\mathbf{t}], \end{aligned} \quad (2)$$

where $\mathbf{R}_{\psi\theta}$ is a rotation θ around the y -axis followed by a rotation of ψ around the x -axis. The translational component of the mobile platform is denoted \mathbf{t} and the rotational component by φ , which is assumed to rotate about the z -axis. The corresponding rotation matrix is denoted by \mathbf{R}_φ . In [8] the camera matrices for the second camera were derived. By accounting for the different image planes they are given by

$$\begin{aligned} P'_A &= \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta T_\tau(b)[\mathbf{I} \mid \mathbf{0}], \\ P'_B &= \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta T_\tau(b) \mathbf{R}_\varphi[\mathbf{I} \mid -\mathbf{t}], \end{aligned} \quad (3)$$

where ψ' and θ' are the tilt angles defined as in the first case, τ is the relative translation between the camera centers and η is the fixed rotation about the z -axis relative to the first camera.

3.4 Homographies

The homographies between two consecutive images are derived as in Valtonen Örnag and Heyden [8], taking the impact of different image planes into account, giving

$$\begin{aligned} \mathbf{H} &\sim \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi T_t(a) \mathbf{R}_{\psi\theta}^T, \\ \mathbf{H}' &\sim \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta T_\tau(b) \mathbf{R}_\varphi T_t(b) T_\tau^{-1}(b) \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T. \end{aligned} \quad (4)$$

The matrices representing the homographies can be made unique by imposing $\det \mathbf{H} = \det \mathbf{H}' = 1$, which will be assumed throughout the paper.

4 Parameter Recovery

By separating the fixed angles from φ and the translation \mathbf{t} in (4) the following relation holds

$$\begin{aligned} \mathbf{R}_\varphi \mathbf{T}_t(a) &= \mathbf{R}_{\psi\theta}^T \mathbf{H} \mathbf{R}_{\psi\theta}, \\ \mathbf{R}_\varphi \mathbf{T}_t(b) &= \mathbf{T}_\tau^{-1}(b) \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b). \end{aligned} \quad (5)$$

4.1 Pure Translation

Impose the constraint $\mathbf{R}_\varphi = \mathbf{I}$, *i.e.* the mobile platform does not rotate. Due to global scale ambiguity one may choose $a = 1$. This leaves the translation vector $\mathbf{t} = (t_x, t_y)$, the fixed parameters and the additional scale parameter b to be estimated. Under these assumptions (5) is simplified to,

$$\begin{aligned} \mathbf{T}_t &= \mathbf{R}_{\psi\theta}^T \mathbf{H} \mathbf{R}_{\psi\theta} \\ \mathbf{T}_t(b) &= \mathbf{T}_\tau^{-1}(b) \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta \mathbf{T}_\tau(b) \end{aligned} \quad (6)$$

Note that in this model the translation vector \mathbf{t} can be estimated from the first homography. Following the same method used by Valtonen Örnhaug and Heyden [8], when treating the case $b = 1$, one may separate the relative translation vector from the relative orientation, which in this case cancels out, due to commutativity of translation matrices

$$\mathbf{T}_t(b) = \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta. \quad (7)$$

Since the eigenvalues of $\mathbf{T}_{t/b}$ do not depend on \mathbf{t} nor b one may multiply both sides with the transpose from the left, which yields

$$\mathbf{T}_t^T(b) \mathbf{T}_t(b) = \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}'^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta. \quad (8)$$

By interpreting the right-hand side of (8) as a similarity transformation, it follows that the eigenvalues of the right-hand side are those of $\mathbf{H}'^T \mathbf{H}'$. In [11] it was shown that the eigenvalues for the general form $\mathbf{T}_s^T \mathbf{T}_s$ are given by

$$\lambda_2 = 1, \quad \lambda_{1,3} = 1 + \frac{|\mathbf{s}|^2}{2} \pm \frac{|\mathbf{s}|}{2} \sqrt{|\mathbf{s}|^2 + 4}, \quad (9)$$

where all eigenvalues are positive and $\lambda_1 \geq 1$, and $\lambda_3 \leq 1$ fulfilling the relation $\lambda_1/\lambda_3 = \lambda_1^2$. Using the same approach as in [8] and the relation $\mathbf{T}_t(b) = \mathbf{T}_{t/b}$, the sum of the eigenvalues gives an equation for b

$$3 + \left| \frac{\mathbf{t}}{b} \right|^2 = \text{tr} \mathbf{H}'^T \mathbf{H}'. \quad (10)$$

For convenience, introduce $\xi = 1/b^2$, then $\xi > 0$. Furthermore, when $b = 1$ it follows that $\xi = 1$, which gives the same equation as when the cameras are mounted at the same height. For a single pair of homographies one gets

$$r_i \xi - h_i = 0, \quad (11)$$

where $r_i = |\mathbf{t}_i|^2$ and $h_i = \text{tr } \mathbf{H}_i'^T \mathbf{H}_i' - 3$. Due to noise one may consider minimizing

$$\min_{\xi \geq 0} \sum_{i=1}^N |r_i \xi - h_i|^2, \quad (12)$$

for N pairs of homographies. Vectorizing the objective function, this can be written as

$$f(\xi) = \|\mathbf{r}\xi - \mathbf{h}\|_2^2 = \xi^2 \mathbf{r}^T \mathbf{r} - 2\xi \mathbf{r}^T \mathbf{h} + \mathbf{h}^T \mathbf{h}, \quad (13)$$

giving an optimal value $\xi^* = (\mathbf{h}^T \mathbf{r}) / (\mathbf{r}^T \mathbf{r})$, which in turn gives an estimate for b , namely

$$b = \sqrt{\frac{\mathbf{r}^T \mathbf{r}}{\mathbf{h}^T \mathbf{h}}}. \quad (14)$$

The practical implications of this result is that one may recover b when no rotations are present, which implies that the mobile platform can be calibrated along a straight path before allowing more general motion. In practice, it is not uncommon to impose a calibration sequence on a mobile platform; however, in the following section we will devise a method that recovers the parameters for general planar motion.

4.2 General Planar Motion

Consider (5) without the constraint $\mathbf{R}_\varphi = \mathbf{I}$. Again, \mathbf{t} and also φ , can be recovered from the monocular case and the second equation can be re-written as

$$\mathbf{T}_{\tau/b} \mathbf{R}_\varphi \mathbf{T}_{t/b} \mathbf{T}_{\tau/b}^{-1} = \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta, \quad (15)$$

where we use the relation $\mathbf{T}_t(b) = \mathbf{T}_{t/b}$ and similarly for τ . Multiplying with the transpose from the left yields

$$\mathbf{T}_{t/b-\tau/b}^T \mathbf{R}_\varphi^T \mathbf{T}_{\tau/b}^T \mathbf{T}_{\tau/b} \mathbf{R}_\varphi \mathbf{T}_{t/b-\tau/b} = \mathbf{R}_\eta^T \mathbf{R}_{\psi'\theta'}^T \mathbf{H}'^T \mathbf{H}' \mathbf{R}_{\psi'\theta'} \mathbf{R}_\eta. \quad (16)$$

It is shown in Valtonen Örnhaug and Heyden [8] that the left-hand side LHS of (16) can be simplified to

$$\text{LHS} = \begin{bmatrix} 1 & 0 & \ell_1 \\ 0 & 1 & \ell_2 \\ \ell_1 & \ell_2 & \ell_3 \end{bmatrix}, \quad (17)$$

which still holds true, with a modification of the values ℓ_1 , ℓ_2 and ℓ_3 . It is also shown that the eigenvalues of LHS are given by $\lambda_2 = 1$ and λ_1, λ_3 such that $\lambda_1\lambda_3 = \ell_3 - \ell_1^2 - \ell_2^2 = 1$. Introducing $\xi = 1/b^2$ for convenience, it follows that

$$\ell_3 = k_1\tilde{\xi}\tau_x + k_2\tilde{\xi}\tau_y + c\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}|\mathbf{t}|^2 + 1, \quad (18)$$

where

$$\begin{aligned} k_1 &= 2(t_x \cos \varphi - t_y \sin \varphi - t_x), \\ k_2 &= 2(t_x \sin \varphi + t_y \cos \varphi - t_y), \\ c &= 2(1 - \cos \varphi). \end{aligned} \quad (19)$$

Furthermore, the right hand side of (16) has the same eigenvalues as $\mathbf{H}'^T \mathbf{H}'$, as they are similar. Since the sum of the eigenvalues is the trace of the corresponding matrix, the following relation holds

$$\text{tr } \mathbf{H}'^T \mathbf{H}' = 2 + \ell_3, \quad (20)$$

which is independent of η . By letting $h = \text{tr } \mathbf{H}'^T \mathbf{H}' - 3$ the relation becomes

$$k_1\tilde{\xi}\tau_x + k_2\tilde{\xi}\tau_y + c\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}|\mathbf{t}|^2 - h = 0. \quad (21)$$

4.3 Solving for the Relative Translation and Unknown Scale

Using multiple pairs of homographies the problem can be formulated as

$$\begin{aligned} k_1^{(1)}\tilde{\xi}\tau_x + k_2^{(1)}\tilde{\xi}\tau_y + c^{(1)}\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}|\mathbf{t}^{(1)}|^2 - h^{(1)} &= 0, \\ k_1^{(2)}\tilde{\xi}\tau_x + k_2^{(2)}\tilde{\xi}\tau_y + c^{(2)}\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}|\mathbf{t}^{(2)}|^2 - h^{(2)} &= 0, \\ &\vdots \\ k_1^{(N)}\tilde{\xi}\tau_x + k_2^{(N)}\tilde{\xi}\tau_y + c^{(N)}\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}|\mathbf{t}^{(N)}|^2 - h^{(N)} &= 0. \end{aligned} \quad (22)$$

The system (22) is over-determined for $N > 3$, hence minimising

$$\min_{\substack{\boldsymbol{\tau} \in \mathbb{R}^2 \\ \tilde{\xi} \in \mathbb{R}^+}} \sum_{i=1}^N \left| k_1^{(i)}\tilde{\xi}\tau_x + k_2^{(i)}\tilde{\xi}\tau_y + c^{(i)}\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}|\mathbf{t}^{(i)}|^2 - h^{(i)} \right|^2, \quad (23)$$

reduces the impact of noise. The minimization problem (23) can be re-formulated as

$$\min_{\substack{\boldsymbol{\tau} \in \mathbb{R}^2 \\ \tilde{\xi} \in \mathbb{R}^+}} \|\mathbf{K}\tilde{\xi}\boldsymbol{\tau} + c\tilde{\xi}|\boldsymbol{\tau}|^2 + \tilde{\xi}\mathbf{r} - \mathbf{h}\|_2^2, \quad (24)$$

where $\mathbf{K} \in \mathbb{R}^{N \times 2}$, $\mathbf{c} \in \mathbb{R}^N$, $\mathbf{r} \in \mathbb{R}^N$ and $\mathbf{h} \in \mathbb{R}^N$. The vector \mathbf{r} consists of the elements $r_i = |\mathbf{t}_i|^2$, and the other follow the naming convention used in Valtonen Örnhaug and Heyden [8]. Introducing $\mathbf{x} = (\zeta\tau_x, \zeta\tau_y, \zeta|\boldsymbol{\tau}|^2, \zeta)^T$ the problem can be reformulated as

$$\min_{x_1^2+x_2^2=x_3x_4} \|\mathbf{M}\mathbf{x} - \mathbf{h}\|_2^2, \quad (25)$$

where $\mathbf{M} = [\mathbf{K} \mid \mathbf{c} \mid \mathbf{r}]$. The constraint can be written $\mathbf{x}^T \mathbf{A} \mathbf{x} = 0$ where

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1/2 \\ 0 & 0 & -1/2 & 0 \end{bmatrix}. \quad (26)$$

The Lagrangian is given by

$$\mathcal{L}(\mathbf{x}; \lambda) = \mathbf{x}^T \mathbf{Q} \mathbf{x} - 2\mathbf{d}^T \mathbf{x} + \lambda \mathbf{x}^T \mathbf{A} \mathbf{x}, \quad (27)$$

where $\mathbf{Q} = \mathbf{M}^T \mathbf{M}$ and $\mathbf{d} = \mathbf{M}^T \mathbf{h}$. A necessary condition for a minimizer is given by $\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}; \lambda) = \mathbf{0}$, which yields

$$\mathbf{x} = (\mathbf{Q} + \lambda \mathbf{A})^{-1} \mathbf{d}. \quad (28)$$

We will denote $\mathbf{Q}_\lambda = \mathbf{Q} + \lambda \mathbf{A}$ in the remaining part of this section. Inserting (28) in the equation obtained by $\nabla_{\lambda} \mathcal{L}(\mathbf{x}; \lambda) = 0$ yields

$$\mathbf{d}^T \mathbf{Q}_\lambda^{-T} \mathbf{A} \mathbf{Q}_\lambda^{-1} \mathbf{d} = 0, \quad (29)$$

which is a rational equation in λ . Assuming $\det \mathbf{Q}_\lambda \neq 0$ this can be turned into a sixth degree polynomial equation

$$\mathbf{d}^T \text{adj} \mathbf{Q}_\lambda^T \mathbf{A} \text{adj} \mathbf{Q}_\lambda \mathbf{d} = 0, \quad (30)$$

where $\text{adj} \mathbf{A}$ denotes the adjoint matrix of \mathbf{A} . The degree of the polynomial equation is due to the adjoint matrix containing cubic terms. This can in turn be transformed into an eigenvalue problem, and solved robustly as the error is negligible in the case of a general 6×6 matrix.

5 Experiments

The experimental setup follows the one presented in [8], by using synthetic data. Firstly, images simulating those taken by a mobile platform was extracted from a high-resolution

image, by transforming image segments into 400×400 pixels. The original image is of a highly textured floor, chosen to yield many keypoints. Secondly, homographies were estimated from these sequences of images estimated by extracting SIFT keypoints and matching them between subsequent images. The best homography was estimated using RANSAC, where, in order to be considered an inlier, the reprojection error for a point pair was set to five pixels.

The field of view of the simulated cameras are normalized to 90 degrees, and the non-fixed parameters for the mobile platform were estimated from the first camera using the method proposed in [9].

The simulated path of the mobile platform starts by translating along the y -direction. This part of the trajectory is intended as a calibration sequence for estimating the parameter b , as explained in Section 4.1, and consists of ten images—the remaining 30 images are generated from general planar motion, as shown in Figure 3.

5.1 Initial Calibration vs. General Motion

This test case highlights the differences between the two proposed methods, *i.e.* as an initial calibration sequence consisting of pure translation or allowing general motion. By only considering the first ten images, containing pure translation, the distance to the ground floor was estimated.

Using the same setup, but considering only the last 30 images including general planar motion the distance to the ground floor was estimated. In both cases five pairs of homographies were used in each frame. The result is shown in Figure 4.

If one restricts the movements of the mobile platform by imposing an initial calibration sequence the estimated height generally gives a better result than allowing general motion. The methods are both suitable for real-time application, and the average speed, computed on a normal laptop, was $700 \mu\text{s}$ for the general case and even faster using the initial calibration sequence.

5.2 Error vs. Noise

Homographies on the form (4) were generated with $a = 1$, and b randomly chosen, as well as the other parameters. Normal distributed noise was added to the parameters t , φ and h after generating the homographies. The angle was distorted by 0.01ε (in radians) and the others ε , where $\varepsilon \in \mathcal{N}(0, \sigma)$. The error was measured in the weighted norm

$$e = \sqrt{\frac{|\tau^* - \tau|^2}{|\tau|^2} + \frac{(b^* - b)^2}{b^2}}, \quad (31)$$

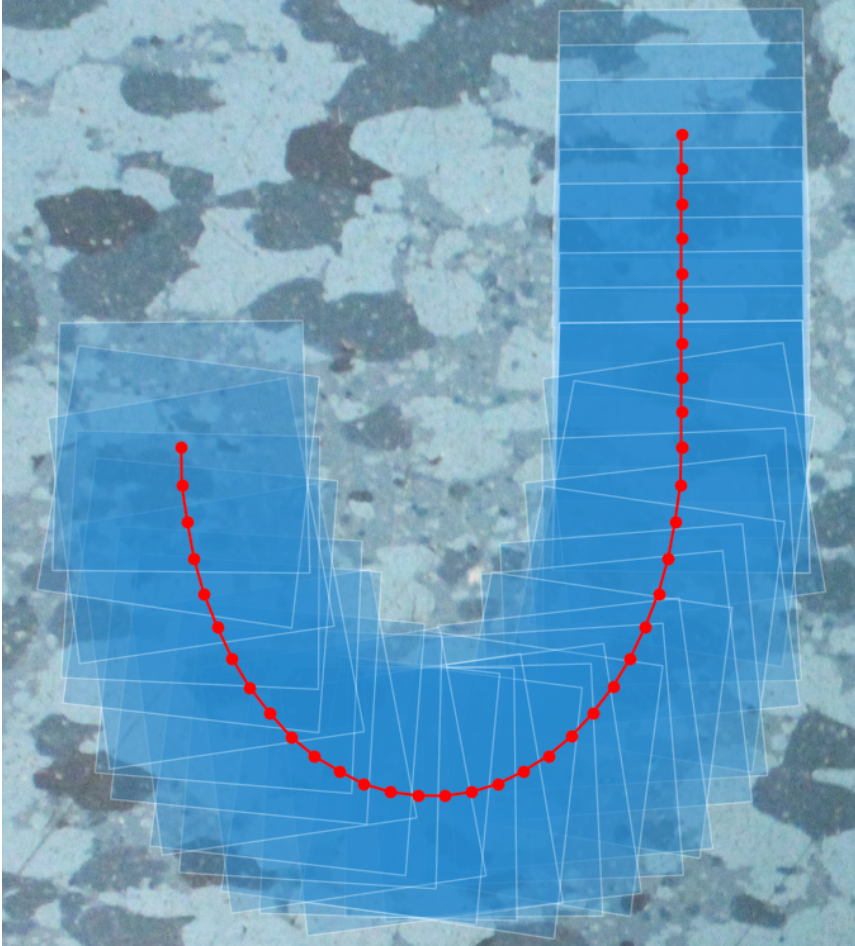


Figure 3: Synthetically generated images cropped from a high-resolution image. The sequence consists of ten images with pure translation initially, which may be used to calibrate the height.

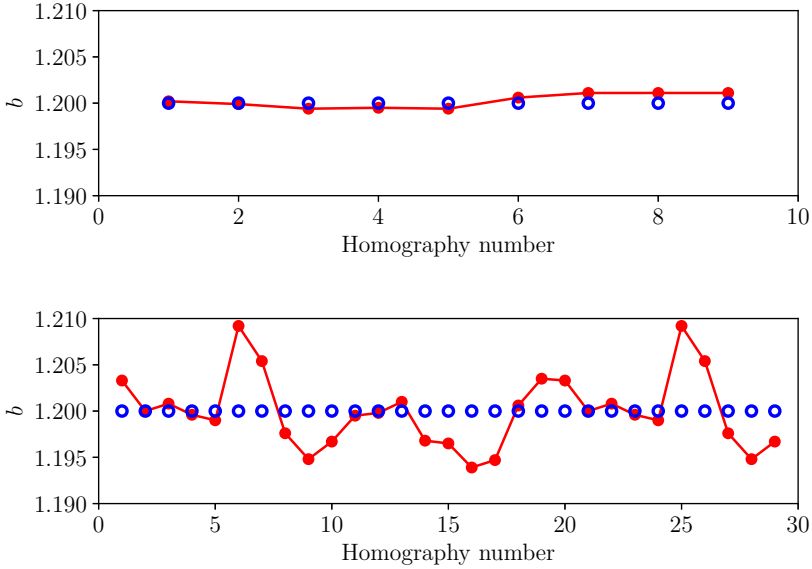


Figure 4: Estimated distance to the ground floor for pure translations (top) and general motion (bottom). The blue circles represent the ground truth and the red dots are the estimated parameters. Five pairs of homographies have been used in each frame.

where the starred variables denote the estimated values, and the values were averaged over 1000 randomised problems for different numbers of homographies and standard deviation. The results are shown in Figure 5.

5.3 Evaluation on the KITTI Dataset

In order to demonstrate the performance of the proposed method, and the advantage of imposing the planar motion model, the KITTI Visual Odometry / SLAM benchmark [2] was chosen. Since the proposed method is intended for planar surfaces where both cameras are tilted towards the floor—with possibly non-overlapping fields of view—the KITTI dataset is not an ideal testset. It does, however, contain subsequences with planar or near planar motion; therefore, by considering these trajectories, a qualitative study can be made. Furthermore, it demonstrates the robustness of the method in cases where the planar motion model is not strictly followed, *e.g.* irregularities in the ground floor (road or sidewalks) or vibrations causing the cameras to move in the direction of the

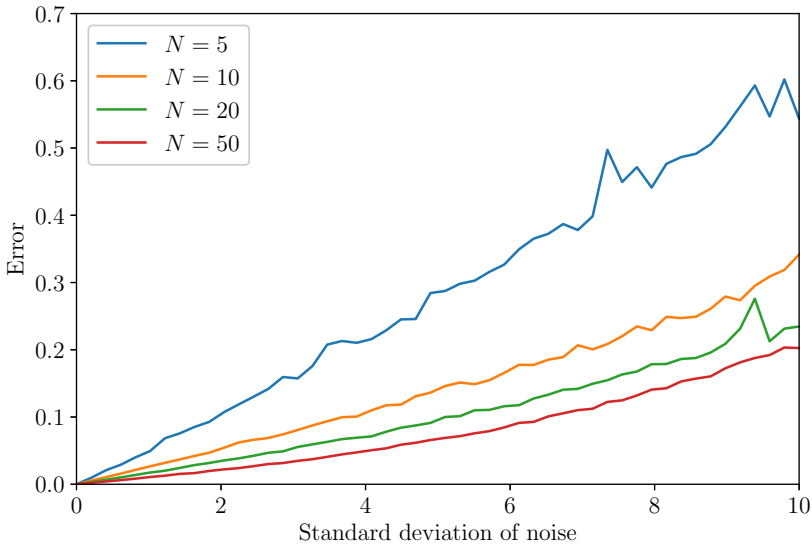


Figure 5: Error vs. noise for different amounts of pairs of homographies N . The error was estimated from the mean of 1000 randomly generated problem instances.

floor normal (speed bumps). Even under more controlled circumstances, it may be hard in practice to impose the planar motion model due to similar, but perhaps less severe, violations of the model assumptions.

In order to be able to estimate a homography compatible with the planar motion model, the images were cropped such that the majority of the image depict a planar surface. This is achieved by cropping the upper parts (sky and trees) and the side (posts and parked vehicles) before estimating the image, see Figure 6.

The homographies were estimated using the same setup as in the previous experiments, and the same method for decomposing the homographies were used. Furthermore, the tilt angles were estimated using all available homographies.

For longer sequences, it is customary to apply bundle adjustment in order to reduce the error propagation of the estimated trajectories. A well-known package for bundle adjustment is Sparse Bundle Adjustment (SBA) by Lourakis and Argyros [7]; however, the available methods do not enforce the planar motion model, but allow a general 6-DoF model (or a model where all motion parameters are decoupled). To this end, we will enforce the planar motion model, thus reducing the number of degrees of freedom and increasing the performance. Furthermore, we do not match any keypoints between



Figure 6: Image from the KITTI Visual Odometry / SLAM benchmark, Sequence 00. The thick black border indicates the area which is cropped prior to estimating the homography. This is only a crude approximation, and cannot guarantee that the image only depicts a planar surface, which is clearly shown as the bicyclists pass by the car, as well as the vehicle parked to the right. This, however, shows that the proposed method is robust enough to deal with violations of the planar motion model. Image credit: KITTI dataset [2].

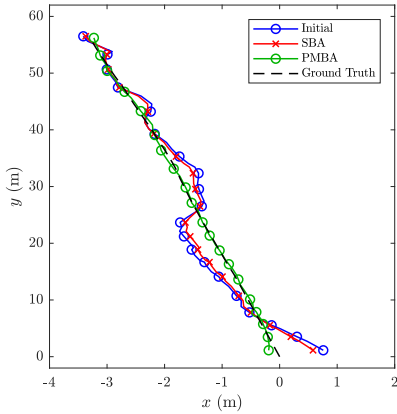
the stereo frames, in order to demonstrate that the increased performance can be obtained by enforcing the planar motion model despite not having overlapping fields of view. The initial trajectory used for the bundle adjustment is the one obtained by the method directly. The results are shown in Figure 7.

In both cases, the trajectories obtained by enforcing the planar motion model is qualitatively better than the ones obtained by a general 6-DoF model. By using the 6-DoF model the trajectory does not change significantly from the initial trajectory, which may be an indication that the underlying optimisation method is unable to find a better local minimum. It is plausible that this is because the general 6-DoF model neglects the (near) constant tilt angles throughout the movement of the car. In Figure 7d the car starts at an intersection with a strong left turn in which the sidewalk and a hedge is severely violating the planar motion model—an image of this is shown in Figure 8. Note, however, except from an over-estimation of the scale and the acuteness of the turn, the trajectory is physically reasonable, which is not the case for the solution obtained from a general 6-DoF model.

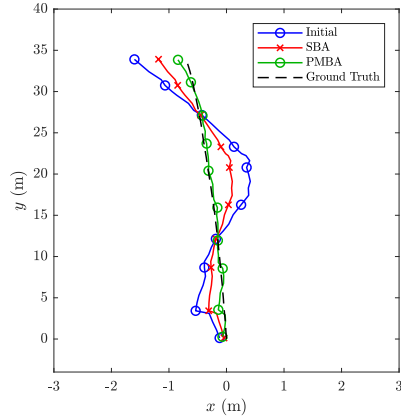
6 Conclusion

This paper has extended the work of Valtonen Örnhaug and Heyden [8] to account for two cameras positioned at different heights relative to the ground floor.

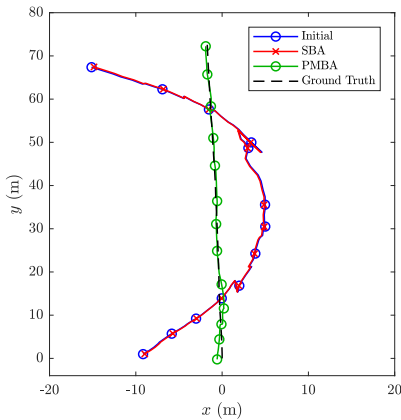
A method comparable in speed and robustness to the simplified case when the cameras are located at the same height has been proposed. The method can easily be incorpo-



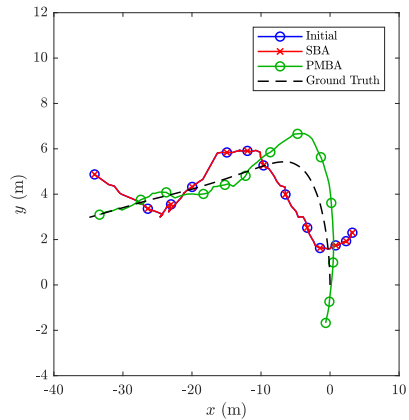
(a) Sequence 00 (60 images).



(b) Sequence 03 (40 images).



(c) Sequence 05 (80 images).



(d) Sequence 07 (80 images).

Figure 7: Estimated trajectories of subsequences of Sequence 00, 03, 05 and 07. In order to align the estimated trajectories with the ground truth, procrustes analysis was employed. The initial trajectory is obtained by the proposed method and the planar motion model enhanced nonlinearly refined by Planar Motion Bundle Adjustment (PMBA) compared to the general 6-DoF Spare Bundle Adjustment (SBA).



Figure 8: Image from the KITTI Visual Odometry / SLAM benchmark, Sequence 07. The image clearly violates the planar motion model, and the method fails in estimating the scale and acuteness of the turn. Image credit: KITTI dataset [2].

rated in a complete system to determine all fixed and non-fixed parameters of the mobile platform, and makes use of several pairs of homographies at each time step. Furthermore, a method for computing the distance to the ground floor for pure translations has been devised, which can be used as an initialization sequence of the mobile platform.

Experimental results using synthetic data show that both methods give a good estimate for the height of the cameras; however, using an initial calibration sequence yields a better result at the cost of imposing pure translation for the first frames.

On real data the proposed method remains stable and robust despite containing obstacles and motions violating the planar motion model. By enforcing the planar motion model using bundle adjustment, it is proven to perform better than generic 6-DoF methods.

Acknowledgment

This work has been funded by the Swedish Research Council through grant no. 2015-05639 ‘Visual SLAM based on Planar Homographies’.

References

- [1] O. D. Faugeras. “What Can Be Seen in Three Dimensions with an Uncalibrated Stereo Rig”. In: *European Conference on Computer Vision (ECCV)*. Santa Margherita Ligure, Italy, May 1992, pp. 563–578.

-
- [2] A. Geiger, P. Lenz, and R. Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Providence, RI, USA, June 2012, pp. 3354–3361.
 - [3] H. Hajjdiab and R. Laganière. “Vision-based Multi-Robot Simultaneous Localization and Mapping”. In: *Canadian Conference on Computer and Robot Vision (CRV)*. London, ON, Canada, May 2004, pp. 155–162.
 - [4] R. I. Hartley. “Estimation of relative camera positions for uncalibrated cameras”. In: *European Conference on Computer Vision (ECCV)*. Santa Margherita Ligure, Italy, May 1992, pp. 579–587.
 - [5] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
 - [6] B. Liang and N. Pears. “Visual Navigation using Planar Homographies”. In: *International Conference on Robotics and Automation (ICRA)*. Washington, DC, USA, May 2002, pp. 205–210.
 - [7] M. Lourakis and A. Argyros. “SBA: A Software Package for Generic Sparse Bundle Adjustment”. In: *ACM Trans. Math. Softw.* 36.1 (Mar. 2009), 2:1–2:30.
 - [8] M. Valtonen Örnthag and A. Heyden. “Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Funchal, Madeira, Portugal, Jan. 2018, pp. 568–575.
 - [9] M. Wadenbäck and A. Heyden. “Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Lisbon, Portugal, Jan. 2014, pp. 635–639.
 - [10] M. Wadenbäck and A. Heyden. “Planar Motion and Hand-Eye Calibration using Inter-Image Homographies from a Planar Scene”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Barcelona, Spain, 2013, pp. 164–168.
 - [11] M. Wadenbäck and A. Heyden. “Trajectory Estimation Using Relative Distances Extracted from Inter-image Homographies”. In: *Canadian Conference on Computer and Robot Vision*. Montréal, Canada, May 2014, pp. 232–237.
 - [12] M. Wadenbäck, M. Karlsson, A. Heyden, A. Robertsson, and R. Johansson. “Visual Odometry from Two Point Correspondences and Initial Automatic Tilt Calibration”. In: *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*. Porto, Portugal, Feb. 2017, pp. 340–346.

Fast Non-Minimal Solvers for Planar Motion Compatible Homographies

MARCUS VALTONEN ÖRNHAG

Centre for Mathematical Sciences, Lund University

Abstract: This paper presents a novel polynomial constraint for homographies compatible with the general planar motion model. In this setting, compatible homographies have five degrees of freedom—instead of the general case of eight degrees of freedom—and, as a consequence, a minimal solver requires 2.5 point correspondences. The existing minimal solver, however, is computationally expensive, and we propose using non-minimal solvers, which significantly reduces the execution time of obtaining a compatible homography, with accuracy and robustness comparable to that of the minimal solver. The proposed solvers are compared with the minimal solver and the traditional 4-point solver on synthetic and real data, and demonstrate good performance, in terms of speed and accuracy. By decomposing the homographies obtained from the different methods, it is shown that the proposed solvers have future potential to be incorporated in a complete Simultaneous Localization and Mapping (SLAM) framework.

1 Introduction

Polynomial systems of equations naturally arise in the field of computer vision, as an instrument of encoding geometric properties and other constraints one wishes to impose on the desired output. To solve a general system of polynomial equations, is, however, a task requiring certain endeavour, in order to produce a solver that is sufficiently fast and numerically stable.

In this paper we will investigate methods for Visual Odometry (VO), where the expected input is a sequence of images from a camera mounted on a mobile platform. The goal is to estimate the ego-motion of the platform, in indoor environments or other challenging scenes containing planar surfaces. This is done by considering homography based methods, where we enforce the general planar motion model. By imposing these constraints, thus lowering the total degrees of freedom of the motion parameters, it is possible to navigate robustly in scenes containing planar structures, which are problematic for VO systems where a general structure of the scene is assumed. It does, however, introduce a number of non-trivial polynomial constraints, and a proper framework for dealing with them must be employed.

The major contributions are:

- i. To derive a new polynomial constraint for the general planar motion model, and show that this, together with known constraints, are sufficient conditions for compatibility.
- ii. To develop a series of non-minimal solvers to enforce a weaker form of the general planar motion model, with a similar accuracy as the existing minimal solver, at a greatly reduced speed.
- iii. To demonstrate that pre-optimization on an early stage in the intended VO pipeline, by enforcing the general planar motion model on the homographies (but not a sequence of homographies), do not necessarily give an increased performance.

2 Related Work

Planar motion models with different complexity have been considered to increase robustness of navigation systems, by decreasing the total number of parameters to be recovered. In [16] a mobile platform with a single camera was considered, where the optical axis was parallel to the floor. This made it possible to parameterize the essential matrix by imposing a planar motion model with two translational components and a single rotational component. They propose a linear 3-point algorithm and a non-linear 2-point algorithm; however, only the direction of the translation can be recovered using their method. Furthermore, the alignment of the optical axis with the floor is not feasible for real-life applications. The same geometrical setup was considered in [1], but the algorithm was extended to stereo vision.

The fundamental matrix (or the essential matrix in the calibrated case) have been successfully used in many computer vision and robotics applications. Despite many promising navigation systems for general scenes, such methods will not work well under planar motion, as it is known to be ill-conditioned [7, 17]. As planar structures are common in man-made environments, researchers have considered alternative methods, many of which are based on inter-image homographies.

In [6] a planar motion model with one tilt parameter was considered, and they proposed a method for estimating all motion parameters. By allowing an arbitrary tilt about the floor normal Liang and Pears showed that the eigenvalues of the homography matrix is related to the rotation of the mobile platform, regardless of the tilt [14]. Their method, however, did not estimate the tilt parameters.

The method proposed in [22, 23] estimates the full set of motion parameters for the general planar motion model with five degrees of freedom by decoupling the overhead

tilt using an iterative scheme. From the same model assumption, but employing a dense matching scheme, Zienkiewicz and Davison devise a non-linear optimization scheme for the full set of motion parameters [25].

Without enforcing any model constraints, a general homography has eight degrees of freedom; however, a homography compatible with the general planar motion model only has five, as it is determined (up to scale) by the motion parameters. Due to the automated process of extracting and matching keypoints, homography solvers often employ RANSAC, or similar frameworks, to reject outliers. The number of iterations N required to select a set of keypoints only containing inliers is dependent on the size of the subset of keypoints n , the desired probability p and inlier ratio w , by the well-known relation $N = \log(1 - p) / \log(1 - w^n)$ [3]. One way of increasing the speed of such an algorithm is to reduce the necessary number of iterations, *i.e.* to create a solver that only uses the minimal number of point correspondences needed for the application. Such a solver is known as a *minimal solver*. In line with this argument, it is natural to consider a minimal homography solver compatible with the planar motion model, which has been constructed in [21]. They parameterize the homography as a linear combination of the basis elements of the null space of the corresponding DLT system, with 2.5 DLT constraints, and construct eleven quartic constraints on the homography matrix, which results in a system of eleven quartic equations in three variables. By generating the corresponding ideal, a basis for the quotient space can be constructed and the original problem solved by employing the action matrix method [15]. Their derivation involves steps of manual selection of basis monomials, which is a time-consuming task.

In [8] an automatic generator for polynomial systems was introduced, and several improvements have been made in recent years to increase the performance, see *e.g.* [9–11, 13]. Such methods have been successfully used in several computer vision applications, *e.g.* [20, 24].

Minimal solvers permit intrinsic constraints to be enforced on the solutions, while minimizing the number of necessary iterations in a RANSAC framework; however, there are cases when the minimal solvers are sensitive to noise, see *e.g.* [19], or when the complexity of minimal solver is large, thus making it very slow, see *e.g.* [12]. Under such circumstances constructing a non-minimal solver is a viable option.

3 Planar Motion

3.1 Problem Geometry

In this paper we consider a mobile platform with a single camera directed towards the floor. The world coordinate system is chosen such that the camera moves in the

plane $z = 0$. The scale is fixed by assuming the ground plane is positioned at $z = 1$, which is illustrated in Figure 1.

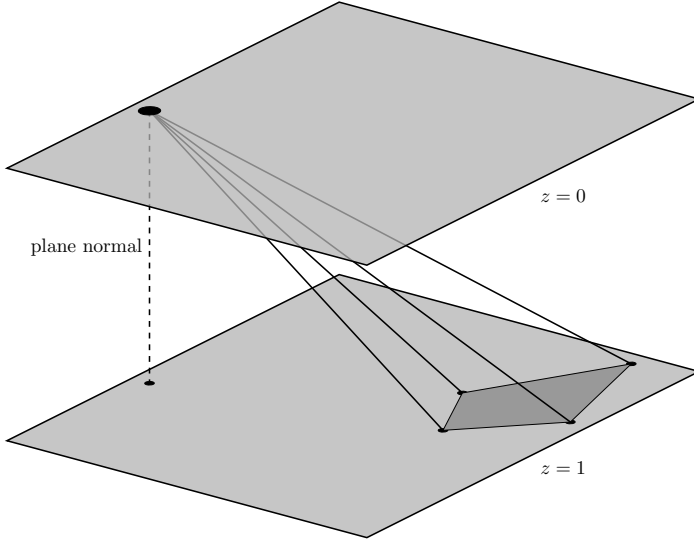


Figure 1: The problem geometry considered in this paper. The camera moves in the plane $z = 0$, and is tilted about the y -axis an unknown angle θ , followed by a rotation about the x -axis by an unknown angle ψ . The ground floor is positioned at $z = 1$.

We employ the parameterization used in [23], where the camera matrices for two consecutive poses A and B , are given by

$$\begin{aligned} \mathbf{P}_A &= \mathbf{R}_{\psi\theta}[\mathbf{I} \mid \mathbf{0}], \\ \mathbf{P}_B &= \mathbf{R}_{\psi\theta}\mathbf{R}_\varphi[\mathbf{I} \mid -\mathbf{t}], \end{aligned} \quad (1)$$

where $\mathbf{R}_{\psi\theta}$ is a rotation θ about the y -axis followed by a rotation ψ about the x -axis. As the mobile platform rotates about the plane normal (or z -axis) the angle φ varies, corresponding to \mathbf{R}_φ . The translation of the mobile platform is modelled by a translation vector $\mathbf{t} = (t_x, t_y, 0)^T$. It follows that the inter-image homography is given by

$$\mathbf{H} \sim \mathbf{R}_{\psi\theta}\mathbf{R}_\varphi\mathbf{T}_t\mathbf{R}_{\psi\theta}^T, \quad (2)$$

where $\mathbf{T}_t = \mathbf{I} - \mathbf{t}\mathbf{n}^T$ is the translation matrix corresponding to the translation \mathbf{t} , and $\mathbf{n} = (0, 0, 1)^T$ is a floor normal. The homography matrix can be made unique by imposing $\det \mathbf{H} = 1$, which will be assumed throughout the paper.

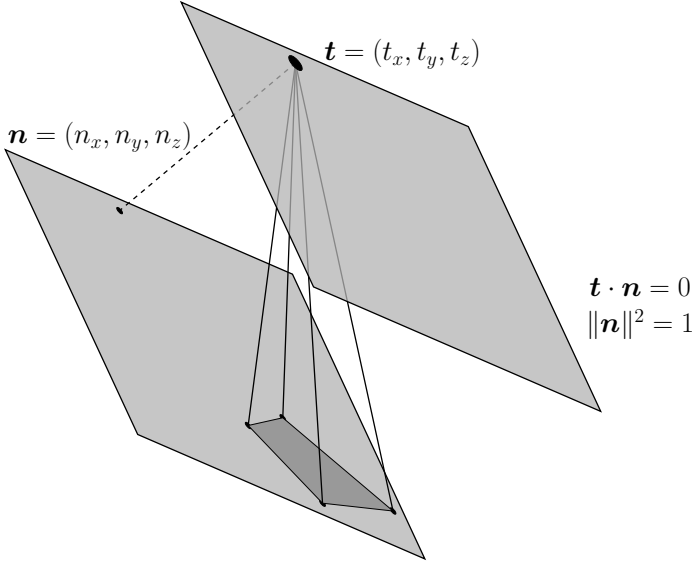


Figure 2: Revised problem geometry. The camera moves parallel to an unknown plane defined by a normal vector \mathbf{n} of unit length.

3.2 Parameter Recovery

In [23] an algorithm for recovering the full set of motion parameters for the general planar motion model was suggested. Their approach is to separate the overhead tilt $\mathbf{R}_{\psi\theta}$ from the nonconstant motion parameters. This can be achieved using a coordinate-descent like optimization scheme, where one of the tilt angles are fixed and the other is solved for. A more robust version by the same authors was introduced in [22] by using more than one homography, and incorporating the assumption of a fixed overhead tilt throughout the entire trajectory of the mobile platform.

4 Compatible Homographies

The *Direct Linear Transform* (DLT) equations for a pair of point correspondences $\mathbf{x} \leftrightarrow \hat{\mathbf{x}}$, where $\mathbf{x} = (x, y, 1)^T$ and $\hat{\mathbf{x}} = (\hat{x}, \hat{y}, 1)^T$ is given by

$$\begin{bmatrix} \mathbf{0} & -x^T & \hat{y}x^T \\ x^T & \mathbf{0} & -\hat{x}x^T \\ -\hat{y}x^T & \hat{x}x^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} = \mathbf{0}, \quad (3)$$

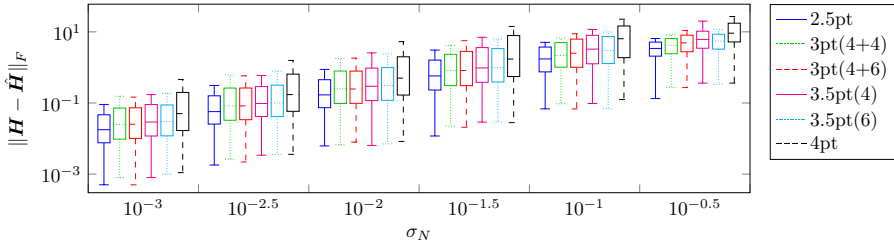


Figure 3: Distribution of homography error in the Frobenius norm for different noise levels σ_N . The proposed non-minimal solvers have a median error between the 2.5-point solver and DLT for all noise levels considered in the experiment.

where \mathbf{h}_k^T is the k :th row of the homography matrix \mathbf{H} . It is only necessary to consider the first two rows of the DLT system matrix in (3), as the third is a linear combination of the others. We will use these equations to parameterize the null space of the homography matrix \mathbf{H} in Section 6.

We will introduce some notation from algebraic geometry in order to outline the method employed to create the polynomial solvers we will consider in this paper.

4.1 The Action Matrix Method

Consider a polynomial system of equations

$$\begin{aligned} f_1(\mathbf{x}) &= 0, \\ &\vdots \\ f_s(\mathbf{x}) &= 0. \end{aligned} \tag{4}$$

The set of all solutions to (4), is known as an affine variety, and denoted $\mathbf{V}(f_1, \dots, f_s) \subset \mathbb{C}[\mathbf{x}]$, where $\mathbb{C}[\mathbf{x}]$ is the set of polynomials in \mathbf{x} with coefficients in \mathbb{C} . The ideal generated by f_1, \dots, f_s , is denoted

$$\langle f_1, \dots, f_s \rangle = \left\{ \sum_{i=1}^s h_i f_i : h_1, \dots, h_s \in \mathbb{C}[\mathbf{x}] \right\}. \tag{5}$$

Every ideal of $\mathbb{C}[\mathbf{x}]$ is finitely generated, thus, the polynomial system of equations (4) is defined by the generated ideal.

Under the assumption that the system has finitely many solutions, $I = \langle f_1, \dots, f_s \rangle$ is zero-dimensional and the quotient space $A = \mathbb{C}[\mathbf{x}]/I$ is finite dimensional [2].

Let $[f] = \{f + h \mid h \in I\}$ denote the coset and consider the operator $T_f : A \rightarrow A$, defined by $T_f([g]) = [fg]$. Since the quotient space A is finite-dimensional this operation can be represented by a matrix M_f , which is known as the *action matrix*. Furthermore, select a monomial basis for A , which we denote $\mathcal{B} = \{[x^{\alpha_j}]\}_{j \in J}$. Typically, such a basis is obtained by using an improved version of Buchberger's algorithm. When the action matrix $M_f = (m_{ij})$ acts on the basis elements a linear combination of the monomials forming the basis is obtained,

$$T_f([x^{\alpha_j}]) = [fx^{\alpha_j}] = \sum_{i \in J} m_{ij}[x^{\alpha_i}]. \quad (6)$$

Consequently, for $\mathbf{x} \in V(I)$,

$$f(\mathbf{x})x^{\alpha_j} = \sum_{i \in J} m_{ij}x^{\alpha_i}. \quad (7)$$

The basis \mathcal{B} may be represented by a vector \mathbf{b} , and since that (7) must hold for all basis elements, the problem can be reduced to an eigenvalue problem, given by

$$f(\mathbf{x})\mathbf{b}(\mathbf{x}) = M_f^T \mathbf{b}(\mathbf{x}). \quad (8)$$

By multiplying (4) by a set of monomials an equivalent, but larger problem is obtained. The problem of finding a suitable monomial basis remains, and there is no exact criterion for doing so; however, it is desirable to find the minimal set of monomials to make the problem solvable (or numerically stable). The coefficient matrix of the expanded matrix is known as an *elimination template*, and numerous attempts of optimizing these have been made, see e.g. [8, 9, 11, 13].

4.2 Necessary Conditions

In [21] eleven quartic constraints for planar motion compatible homographies were derived, and used to create a minimal solver. These constraints were experimentally obtained by randomly generating points on the manifold defining the planar motion compatible homographies. In order to be able to work over a finite field, the rotation matrices were constructed using Pythagorean triplets, hence integer versions of the problem were obtained. By doing so, the coefficients for the polynomials could be found. With this approach, however, one cannot rule out the existence of higher order polynomials, and thus, sufficiency cannot be proven. We will approach this differently in the next Section.

5 Sufficient Conditions

In this section, we will show that the eleven quartic constraints discussed in Section 4.2 are necessary but not sufficient. Furthermore, we show that by adding a sixth degree polynomial to the existing quartic constraints one may guarantee sufficiency. This is done by a novel parameterization, which allows one to efficiently compute the relevant elimination ideal.

Instead of considering the original camera matrices (1), choose the coordinate system such that $P_A = [I \mid \mathbf{0}]$, giving $P_B = [R_n(\varphi) \mid -\mathbf{t}]$. This can be thought of as travelling parallel to an unknown plane, as is illustrated in Figure 2, hence the homography can be parameterized as

$$H = R_n(\varphi) + \mathbf{t}\mathbf{n}^T, \quad (9)$$

where $\mathbf{t} = (t_x, t_y, t_z)^T$ is a translation vector orthogonal to the plane normal $\mathbf{n} = (n_x, n_y, n_z)^T$. The constraint of travelling parallel to the plane can be expressed as $\mathbf{t} \cdot \mathbf{n} = 0$ and, to fix the scale, one may assume that $\|\mathbf{n}\|^2 = 1$.

Let $\mathbf{q} = (1, q_x, q_y, q_z)$ be a unit quaternion, then $\mathbf{n} = (q_x, q_y, q_z)$ and the corresponding rotation matrix $R = R(\mathbf{q})$. Let the ideal generated by $\lambda H - R(\mathbf{q}) - \mathbf{t}\mathbf{n}^T = 0$ and $\mathbf{t} \cdot \mathbf{n} = 0$ be denoted I , then we seek the elimination ideal $I \cap \mathbb{K}[H]$, for some suitable field \mathbb{K} . Over a finite field, this can be done using Macaulay2 [5]. This results in the eleven quartic constraints found in [21] and an additional sixth degree polynomial¹. The constraints were symbolically verified to hold over \mathbb{C} as well.

6 Non-Minimal Solvers

In this section we consider different non-minimal solvers as an alternative to the minimal solver proposed in [21]. The main reason we consider such solvers is due to the minimal solver being computationally expensive. Also, there is no advantage between a 2.5-point solver and a 3-point solver, in terms of the number of iterations required to select a subset of inliers with a certain probability—this is a favourable trait, compared to the standard 4-point DLT solver. Furthermore, we would like to explore different methods using the novel sixth degree polynomial, derived in Section 5. Lastly, and most importantly for practical applications, the condition that the overhead tilt remains constant is not enforced by using any of the described method, but is achieved at a later step in the VO pipeline. Therefore, it is uncertain if pre-optimizing the homographies will yield a better end result.

¹ See the Appendix for implementation details.

In this paper we construct four different non-minimal solvers

3pt(4+4) a 3-point solver enforcing two quartic constraints,

3pt(4+6) a 3-point solver enforcing one quartic constraint and the sextic constraint,

3.5pt(4) a 3.5-point solver enforcing one quartic constraint,

3.5pt(6) a 3.5-point solver enforcing the sextic constraint.

These solvers were chosen because they enforce a weaker version of the general planar motion model in different ways. Furthermore, we will compare with the minimal solver and the standard 4-point DLT solver.

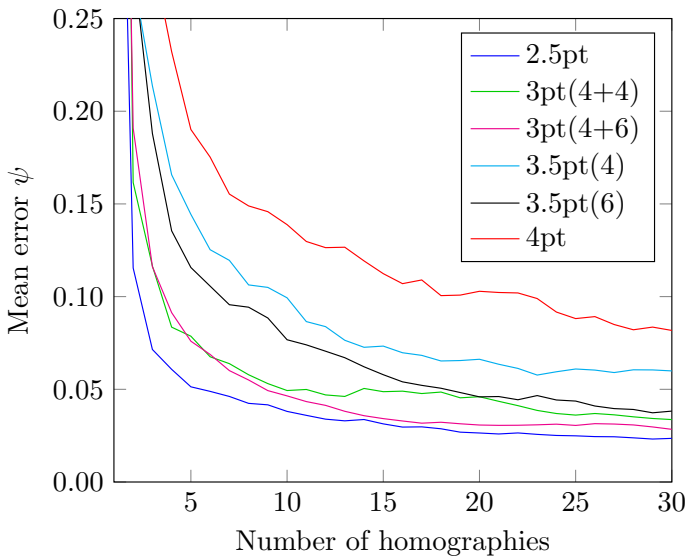


Figure 4: Mean error for ψ (in degrees) for 100 iterations.

From the eleven quartic constraints the first two w.r.t. DEGLEXP was chosen to construct the solvers; however, this was only chosen for reproducibility, as empirical tests showed that the size of the elimination template did not change by considering other pairs. Similarly, when constructing the 3pt(4+6) and 3.5pt(4) solvers only the first quartic constraint was chosen.

6.1 Parameterizing the Null Space

The standard approach, which we will utilize, is to parameterize the null space, and use the parameterization to obtain the desired polynomial system of equations. Using three DLT constraints (3) the homography matrix \mathbf{H} has a 3-dimensional null space, hence can be parameterized as

$$\mathbf{H}(\mathbf{z}) = \mathbf{H}_0 + z_1\mathbf{H}_1 + z_2\mathbf{H}_2. \quad (10)$$

Similarly, for the 3.5-point solvers, the null space is two-dimensional, leaving a single variable. For such systems, the action matrix method is replaced with a simpler root finding algorithm, involving the companion matrix [18]. In both cases this allow the homography to be written as a function $\mathbf{H} = \mathbf{H}(\mathbf{z})$, and inserted to any of the polynomial constraints f_i , yields an equation $f_i(\mathbf{H}(\mathbf{z})) = 0$ in the variable \mathbf{z} . For the 3-point solvers a system of two equations are obtained, and the corresponding ideal $I = \langle f_1, f_2 \rangle$ can be studied. The number of basis elements in the quotient space determines the number of solutions (for non-degenerate configurations).

For the 3pt(4+4) solver the basis consists of 16 elements, hence the polynomial system of equations has at most 16 solution. By using the automatic generator proposed by [10] an elimination template of size 20×36 was constructed. Due to the increased complexity of the sextic constraint the 3pt(4+6) solver has in general 24 solutions, and the corresponding elimination template is of size 31×55 . Furthermore, the coefficients in the elimination template are significantly more complex.

7 Experiments

7.1 Noise Sensitivity

In order to be comparable with the study of noise sensitivity for the minimal solver proposed in [21], the same setup was used. Homographies \mathbf{H}_j compatible with the general planar motion model were generated together with randomly generated keypoints \mathbf{x}_k with zero mean and unit variance. The image correspondences $\hat{\mathbf{x}}_k \sim \mathbf{H}_j\mathbf{x}_k$ were computed and normalized to unit variance. To simulate noise, a normal distributed term was added to \mathbf{x}_k and $\hat{\mathbf{x}}_k$ with standard deviation σ_N . The homographies obtained from the solvers were normalized such that $\det \mathbf{H}_j = 1$, and the error measured in the Frobenius norm of the difference between the ground truth and the estimated homographies, see Figure 3. The median sensitivity to noise of the proposed solvers is between the corresponding values for the 2.5-point solver and the 4-point (DLT) solver for all noise levels, which indicates a trade-off between accuracy and speed.

Secondly, we generate a sequence of homographies compatible with the general planar motion model, and use the method described in Section 3.2 to recover the motion parameters. For this example the noise level was kept constant while increasing the number of homographies used to estimate the motion parameters. For $\sigma_N = 10^{-2}$ the results are shown in Figure 4. The results for the other motion parameters, and different standard deviation σ_N follow the same trend, and can be found in the Appendix. Note, that the mean error for the 3-point solvers are close to the minimal solver after approximately 15 homographies are used.

Table 1: Mean execution time for 10,000 randomly generated problems.

Solver	Exec. time (ms)
2.5pt	0.7960
3pt(4+4)	0.1334
3pt(4+6)	1.6161
3.5pt(4)	0.0344
3.5pt(6)	0.2919
4pt	0.0334

7.2 Speed Evaluation

The solvers were implemented in MATLAB, with mex-compiled C++ routines. For a fair comparison, the 2.5-point method and the 4-point (DLT) method were also constructed this way. The 2.5-point method was generated using the automatic solver by Larsson *et al.* [10], thus producing a different elimination template than the one proposed in [21].

The execution time of the solvers was tested on a standard laptop computer, and the measurements include the complete process of estimating a homography, *i.e.* starting from point correspondences, the construction of the DLT system, extracting the null space through SVD, and—except for the 4-point solver—the parameter estimation for the basis elements of the null space, and construction of putative homographies.

The timing comparison is shown in Table 1, and the speed-up between the minimal 2.5-point solver and 3pt(4+4) solver is clear, however, in terms of speed the traditional 4-point solver is faster. In a complete RANSAC framework, however, the 3-point solver and the 4-point solver is closer in terms of speed, due to the 4-point solver requiring more iterations to achieve the same probability of selecting a subset containing only inliers.

When enforcing the sixth degree polynomial in the 3-point solver the number of solutions increase, hence the corresponding elimination template. Furthermore the com-

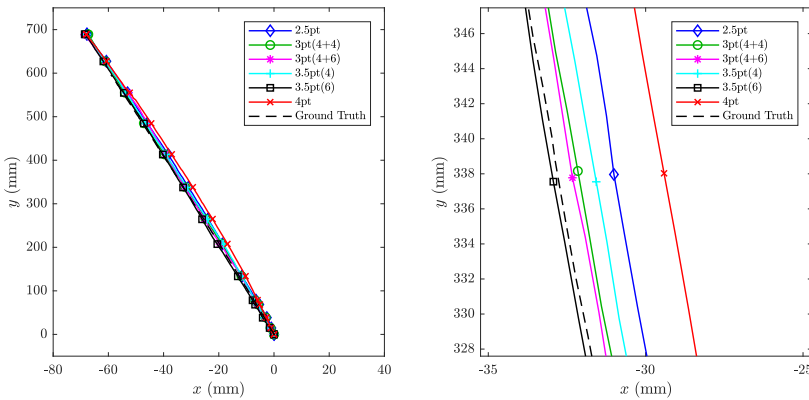


Figure 5: The sequence of the planar motion test set. The entire trajectory is shown to the left, and to the right a zoomed in version showing the difference in estimates by using different solvers. The total sequence contains 320 images.

plexity of the coefficients increase, and this holds true in the case of the 3.5-point solvers as well, which is why the execution time is faster without the sextic constraint.

7.3 Synthetic Image Evaluation

A sequence of synthetic images were generated, compatible with the general planar motion model, by cropping out images from a high-resolution image. In order to simulate the overhead tilt the original image was transformed prior to cropping it. An elliptic path containing 49 images were generated, as well as the corresponding ground truth.

The homographies were computed by extracting and matching SURF features. In order to estimate the trajectory, the homographies were decomposed using the method described in Section 3.2. The point correspondences were normalized prior to estimating the homographies to increase numerical stability. The extracted motion parameters were compared to the ground truth. Due to the random nature of the RANSAC, the recovered parameters were averaged over 100 iterations. The results are shown in Table 2.

We observe that the minimal solver is no longer producing the best results, and one may note the advantage of using non-minimal solvers. The exact reason why this phenomenon arises when considering synthetically generated sequences of images and not synthetically generated sequences of homographies is hard to pinpoint; however, we note the following differences:

- (a) the matched points, which are the input to the solvers do not follow a Gaussian distribution, and

Table 2: Mean estimation error for the motion parameters for the synthetic image test averaged over 100 iterations. Angles are measured in degrees, and translation in pixels. The best performance is in bold.

	2.5pt	3pt(4+4)	3pt(4+6)	3.5pt(4)	3.5pt(6)	4pt
ψ	0.0035	0.0005	0.0029	0.0004	0.0006	0.0007
θ	0.0019	0.0006	0.0013	0.0005	0.0006	0.0007
φ	1.08	0.68	0.77	0.76	0.73	0.86
\hat{t}	20.36	6.85	10.49	6.99	7.65	11.29

- (b) when analyzing the numerical rank of the elimination template it often has less than full rank, which is not the case for the Gaussian distributed noise. This holds true regardless of whether one normalizes the image points or not.
- (c) the numerical rank of the other elimination templates do not change when going from synthetic homographies to synthetic images.

7.4 Planar Motion Evaluation

The following experiments were conducted using a mobile robot with omnidirectional wheels, of model Fraunhofer IPA rob@work. A camera was mounted, directed towards the floor, and the ground truth was measured using a Nikon Metrology K600 optical tracking system. The system has an absolute accuracy of 100 μm . In Figure 6 example images from one of the sequences are shown. In the first test sequences the robot travels along a straight line, while keeping the orientation constant.

The same approach for computing the homographies in Section 7.3, after first compensating for geometric distortion. The results are shown in Figure 5, and more test results are shown in the Appendix. Note, that the minimal solver and the 4-point solver are the ones to deviate the most from the ground truth trajectory.

7.5 Evaluation on the KITTI Dataset

The KITTI Visual Odometry / SLAM benchmark [4] is a well-known evaluation dataset for SLAM frameworks, and contains several sequences with planar or near planar motion. A large portion of the images depict the roads, on which the car travels, however, one must note that this is only a coarse approximation to the general planar motion model, as all sequences contain non-planar structures to some extent, *e.g.* passing vehicles, pedestrians, traffic barriers and road signs. This, however, is a good way of testing the robustness of the proposed solvers, as future applications may not entirely fulfill the general planar motion model. Furthermore, in order to be able to consider the input



Figure 6: Example image from the planar motion sequence.

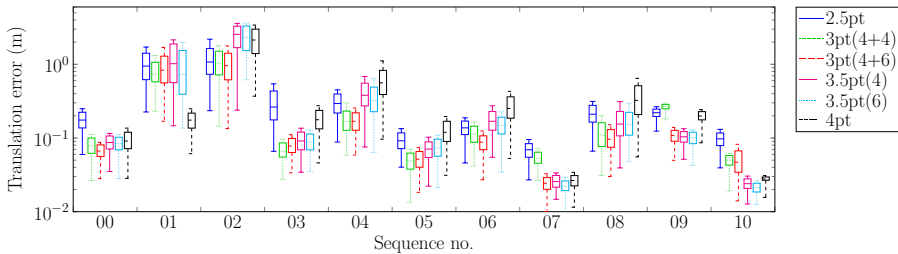


Figure 7: Mean translation error for the first ten images of the first camera of Sequence 00–10 of the KITTI dataset. These are all image sequences containing ground truth. At least one of the proposed solvers has the lowest mean translation error in all but one of the eleven cases. The mean translation error is computed over 500 iterations.

images as depicting a (near) planar scene, a subset of the image is cropped out before estimating the homographies, as is illustrated in Figure 8.

From the cropped images of the first eleven sequences of the KITTI dataset, for which ground truth data is available, the homographies were computed using SURF features, as in the previous experiments. The performance of the solvers was measured as the mean error of the Euclidean distance between the ground truth positions and estimated positions. The results are averaged over 500 tests, and include the first ten images of each sequence. This number was chosen to reduce the impact of error propagation, while still



Figure 8: Image from the KITTI Visual Odometry / SLAM benchmark, Sequence 03. The cropped area (contained within the thick border) depict the image used to compute the homographies. Image credit: KITTI dataset [4].

having a noticeable effect of imposing the general planar motion model. The differences of the estimation is due to the matching algorithm as well as the RANSAC framework for estimating the homographies. No non-linear refinement of the homographies were used. The results are shown in Figure 7.

The proposed solvers are robust, and produce a good initial estimate for the trajectory, and at least one of the proposed solvers has the lowest median translation error in all sequences, except Sequence 01.

This result may, to some extent, be unanticipated, however, one must not forget that the conditions for the general planar motion model is not fulfilled in the KITTI dataset. The disadvantage of using the minimal 2.5-point solver, it seems, is that the model is imposed exactly, whereas for the 4-point DLT solver, the model is completely disregarded, and instead are determined solely by the data. One possible answer to the results we observe on the KITTI dataset, which favours the the proposed solvers, is that it enforces a weaker form of the general planar motion model (since only one or two of the defining equations are considered) and tunes to the data in cases where the model assumptions are invalid.

8 Conclusions

In this paper a novel non-minimal polynomial constraint for homographies compatible with the general planar motion model has been derived. A series of non-minimal solvers have been proposed, which enforces one or two of the defining constraints. They have been demonstrated on synthetic and real data to perform well, and two of them are reported to be faster than the minimal solver.

In cases where the general planar motion model is a coarse approximation of the actual scene it is likely that the proposed solvers are more robust, compared to both the minimal solver and the 4-point solver. Hence, it has been demonstrated that pre-

maturely enforcing the planar motion model, without incorporating the fixed overhead tilt constraint, does not necessarily yield a better end result.

By incorporating the proposed solver in a complete SLAM system, it is likely that the total execution time will decrease due to a lower number of iterations needed by non-linear refinement of poses and scene points in a bundle adjustment framework.

Acknowledgments

The author gratefully acknowledges Märten Wadenbäck and Martin Karlsson for providing the data for the planar motion compatible sequences. This work has been funded by the Swedish Research Council through grant no. 2015-05639 ‘Visual SLAM based on Planar Homographies’.

References

- [1] T. Chen and Y.-H. Liu. “A robust approach for structure from planar motion by stereo image sequences”. In: *Machine Vision and Applications (MVA)* 17.3 (Aug. 2006), pp. 197–209.
- [2] D. A. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Second edition. Graduate Texts in Mathematics. Springer-Verlag New York, 2005.
- [3] M. Fischler and R. Bolles. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: *Communications of the ACM* 24.6 (1981), pp. 381–395.
- [4] A. Geiger, P. Lenz, and R. Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Providence, RI, USA, June 2012, pp. 3354–3361.
- [5] D. R. Grayson and M. E. Stillman. *Macaulay2 – a software system for research in algebraic geometry*. Available at <http://www.math.uiuc.edu/Macaulay2/>.
- [6] H. Hajjdiab and R. Laganière. “Vision-based Multi-Robot Simultaneous Localization and Mapping”. In: *Canadian Conference on Computer and Robot Vision (CRV)*. London, ON, Canada, May 2004, pp. 155–162.
- [7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
- [8] Z. Kukulova, M. Bujnak, and T. Pajdla. “Automatic generator of minimal problem solvers”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2008), pp. 302–315.

-
- [9] V. Larsson and K. Åström. “Uncovering Symmetries in Polynomial Systems”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2016), pp. 252–267.
- [10] V. Larsson, K. Åström, and M. Oskarsson. “Efficient Solvers for Minimal Problems by Syzygy-based Reduction”. In: *Computer Vision and Pattern Recognition (CVPR)* (July 2017), pp. 2383–2392.
- [11] V. Larsson, K. Åström, and M. Oskarsson. “Polynomial Solvers for Saturated Ideals”. In: *International Conference on Computer Vision (ICCV)* (Oct. 2017), pp. 2307–2316.
- [12] V. Larsson, Z. Kukelova, and Y. Zheng. “Camera Pose Estimation with Unknown Principal Point”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 2984–2992.
- [13] V. Larsson, M. Oskarsson, K. Åström, A. Wallis, Z. Kukelova, and T. Pajdla. “Beyond Gröbner Bases: Basis Selection for Minimal Solvers”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 3945–3954.
- [14] B. Liang and N. Pears. “Visual Navigation using Planar Homographies”. In: *International Conference on Robotics and Automation (ICRA)*. Washington, DC, USA, May 2002, pp. 205–210.
- [15] H. M. Möller and H. J. Stetter. “Multivariate polynomial equations with multiple zeros solved by matrix eigenproblems”. In: *Numerische Mathematik* 70.3 (1995), pp. 311–329.
- [16] D. Ortín and J. M. M. Montiel. “Indoor robot motion based on monocular images”. In: *Robotica* 19.3 (May 2001), pp. 331–342.
- [17] P. H. S. Torr, A. Zisserman, and S. J. Maybank. “Robust Detection of Degenerate Configurations while Estimating the Fundamental Matrix”. In: *Computer Vision and Image Understanding* 71.3 (1998), pp. 312–333.
- [18] L. Trefethen and D. Bau. *Numerical Linear Algebra*. SIAM, 1997.
- [19] B. Triggs. “Camera Pose and Calibration from 4 or 5 known 3D Point”. In: *International Conference on Computer Vision (ICCV)* (1991), pp. 278–284.
- [20] J. Ventura, C. Arth, G. Reitmayr, and D. Schmalstieg. “A minimal solution to the generalized pose-and-scale problem”. In: *Computer Vision and Pattern Recognition (CVPR)* (2014), pp. 422–429.
- [21] M. Wadenbäck, K. Åström, and A. Heyden. “Recovering Planar Motion from Homographies Obtained using a 2.5-Point Solver for a Polynomial System”. In: *International Conference on Image Processing (ICIP)*. Phoenix, AZ, USA, Sept. 2016, pp. 2966–2970.

- [22] M. Wadenbäck and A. Heyden. “Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Lisbon, Portugal, Jan. 2014, pp. 635–639.
- [23] M. Wadenbäck and A. Heyden. “Planar Motion and Hand-Eye Calibration using Inter-Image Homographies from a Planar Scene”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Barcelona, Spain, 2013, pp. 164–168.
- [24] Y. Zheng, Y. Kuang, S. Sugimoto, K. Åström, and M. Okutomi. “Revisiting the pnp problem: A fast, general and optimal solution”. In: *International Conference on Computer Vision (ICCV)* (2013), pp. 2344–2351.
- [25] J. Zienkiewicz and A. J. Davison. “Extrinsics Autocalibration for Dense Planar Visual Odometry”. In: *Journal of Field Robotics (JFR)* 32.5 (Aug. 2015), pp. 803–825.

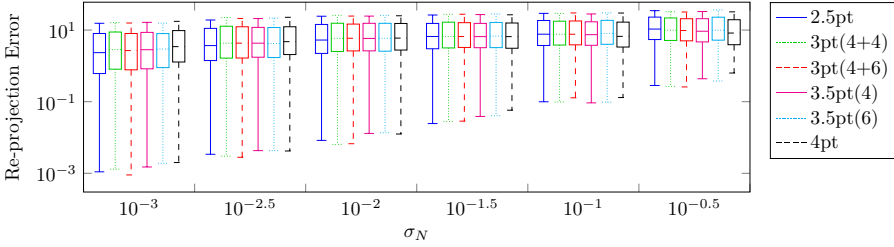


Figure 9: Mean re-projection error for different noise levels σ_N . Similar to Figure 3 the median value for the proposed solvers is between the corresponding values for the 2.5-point solver and DLT; however, for higher noise levels, we note an advantage for the non-minimal solvers and DLT compared to the minimal 2.5-point solver.

Appendix

Synthetic Experiments

Re-projection Error

In [74] the re-projection error for the point correspondences not used in order to obtain the estimated homographies were analysed, and we reproduce it here together with the proposed solvers. As seen in Figure 9, the trend is similar to what was observed in the previous case (*cf.* Frobenius norm estimate, Figure 3) namely, the median re-projection errors for the non-minimal solvers are between the corresponding values of the minimal 2.5-point solver and the 4-point solver (DLT) for all noise levels. Note, however, that for large noise levels, the 2.5-point solver does not perform as well as the non-minimal solvers or the traditional 4-point solver.

Motion Parameters

Additional plots for the second synthetic experiment, regarding the estimation of the motion parameters, is shown in Figure 11. The mean errors are for noise level $\sigma_N = 10^{-2}$. In Figure 12 the same parameters, but for $\sigma_N = 10^{-1}$ are shown.

Synthetic Images

Example images that were used in the synthetic image experiment is shown in Figure 10, and example output for the reconstructed path is shown in Figure 13 for all solvers. Neither the 4pt solver, nor the minimal solver performs best in this case.

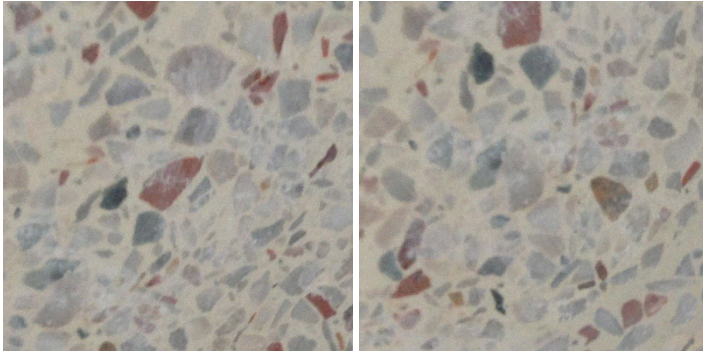


Figure 10: Two consecutive images from the synthetic dataset.

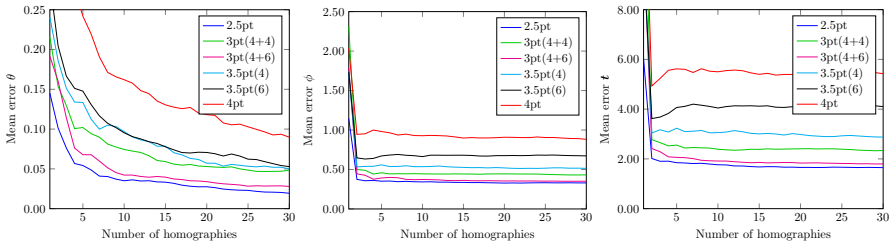


Figure 11: Mean error for θ , φ (in degrees) and t for 100 iterations with $\sigma_N = 10^{-2}$.

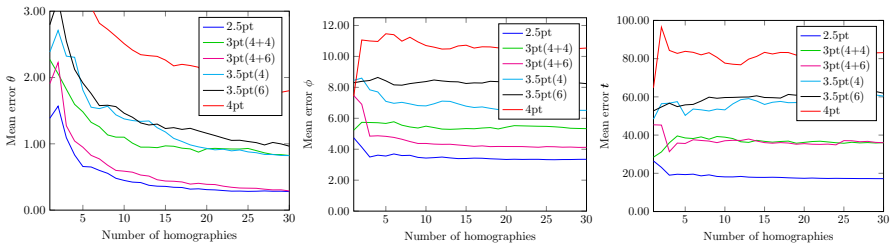


Figure 12: Mean error for θ , φ (in degrees) and t for 100 iterations with $\sigma_N = 10^{-1}$.

Experiments on Real Data

Planar Motion

We here show two more test cases conducted with the omnidirectional robot rob@work, from Section 7.4. In the first test case the robot moves forward (in relation to its own

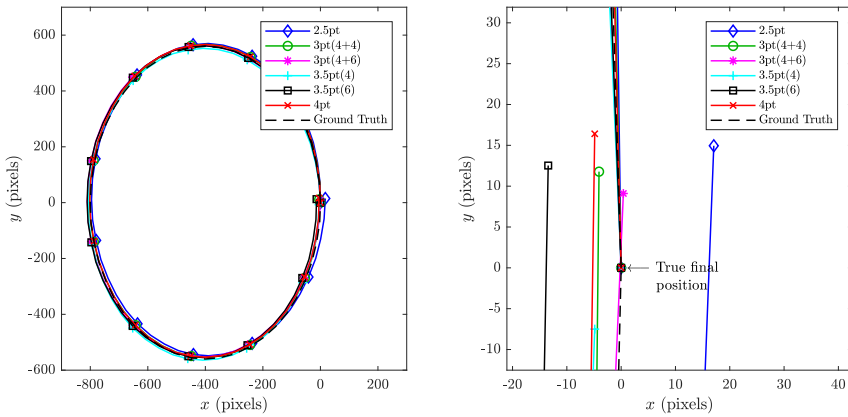


Figure 13: Example of reconstructed path for the synthetic image sequence for all solvers, and zoomed in at the final position (right image).

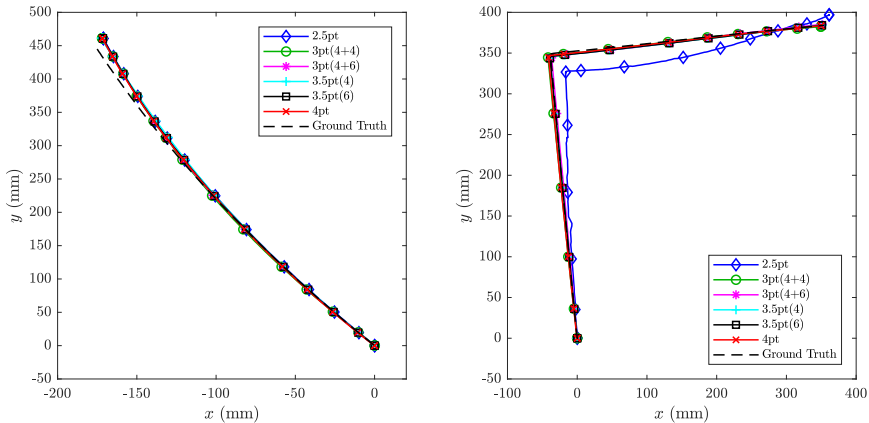


Figure 14: Estimated trajectories of the mobile robot used in the planar motion experiments for the “turn” experiment (left) and the “parallel parking” experiment.

frame), while rotating, thus creating a light turn. The sequence contains 344 images. In the second test case the robot simulates a sequences of parallel parking, by first driving straight and then making a sharp turn, while keeping the orientation constant. This sequence contains 325 images. The estimated paths are shown in Figure 14.

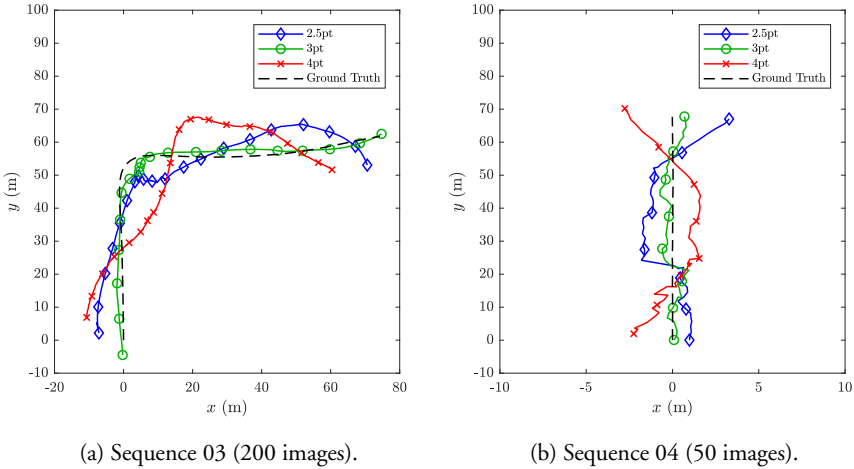


Figure 15: Estimated trajectories of subsequences of Sequence 03 and 04 of the KITTI dataset. Procrustes analysis has been carried out to align the estimated trajectories with the ground truth. Note that the aspect ratio differs between sequences, in order to clearly visualize the differences between the estimated trajectories. No non-linear refinement has been carried out in any of the test cases. The 3-point solver used is the $3pt(4+4)$ solver.

KITTI Dataset

To demonstrate some qualitative usage of the proposed solvers, four longer subsequences of the KITTI dataset were evaluated, see Figure 15. Only the $3pt(4+4)$ solver is shown in order to make the plots legible. For sequences of this length it is customary to use bundle adjustment, or some other non-linear refinement that minimize a physically meaningful error such as the geometric re-projection error or photometric error. In order for such optimization schemes to converge in a reasonable amount of time it is often necessary to supply a good initial guess of the trajectory. Due to the reduced computational complexity, comparable to using the 4-point DLT solver, when considered in a RANSAC framework, and the qualitative performance on the KITTI dataset, we find the proposed 3-point solver to be a suitable alternative to be incorporated in a SLAM system.

Efficient Radial Distortion Correction for Planar Motion

MARCUS VALTONEN ÖRNHAG

Centre for Mathematical Sciences, Lund University

Abstract: In this paper we investigate simultaneous radial distortion calibration and motion estimation for vehicles travelling parallel to planar surfaces. This is done by estimating the inter-image homography between two poses, as well as the distortion parameter. Radial distortion correction is often performed as a pre-calibration step; however, accurately estimating the distortion profile without special scene requirements may make such procedures obsolete. As many modern day consumer cameras are affected by radial distortion to some degree, there is a great potential to reduce production time, if properly implemented.

We devise two polynomial solvers, for radially distorted homographies compatible with different models of planar motion. We show that the algorithms are numerically stable, and sufficiently fast to be incorporated in a real-time frameworks. Furthermore, we show on both synthetic and real data, that the proposed solvers perform well compared to competing methods.

1 Introduction

When designing a Visual Odometry (VO) pipeline it is beneficial to integrate any prior knowledge of the intended environment or known motion model parameters. One particular instance, that will be further investigated in this paper, is the planar motion model, in which a vehicle travels on—or parallel to—a planar surface. Such a scenario is common in man-made environments, but can also accurately approximate outdoor scenarios under certain conditions, such as cars travelling on a highway. In the current literature we find several papers on planar motion models, restricted to fit particular use cases or pre-calibrated parameters [3, 6, 20]. The general case, however, was first introduced in [17] which incorporates two unknown overhead tilt angles, which are assumed to be constant throughout the trajectory of the vehicle. They assumed the floor is in the field of view of the camera, allowing them to compute the motion parameters through inter-image homographies. Another approach utilizing the floor to compute the motions was done by [6]. More recent development was done by [27, 28], and is the first to accurately recover the complete set of motion parameters using inter-image homographies. Other notable approaches include that of [30], in which a planar VO pipeline using a dense matching scheme was proposed.

In the most general setting, assuming the camera is rigidly mounted on the vehicle, the number of motion parameters are reduced to five, which should be compared to the general homography, which has eight degrees of freedom [7]. These parameters consist of the two overhead tilt parameters, and three non-constant parameters: one rotational angle (about the floor normal), and two translational components.

In order to obtain a homography, keypoints are extracted and matched. These keypoints then serve as input to the homography estimation algorithm. Since the extraction and matching steps are imperfect for any realistic image sequence, outliers and noise are prone to exist. Typical steps taken to resolve this issue include the use of robust estimation frameworks, *e.g.* RANSAC. It is at this point the benefit of working with fewer motion parameters come to light. Since fewer motion parameters demand fewer point correspondences in order to be estimated, one can select a minimal amount of points in the RANSAC framework. The fewer points you are able to select, the greater the probability of selecting only inliers. By doing so, one can reduce the number of RANSAC iterations.

In the general case, with four point correspondences, one may linearly extract the homography; however, if any of the motion parameters are known or constrained, this may no longer be the case, as the resulting systems of equations often are nonlinear. This poses a new type of problem—can we solve these equations sufficiently fast and accurate? Luckily, many methods from computational algebraic geometry [4] has been used in many computer vision problems, and certain frameworks already exist for how to proceed. One of the earliest, and still used today, was [10].

This paper is a revised journal version of [24], where we will consider the general planar motion model with unknown radial distortion, and devise a polynomial solver that can accurately recover the motion and distortion parameters in real-time applications. Furthermore, we propose a planar motion compatible minimal two point solver with radial distortion when the tilt angles are known. This situation arises when the tilt is pre-calibrated or can be accessed using external sensors, such as an IMU to extract the gravity direction. For indoor scenarios, assuming that the gravity direction is aligned with the floor normal—which often is a valid approximation—is equivalent to knowing the tilt angles. There are situations where similar assumptions can be made, without significant loss in accuracy, *e.g.* aerial imagery. Regardless of the situation, radial distortion is necessary to account for in any accurate VO pipeline, and is often done in a pre-calibration step, where the distortion parameters are obtained. By incorporating the parameter in the homography estimation process, we hope to eliminate this pre-processing step.

2 Related Work

2.1 Homography Estimation

The *Direct Linear Transform* (DLT) equations is a linear system of equations to extract a homography \mathbf{H} given a number of point correspondences. In the general setting, with eight degrees of freedom, the minimal case requires four point correspondences. To see this, let us consider a single pair of point correspondences on a common scene plane, denoted by $\mathbf{x} \leftrightarrow \hat{\mathbf{x}}$. They are related by a homography \mathbf{H} as $\lambda \hat{\mathbf{x}} = \mathbf{H}\mathbf{x}$, for some scalar $\lambda \neq 0$. Equivalently, we may express this as $\hat{\mathbf{x}} \times \mathbf{H}\mathbf{x} = \mathbf{0}$, thus eliminating the scale parameter λ , or,

$$\begin{bmatrix} \mathbf{0} & -\hat{w}\mathbf{x}^T & \hat{y}\mathbf{x}^T \\ \hat{w}\mathbf{x}^T & \mathbf{0} & -\hat{x}\mathbf{x}^T \\ -\hat{y}\mathbf{x}^T & \hat{x}\mathbf{x}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} = \mathbf{0}, \quad (1)$$

assuming homogeneous coordinates, *i.e.* $\mathbf{x} = [x, y, w]^T$ and $\hat{\mathbf{x}} = [\hat{x}, \hat{y}, \hat{w}]^T$, respectively. Here \mathbf{h}_k^T is the k :th row of the homography matrix \mathbf{H} . As the cross product introduces a linear dependence, only two of the equations are necessary, hence explaining why four point correspondences are minimal in the general case. Thus, using four point correspondences the problem can be transformed into finding the one-dimensional null space $\mathbf{h} = [\mathbf{h}_1^T \ \mathbf{h}_2^T \ \mathbf{h}_3^T]^T$, which is typically obtained using SVD of the coefficient matrix.

In the general planar motion model there are only five motion parameters, hence the minimal case requires but 2.5 point correspondences. Similarly, we may construct a system of equations, by using three point correspondences and discard the last equation in the corresponding DLT system. This can be written as $\mathbf{C}\mathbf{h} = \mathbf{0}$ where $\mathbf{C} \in \mathbb{R}^{5 \times 9}$ is the coefficient matrix. Again, this is a problem of finding the null space of \mathbf{C} ; however, the null space is now four-dimensional. As an additional step, one must now find the null space coefficients which makes \mathbf{H} a homography compatible with the general planar motion model. It was shown in [23, 26] that there are eleven quartic constraints (as well as a sextic constraint) in the elements of \mathbf{H} that has to be fulfilled in order to guarantee compatibility.

2.2 Modelling Radial Distortion

In order to compensate for the radial distortion, several models have been proposed. A classic method, still in use today, is the Brown–Conrady model [2], in which also tangential distortion is corrected. The division model introduced in [5], has gained attention as

it provides accurate approximations of the distortion profile with fewer parameters. For this reason, we will only consider the distortion model, and restrict ourselves to a single distortion parameter, as this allows us to use fewer point correspondences.

Let λ denote the distortion parameter. Then the distorted (or measured) image points can be expressed as $\mathbf{x}_i = [x_i, y_i, 1]^T$

$$\mathbf{x}_i^u = f(\mathbf{x}_i, \lambda) = \begin{bmatrix} x_i \\ y_i \\ 1 + \lambda(x_i^2 + y_i^2) \end{bmatrix}, \quad (2)$$

where \mathbf{x}_i^u are the undistorted image points, assuming the distortion center is aligned to the center of the image. Furthermore, we select the coordinate system such that the origin is aligned with the distortion center.

We may now modify the DLT equations (1) as the distortion parameter only appears in the homogeneous coordinates. Consider two point correspondences $\mathbf{x}_i \leftrightarrow \hat{\mathbf{x}}_i$, then

$$f(\hat{\mathbf{x}}_i, \lambda) \times \mathbf{H}f(\mathbf{x}_i, \lambda) = \mathbf{0}. \quad (3)$$

This approach has been used for the general case of radially distorted homographies [9], conjugate translations with radial distortion [21], and the case of jointly estimating lens distortion and affine rectification from coplanar features [22]. The last two use an explicit parameterization of the motion parameters, instead of trying to parameterize the null space of the DLT system. In common for all methods is that the resulting problem is a polynomial system of equations, and is solved by further reduction to an eigenvalue problem [4]. Automatic solvers for polynomial systems have been proposed, primarily using Gröbner bases, such as [10, 12–14, 16], or resultant based methods [1]. Alternative approaches include considering the problem as a Quadratic Eigenvalue Problem (QEP) [5, 8, 11].

3 The General Planar Motion Model

Consider a camera mounted rigidly on a vehicle travelling on a planar surface. We model this scenario by assuming that the camera moves in the plane $z = 0$, parallel to the surface on which the vehicle moves, located in $z = 1$. This parameterization also fixes the scale of the global coordinate system.

Consider two consecutive views, A and B , with the corresponding camera matrices

$$\begin{aligned} P_A &= \mathbf{R}_{\psi\theta} [\mathbf{I} \mid \mathbf{0}], \\ P_B &= \mathbf{R}_{\psi\theta} \mathbf{R}_\varphi [\mathbf{I} \mid -\mathbf{t}], \end{aligned} \quad (4)$$

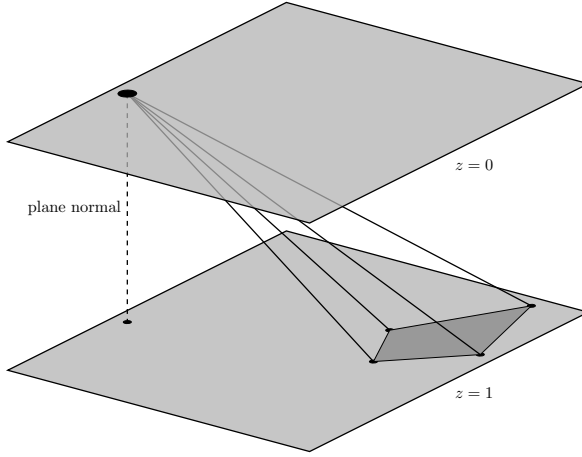


Figure 1: Illustration of the problem geometry considered in the paper. The camera is mounted rigidly on a mobile platform, thus travelling parallel to the ground floor in the plane $z = 0$. We allow a constant, but generally unknown, overhead tilt to be present, which is modelled by the angles ψ (about the x -axis) and θ (about the y -axis). Furthermore, the camera can rotate about the z -axis (by an angle φ) and translate in the plane $z = 0$, *i.e.* there are in total five degrees of freedom—three rotations and two translations. Figure reproduced from [24].

where the constant overhead tilt is modeled by $\mathbf{R}_{\psi\theta}$, and consists of a rotation θ about the y -axis followed by a rotation of ψ about the x -axis. Furthermore, we allow the vehicle to rotate an angle φ about the z -axis, which may vary. As the camera is assumed to be mounted rigidly on the vehicle, the height above the floor is constant, hence we may assume that it travels in the plane $z = 0$, leaving two translation components t_x and t_y , see Figure 1. From this, one may derive the corresponding inter-image homography

$$\mathbf{H} = \lambda \mathbf{R}_{\psi\theta} \mathbf{R}_{\varphi} \mathbf{T}_t \mathbf{R}_{\psi\theta}^T, \quad (5)$$

where $\mathbf{T}_t = \mathbf{I} - \mathbf{t}\mathbf{n}^T$ is a translation matrix, for the translation $\mathbf{t} = [t_x, t_y, 0]^T$, relative the plane normal $\mathbf{n} = [0, 0, 1]^T$. The homography matrix can be made unique by *e.g.* imposing $\det(\mathbf{H}) = 1$.

In addition to the DLT constraints, the elements of a homography compatible with the general planar motion model must satisfy a number of polynomial constraints. Such constraints were numerically derived in [26], where it was shown that there are at least eleven quartic constraints. The novel theoretical framework used in [23], showed that these constraints were necessary, but not sufficient; however, by adding a sextic constraint, it was shown that they are sufficient.

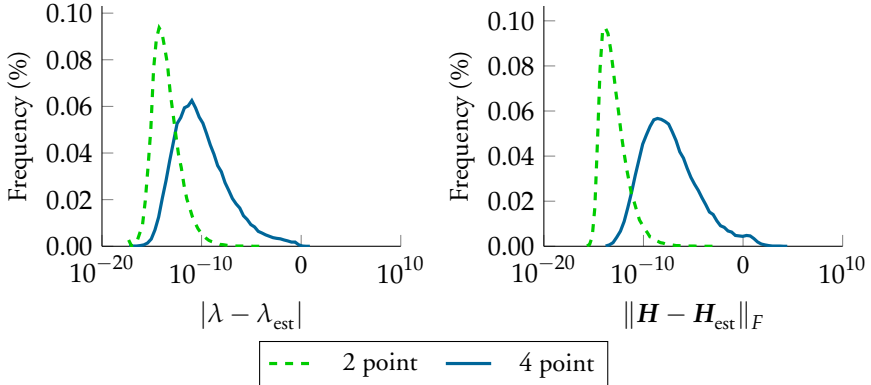


Figure 2: Error histogram of the estimated distortion parameter λ (left) and the homography \mathbf{H} for 100,000 random instances, for both of the proposed methods.

4 Polynomial Solvers

4.1 A Non-Minimal Relaxation (4 point)

In theory, one would be able to construct a minimal solver with three point correspondences, as there are six degrees of freedom—the five motion parameters discussed in Appendix 3, and the distortion parameter. In practice, however, this problem is hard, and we have yet to find a tractable solution which is numerically stable and sufficiently fast for real-time applications. Consequently, we have opted for a non-minimal four point relaxation. We do believe this is an acceptable compromise, as a general homography with a single distortion parameter requires 4.5 point correspondences for the minimal configuration. This effectively means one has to sample five point pairs to estimate a hypothesis. This section is largely reproduced from [24].

Similarly, to the approach in [9] we expand the third row of (3); however, we consider using only four point correspondences. This results in the following equation

$$(-\hat{y}_i h_{11} + \hat{x}_i h_{21})x_i + (-\hat{y}_i h_{12} + \hat{x}_i h_{22})y_i + (-\hat{y}_i h_{13} + \hat{x}_i h_{23})w_i = 0, \quad (6)$$

where $w_i = 1 + \lambda(x_i^2 + y_i^2)$ and $\hat{w}_i = 1 + \lambda(\hat{x}_i^2 + \hat{y}_i^2)$ are functions of the radial distortion parameter λ . There are eight monomials involved in this expression, namely

$$\mathbf{v}_1 = [h_{11} \ h_{12} \ h_{13} \ h_{21} \ h_{22} \ h_{23} \ \lambda h_{13} \ \lambda h_{23}]^T. \quad (7)$$

Using four point correspondences results in a system of equations, which can be written as

$$\mathbf{M}_1 \mathbf{v}_1 = \mathbf{0}, \quad (8)$$

where M_1 is a 4×8 matrix. For non-degenerate configurations the null space of M_1 is four-dimensional. Consequently, we may parameterize v_1 as

$$v_1 = \sum_{i=1}^4 \gamma_i n_i, \tag{9}$$

where γ_i are unknown basis coefficients. Since the last two monomials of v_1 depend on the previous elements, this relation has to be enforced when computing the basis coefficients γ_i . These give rise to two equations

$$v_8 = \lambda v_6 \quad \text{and} \quad v_7 = \lambda v_3. \tag{10}$$

Furthermore, we proceed to fix the scale by letting $\gamma_4 = 1$.

We will now use the second row of (3). Similarly, we may write this as

$$M_2 v_2 = \mathbf{0}, \tag{11}$$

where $M_2 \in \mathbb{R}^{4 \times 16}$. Here the null space vector v_2 consists of seven variables, and 16 monomials: $h_{31}, h_{32}, h_{33}, \lambda h_{33}$ and $\lambda^2 \gamma_i, \lambda \gamma_i, \gamma_i$ for $i = 1, 2, 3$ and $\lambda^2, \lambda, 1$. We may now proceed to eliminate the first three variables— h_{31}, h_{32} and h_{33} —as they are only present in four monomials. As we are using four point correspondences, yielding four equations, Gauss–Jordan elimination can be used. We obtain the following upon performing the elimination

$$\hat{M}_2 = \begin{bmatrix} h_{31} & h_{32} & \lambda h_{33} & h_{33} & \lambda^2 \gamma_1 & \lambda \gamma_1 & \gamma_1 & \dots & \lambda^2 & \lambda & 1 \\ 1 & & & & \bullet & \bullet & \bullet & \dots & \bullet & \bullet & \bullet \\ & 1 & & & \bullet & \bullet & \bullet & \dots & \bullet & \bullet & \bullet \\ & & 1 & & \bullet & \bullet & \bullet & \dots & \bullet & \bullet & \bullet \\ & & & 1 & \bullet & \bullet & \bullet & \dots & \bullet & \bullet & \bullet \end{bmatrix}. \tag{12}$$

It turns out that the columns of the right 4×12 submatrix are not independent. In order to generate a correct solver, it is important to generate integer instances satisfying these dependencies.

From the eliminated system $\hat{M}_2 v_2 = \mathbf{0}$ we get the four equations

$$\begin{aligned} h_{31} + f_1(\gamma_1, \gamma_2, \gamma_3, \lambda) &= 0, \\ h_{32} + f_2(\gamma_1, \gamma_2, \gamma_3, \lambda) &= 0, \\ \lambda h_{33} + f_3(\gamma_1, \gamma_2, \gamma_3, \lambda) &= 0, \\ h_{33} + f_4(\gamma_1, \gamma_2, \gamma_3, \lambda) &= 0, \end{aligned} \tag{13}$$

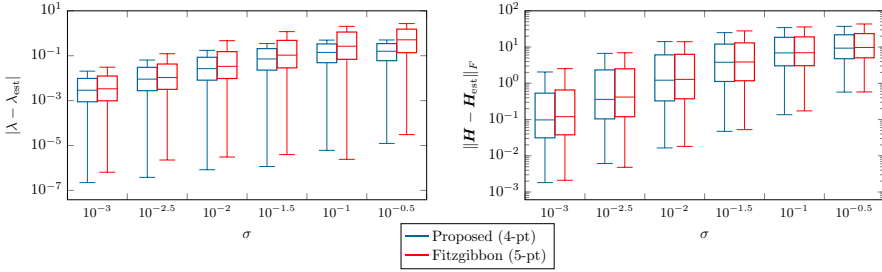


Figure 3: Distribution of estimation error in the distortion parameter λ , and the the homography \mathbf{H} (measured in the Frobenius norm) for different noise levels σ and unknown tilt. The proposed solver is compared to the five point solver [5]. Figure and caption reproduced from [24].

where $f_i(\gamma_1, \gamma_2, \gamma_3, \lambda)$ are polynomials in the variables $\gamma_1, \gamma_2, \gamma_3, \lambda$. Exploiting the relations between the last two equations of (13), an additional constraint is obtained

$$\lambda f_4(\gamma_1, \gamma_2, \gamma_3, \lambda) = f_3(\gamma_1, \gamma_2, \gamma_3, \lambda). \quad (14)$$

The eliminated variables h_{31}, h_{32} and h_{33} are polynomials of degree three, thus making (14) of degree four. Together with (10) we have three equations in four unknowns. Since we are able to express all elements of the homography \mathbf{H} as a function of four variables, we can enforce one of the 11 quartic constraints originally found in [26]. Evaluating these constraints using \mathbf{H} it turns out that ten of the constraints are of degree 12 and one of degree 10 due to cancellation of higher order terms. We choose the smallest one to build the polynomial solver.

Using the automatic generator [12] we find that there are 18 solutions to the problem in general, and by sampling a basis based on the heuristic presented in [15] an elimination template of size 177×195 could be created.

4.2 Minimal Solver with Known Tilt (2 point)

If the tilt angles are known, we can treat the planar motion case with radial distortion. In this case there are four degrees of freedom, and thus the minimal configuration requires two point correspondences. In this section, we will derive a novel solver for this case. Using a different approach than in the previous section, we may explicitly parameterize the homography. Let us use the following parameterization for the rotation matrix

$$\mathbf{R}_z = \begin{bmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (15)$$

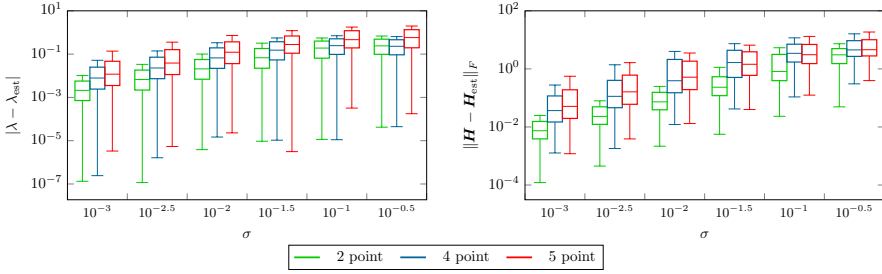


Figure 4: Distribution of estimation error in the distortion parameter λ , and the the homography \mathbf{H} (measured in the Frobenius norm) for different noise levels σ and known tilt (assumed to be compensated for). The proposed two point solver is compared to the four point and five point solver [5].

where $c^2 + s^2 = 1$, hence the sought homography is given by $\mathbf{H} \sim \mathbf{R}_z + \mathbf{t}\mathbf{n}^T$, where $\mathbf{t} = [t_x, t_y, 0]^T$ is a translation vector and $\mathbf{n} = [0, 0, 1]^T$ is a floor normal. Let us consider the modified DLT equations (3) again, but this time using two point correspondences. Using the first and third rows, we note that there are in total five unknowns— c , s , t_x , t_y and the radial distortion parameter λ —and in total eleven monomials, hence we may write the system as $\mathbf{M}\mathbf{v} = \mathbf{0}$, where \mathbf{M} is a 4×11 matrix and $\text{vec } \mathbf{v}$ is the vector of monomials. Furthermore, of these eleven monomials, we find only four which contain the variables c and s . Therefore, it is possible to use Gauss–Jordan elimination to eliminate these variables. The corresponding system, after elimination, is on the form

$$\hat{\mathbf{M}} = \begin{bmatrix} \lambda c & c & \lambda s & s & \lambda t_x & t_x & \lambda^2 t_y & \lambda t_y & t_y & \lambda & 1 \\ 1 & & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & 1 & & & \bullet & \bullet & 0 & \bullet & \bullet & 0 & 0 \\ & & 1 & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & & & 1 & \bullet & \bullet & 0 & \bullet & \bullet & 0 & 0 \end{bmatrix}. \quad (16)$$

Notice the pattern of zeros emerging in the eliminated system. This, and other more intricate relations, between the coefficients are necessary to account for in order to create an accurate polynomial solver.

From the above system we may introduce the functions g_i , such that

$$\begin{aligned} \lambda c + g_1(t_x, t_y, \lambda) &= 0, \\ c + g_2(t_x, t_y, \lambda) &= 0, \\ \lambda s + g_3(t_x, t_y, \lambda) &= 0, \\ s + g_4(t_x, t_y, \lambda) &= 0. \end{aligned} \quad (17)$$

where $g_i(t_x, t_y, \lambda)$ are polynomials in the variables t_x , t_y and λ . Furthermore, we utilize the two relations

$$\begin{aligned}g_1(t_x, t_y, \lambda) &= \lambda g_2(t_x, t_y, \lambda), \\g_3(t_x, t_y, \lambda) &= \lambda g_4(t_x, t_y, \lambda).\end{aligned}\tag{18}$$

The constraint $c^2 + s^2 = 1$ translates into

$$g_2^2(t_x, t_y, \lambda) + g_4^2(t_x, t_y, \lambda) = 1.\tag{19}$$

Now, we have a reduced system with three unknowns— t_x , t_y and λ —given by (18) and (19). It turns out that (18) are cubic and (19) are quartic, and by analyzing the dimension of the corresponding quotient ring, we find that the system has six solutions in total (it can be verified that the original system has six solutions as well). Using [15] an elimination template of size 18×24 was constructed.

5 Experiments

5.1 Synthetic Data

In this section we investigate the numerical stability and noise sensitivity of the proposed solver. We generate synthetic homographies, compatible with the general planar motion model (with and without tilt), as well as distortion parameters. Random scene points are generated using the homography and subsequently distorted using the division model.

The polynomial solvers were generated according to Appendix 4 in C++, and the mean runtime for the 4 point solver is 730 μ s and for the 2 point solver 13 μ s (measured over 100,000 instances on a standard desktop computer).

5.2 Numerical Stability

By using the described method, we generate noise-free problem instances. Similarly to [9], we use physically reasonable parameters, and cover a wide range of distortions by allowing the distortion parameter λ to be chosen at random in the interval $[-0.7, 0]$. In Figure 2 we show the error histogram for 100,000 random problem instances. When measuring the Frobenius norm error, the homographies have been normalized to $h_{33} = 1$.

From the histogram of the four point solver, we conclude that most parameters are estimated accurately, with an error in the range of 10^{-10} . Such an error is acceptable for most applications; however, some errors are higher, reaching an error around 10^{-2} . After careful analysis, we attribute this to the ten degree polynomial, which was added to

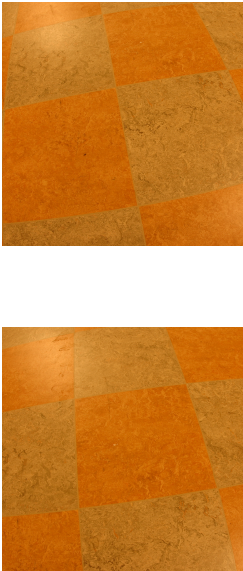
Input**Output**

Figure 5: Two radially distorted images (left) and the rectified and stitched panorama. The distortion parameter and homography was obtained using the proposed solver in a RANSAC framework. Blue border added for visualization. Figure and caption reproduced from [24].

conform with one of the original quartic constraints necessary for making the proposed solver compatible with the general planar motion model. Luckily, errors of the higher magnitude is less frequently occurring, and can be efficiently discarded in a robust framework, such as RANSAC. We will show that this is the case in the coming sections.

For the two point solver the errors are negligible for most computer vision applications, and is also a strong candidate for a robust framework, given that the assumptions of known tilt are met.

5.3 Noise Sensitivity

Similar to the previous section we generate synthetic problem instances, but corrupt the radially distorted image coordinates with Gaussian noise with a variance σ^2 . The noise is varied from mild to severe and at every noise level 10,000 problem instances were generated and the corresponding error measured. As a comparison, the five point

method based on the QEP approach [5] was used.

The result is shown in Figure 3. Note that the mean error for both quantities are lower for the proposed method compared to the five point method, for all noise levels. Analogously, but with known overhead tilt, we compare the two point solver to the other methods, see Figure 4. Here we see a clear benefit over the other, more general, methods.



Figure 6: Setup used in the panorama stitching experiment. Figure and caption reproduced from [24].

5.4 Image Stitching

In this section, we use the proposed four point solver in a classic stitching pipeline based on a standard approach for estimating a homography. The pipeline consists of first detecting and extracting SURF keypoints, followed by nearest neighbor matching. From all matched keypoints we select four at random and feed to the proposed solver in a RANSAC framework. The input images are taken using a digital camera with a fish-eye lens mounted on a tripod, overlooking a textured floor, see Figure 6. The camera tilt was fixed during the experiment, and only the tripod itself was moved, hence generating a motion compatible with the general planar motion model.

The output from the experiment is shown in Figure 5. Bundle adjustment or other

non-linear refinements of the obtained homography was not performed. Apart from being aligned with the correct edges we also note that lines that are straight in reality also appear straight in the final panorama, thus indicating that the radial distortion parameter was correctly estimated.

In terms of the efficiency of the robust framework, we use the same input images and compare the five point solver [5] with the proposed solver. This is done by recording the number of inliers as a function of the number of RANSAC iterations. We repeat the experiment 500 times, and show the average result in Figure 7, which shows that the proposed method consistently has a higher number of inliers.

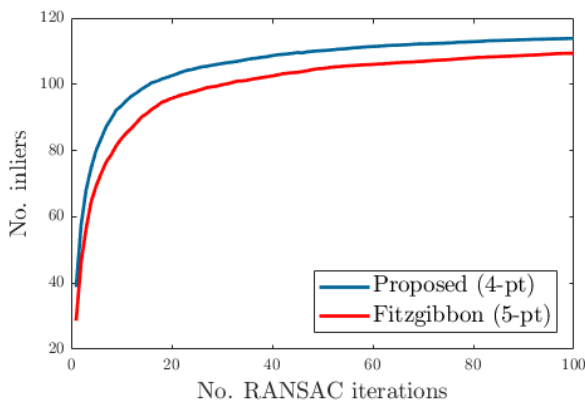


Figure 7: Number of inliers vs. number of RANSAC iterations for the images in Figure 5. The data has been averaged over 500 test instances. Figure and caption reproduced from [24].

5.5 Application to Visual Odometry

In this section we use real data from a mobile robot of model Fraunhofer IPA rob@work. The sequence was originally used in [29], but the radial distortion profile was pre-calibrated. On the mobile robot a camera is mounted rigidly, with an unknown overhead tilt, which excludes the application of the two point solver. The distortion is clearly noticeable and the field of view is almost entirely of the textured floor upon which the robot travels. Furthermore, the robot is equipped with omni-directional wheels, which allows for pure rotations. A reference system with an absolute accuracy of $100 \mu\text{m}$ tracks the robot as it moves about, and the resulting data is used as ground truth.

We consider three sequences:

Line Forward motion in a straight line with a constant orientation (320 images),

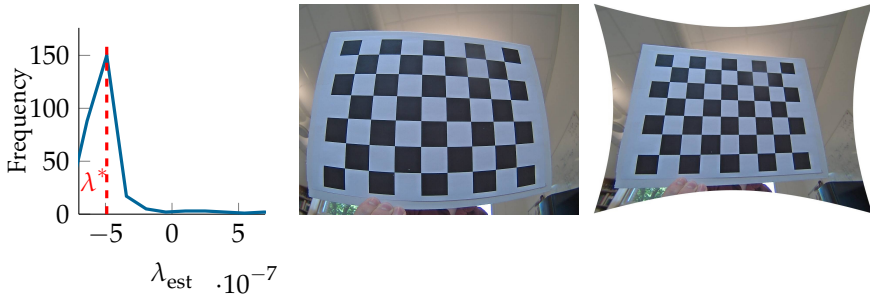


Figure 8: (Left) Histogram of estimated distortion parameters for the proposed method evaluating during the *parallel parking* sequence. The selected parameter λ^* is marked with a dashed line. (Middle) Undistorted image of a calibration chart, not part of the sequence. (Right) Rectified image using the estimated parameter λ^* . Figure and caption reproduced from [24].

Turn Forward motion while rotating, resulting in a slight turn (344 images),

Parallel Parking Forward motion followed by a sharp turn, while keeping constant rotation (325 images).

We consider a standard VO pipeline, including an initial solution via homography estimation, from which the initial camera poses are estimated (both intrinsic and extrinsic parameters) and finally a non-linear refinement step using bundle adjustment. Both the proposed method and the five point method [5] are capable of producing an initial estimation through inter-image homographies. Given a pair of consecutive images we may estimate the distortion parameter as well as the homography, using either solver, in a RANSAC framework. To extract the full set of motion parameters, we use the method in [27], hence establishing the initial poses. The estimated robot trajectory can then be extracted and compared to the ground truth. Note that in a complete VO pipeline, the initial position is important in order to avoid excessive amounts of bundle adjustment iterations, as these typically become large-scale optimization problems. Therefore, it is of interest to decrease the number of necessary iterations, by supplying a good initial guess.

The methods are comparable in terms of accuracy, as can be seen in Figure 9, with a slight preference for the proposed method. As noted in [23, 25], there is no significant boost in performance by pre-optimizing early on in the VO pipeline. One of the main issues is that the constant overhead tilt, due to the camera being rigidly mounted onto the robot, is not enforced throughout the entire trajectory by only considering a single homography. For consistency, one must consider an entire sequence of homographies. Nevertheless, the proposed method benefits from the same performance gain as was noted in Appendix 5.4; namely, that the number of RANSAC iterations required are

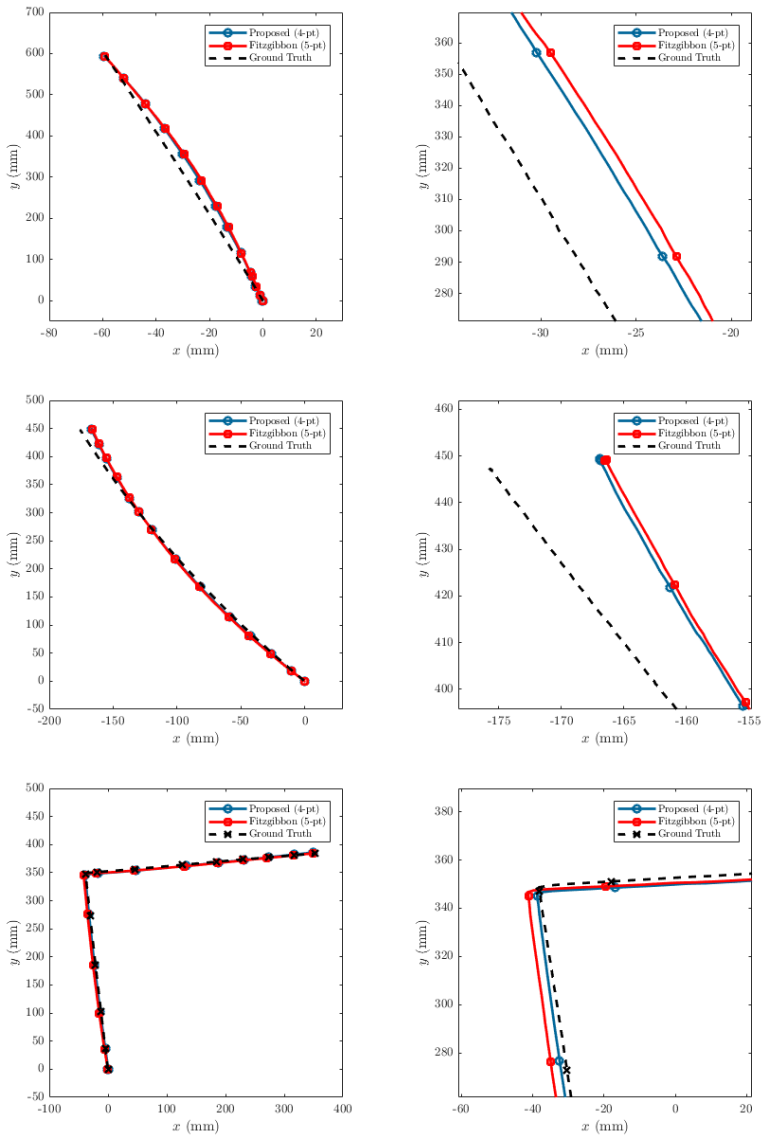


Figure 9: Estimated trajectories for *line*, *turn* and *parallel parking* of the VO experiment in Appendix 5.5. Images to the left show the entire trajectory, and the ones to the right are zoomed in on a region of interest. Figure and caption reproduced from [24].



Figure 10: Mosaics from the 1000 m sequence (top) and 1500 m sequence of the TAVT dataset [19] obtained using the proposed two point solver.

fewer than for the five point method.

The problem with considering only a single homography also affects the estimation of the radial distortion coefficient. In return, every pair of consecutive images yields a new estimate; however, we know a priori that it is constant through the trajectory. We propose using histogram voting as a robust way to obtain an initial guess. To evaluate the performance we use previously unseen images of calibration charts, that were acquired during the creation of the robot test sequences. We proceed by considering the *parallel parking* test sequence, and use the estimated parameters as a basis for the histogram voting experiment, see Figure 8. As can be seen, the chosen parameter λ^* , yields an acceptable initial solution, to be refined in a bundle adjustment framework.

5.6 Application to Aerial Imagery

In this final section we test the novel two point solver for aerial imagery. We use the TNT Aerial VideoTestset (TAVT) [19]. In this dataset, video sequences from a UAV have been recorded, at varying flight heights. The onboard global shutter camera is recording in full HDTV resolution at 30 fps, and suffers from mild radial distortion. Although the distortion is not severe, it was shown in [18] that failure to compensate for it results in severely distorted mosaicing attempts.

We use the sequences recorded at higher altitudes, in this case 1000 m and 1500 m

above ground, as these are not affected as much by potential non-zero and non-constant tilt, making the two point solver suitable. The solver is incorporated in a RANSAC framework, and the pipeline is identical to previous setups for real images. The sequences are subsampled to include every tenth image of the original sequences, hence contain 117 and 158 images each. The resulting mosaics are shown in Figure 10. Note that no non-linear optimization has been performed, nor histogram voting to determine the distortion profile. Yet, even for this simple pipeline, we manage to produce visually acceptable results, similar to those of the original articles [18, 19]. Perhaps the only noticeable difference is the lack of blending, seam-finding and other processing involved; however, these artifacts do not stem from the solver.

6 Conclusions

In this paper, we studied simultaneous radial distortion correction and motion estimation for planar motion. We proposed two polynomial solvers for estimating the homography and distortion parameter, and showed that they are sufficiently numerically robust and fast to be incorporated in a real-time VO pipeline. The proposed solvers were tested rigorously on both synthetic and real data, and were shown to be on par or superior to competing methods.

References

- [1] S. Bhayani, Z. Kukulova, and J. Heikkila. “A Sparse Resultant Based Method for Efficient Minimal Solvers”. In: *Computer Vision and Pattern Recognition (CVPR)*. June 2020.
- [2] D. C. Brown. “Decentering Distortion of Lenses”. In: *Photogrammetric Engineering* 32 (Mar. 1966), pp. 444–462.
- [3] T. Chen and Y.-H. Liu. “A robust approach for structure from planar motion by stereo image sequences”. In: *Machine Vision and Applications (MVA)* 17.3 (Aug. 2006), pp. 197–209.
- [4] D. A. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Second edition. Graduate Texts in Mathematics. Springer-Verlag New York, 2005.
- [5] A. W. Fitzgibbon. “Simultaneous linear estimation of multiple view geometry and lens distortion”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Dec. 2001.

- [6] H. Hajjdiab and R. Laganière. “Vision-based Multi-Robot Simultaneous Localization and Mapping”. In: *Canadian Conference on Computer and Robot Vision (CRV)*. London, ON, Canada, May 2004, pp. 155–162.
- [7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
- [8] G. Kayumbi and A. Cavallaro. “Multiview Trajectory Mapping Using Homography with Lens Distortion Correction”. In: *EURASIP Journal on Image and Video Processing* (Nov. 2008), p. 145715.
- [9] Z. Kukulova, J. Heller, M. Bujnak, and T. Pajdla. “Radial distortion homography”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 639–647.
- [10] Z. Kukulova, M. Bujnak, and T. Pajdla. “Automatic generator of minimal problem solvers”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2008), pp. 302–315.
- [11] Z. Kukulova, M. Bujnak, and T. Pajdla. “Polynomial Eigenvalue Solutions to the 5-pt and 6-pt Relative Pose Problems”. In: *British Machine Vision Conference (BMVC)*. Jan. 2008.
- [12] V. Larsson and K. Åström. “Uncovering Symmetries in Polynomial Systems”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2016), pp. 252–267.
- [13] V. Larsson, K. Åström, and M. Oskarsson. “Efficient Solvers for Minimal Problems by Syzygy-based Reduction”. In: *Computer Vision and Pattern Recognition (CVPR)* (July 2017), pp. 2383–2392.
- [14] V. Larsson, K. Åström, and M. Oskarsson. “Polynomial Solvers for Saturated Ideals”. In: *International Conference on Computer Vision (ICCV)* (Oct. 2017), pp. 2307–2316.
- [15] V. Larsson, Z. Kukulova, and Y. Zheng. “Camera Pose Estimation with Unknown Principal Point”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 2984–2992.
- [16] V. Larsson, M. Oskarsson, K. Åström, A. Wallis, Z. Kukulova, and T. Pajdla. “Beyond Gröbner Bases: Basis Selection for Minimal Solvers”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 3945–3954.
- [17] B. Liang and N. Pears. “Visual Navigation using Planar Homographies”. In: *International Conference on Robotics and Automation (ICRA)*. Washington, DC, USA, May 2002, pp. 205–210.

-
- [18] H. Meuel, S. Ferez, M. Munderloh, H. Ackermann, and J. Ostermann. “In-loop Radial Distortion Compensation for Long-term Mosaicking of Aerial Videos”. In: *Proc. of the 23rd IEEE International Conference on Image Processing (ICIP)* (Sept. 2016), pp. 2961–2965.
- [19] H. Meuel, M. Munderloh, M. Reso, and J. Ostermann. “Mesh-based Piecewise Planar Motion Compensation and Optical Flow Clustering for ROI Coding”. In: *APSIPA Transactions on Signal and Information Processing 4* (Oct. 2015).
- [20] D. Ortín and J. M. M. Montiel. “Indoor robot motion based on monocular images”. In: *Robotica* 19.3 (May 2001), pp. 331–342.
- [21] J. Pritts, Z. Kukelova, V. Larsson, and O. Chum. “Radially-Distorted Conjugate Translations”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.
- [22] J. Pritts, Z. Kukelova, V. Larsson, and O. Chum. “Rectification from Radially-Distorted Scales”. In: *Asian Conference of Computer Vision (ACCV)*. 2018, pp. 36–52.
- [23] M. Valtonen Örnhaug. “Fast Non-minimal Solvers for Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 40–51.
- [24] M. Valtonen Örnhaug. “Radially Distorted Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Valletta, Malta, Feb. 2020, pp. 568–575.
- [25] M. Valtonen Örnhaug and M. Wadenbäck. “Enforcing the General Planar Motion Model: Bundle Adjustment for Planar Scenes”. In: *Pattern Recognition Applications and Methods. ICPRAM 2019. Lecture Notes in Computer Science*. Vol. 11996. Springer International Publishing, 2020, pp. 119–135.
- [26] M. Wadenbäck, K. Åström, and A. Heyden. “Recovering Planar Motion from Homographies Obtained using a 2.5-Point Solver for a Polynomial System”. In: *International Conference on Image Processing (ICIP)*. Phoenix, AZ, USA, Sept. 2016, pp. 2966–2970.
- [27] M. Wadenbäck and A. Heyden. “Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Lisbon, Portugal, Jan. 2014, pp. 635–639.

- [28] M. Wadenbäck and A. Heyden. “Planar Motion and Hand-Eye Calibration using Inter-Image Homographies from a Planar Scene”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Barcelona, Spain, 2013, pp. 164–168.
- [29] M. Wadenbäck, M. Karlsson, A. Heyden, A. Robertsson, and R. Johansson. “Visual Odometry from Two Point Correspondences and Initial Automatic Tilt Calibration”. In: *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*. Porto, Portugal, Feb. 2017, pp. 340–346.
- [30] J. Zienkiewicz and A. J. Davison. “Extrinsics Autocalibration for Dense Planar Visual Odometry”. In: *Journal of Field Robotics (JFR)* 32.5 (Aug. 2015), pp. 803–825.

Enforcing the General Planar Motion Model: Bundle Adjustment for Planar Scenes

MARCUS VALTONEN ÖRNHAG¹ AND MÅRTEN WADENBÄCK²

¹*Centre for Mathematical Sciences, Lund University*

²*Department of Mathematical Sciences, Chalmers University of Technology
and the University of Gothenburg*

Abstract: In this paper we consider the case of planar motion, where a mobile platform equipped with two cameras moves freely on a planar surface. The cameras are assumed to be directed towards the floor, as well as being connected by a rigid body motion, which constrains the relative motion of the cameras and introduces new geometric constraints. In the existing literature, there are several algorithms available to obtain planar motion compatible homographies. These methods, however, do not minimise a physically meaningful quantity, which may lead to issues when tracking the mobile platform globally. As a remedy, we propose a bundle adjustment algorithm tailored for the specific problem geometry. Due to the new constrained model, general bundle adjustment frameworks, compatible with the standard six degree of freedom model, are not directly applicable, and we propose an efficient method to reduce the computational complexity, by utilising the sparse structure of the problem. We explore the impact of different polynomial solvers on synthetic data, and highlight various trade-offs between speed and accuracy. Furthermore, on real data, the proposed method shows an improvement compared to generic methods not enforcing the general planar motion model.

1 Introduction

The prototypical problem in geometric computer vision is the so called Structure from Motion (SfM) problem [12, 24]; the objective of which is to recover the scene geometry and camera poses from a collection of images of a scene. The SfM problem has, in some form or other, been studied since the very earliest days of photography, and many fundamental aspects of SfM were well understood already by the end of the 19th century [23]. Solving SfM problems of meaningful size and with actual image data, however, has been made possible only through the computerisation efforts that were commenced in the late 1970s, and which have since led to increasingly automatic methods for SfM. Modern SfM systems, *e.g.* *Bundler* [22] and other systems under the wider *BigSfM* banner¹ [1,

¹<http://www.cs.cornell.edu/projects/bigsfm/>

8], have managed to produce impressive city-scale reconstructions from large unordered and unlabelled sets of images.

A major paradigm in SfM, which has proven hugely successful, is Bundle Adjustment (BA) [26], which treats SfM as a large optimisation problem. With a parameterisation describing the scene geometry and the cameras, BA employs numerical optimisation techniques to find parameter values which best explain the observed images. Here, “best” is determined by evaluating a cost function which is often—but not always—chosen as the sum of squared geometric reprojection errors. The BA formulation of the SfM problem puts it in a unified framework which still has extensive model flexibility, *e.g.* with regards to (a) assumptions on the camera calibration, (b) different cost functions, and (c) different parameterisations of the cameras and the scene geometry—including implicit and explicit constraints to enforce a particular motion model.

While camera based Simultaneous Localisation and Mapping (SLAM) and Visual Odometry (VO) can be thought of as special classes of SfM, the computational effort to approach SLAM and VO via BA has traditionally been inhibiting, and for this reason, BA has mostly been used in offline batch processing systems such as the BigSfM systems mentioned earlier. During the last two decades, however, SLAM and VO systems have started incorporating regular BA steps to improve the consistency of the reconstruction and the precision of the camera pose estimation. Performance improvements across the spectrum—the algorithms, their implementation, the hardware—are paving the way for application specific BA to make its entrance in the area of real-time systems.

Especially in the case of visual SLAM, there are a number of factors which can be exploited to alleviate the computational burden compared to a more generic SfM system. The images are acquired in an ordered sequence, and this can significantly speed up the search for correspondences by avoiding the expensive “all-vs-all” matching. Additionally, a suitable motion model may often be incorporated in a SLAM system, which can be used *e.g.* (a) to further speed up the search for correspondences by predicting feature locations in subsequent images [5, 6], (b) to facilitate faster and more accurate local motion estimation via nonholonomic constraints [20, 21, 39] or other constraints which reduce the set of parameters [27, 31], or (c) to enforce globally a planar motion assumption on the camera motion [10, 18, 32].

In this paper, we present a BA approach to visual SLAM for the case of a stereo rig, where the cameras do not necessarily have an overlapping field of view, and where each of the two cameras move in parallel to a common ground plane. The present paper is an extension of the system described earlier in [30], to which a more extensive experimental evaluation has been added. In particular, we have investigated how initialisation using planar motion compatible homographies based on minimal [31] or non-minimal [27] polynomial solvers affect the final reconstruction.

2 Related Work

Planar Motion is a frequently occurring constrained camera motion, which arises naturally when cameras are attached to a ground vehicle operating on a planar ground surface. As mentioned in the introduction, deliberately enforcing planar motion can help to improve the quality of the reconstruction.

An early SfM approach to plane constrained visual navigation was proposed by Wiles and Brady [34, 35]. They suggested a hierarchical framework of camera parameterisations, and explored in detail the remaining structural ambiguity for each of these. The lasting contribution of this work lies chiefly in its classification and description of the different modes of motion. The least ambiguous level in the case of planar motion—which they called α -structure—contains only an arbitrary global scaling ambiguity and an arbitrary planar Euclidean transformation parallel to the ground plane, and is precisely the level aimed at in the present paper.

If the optical axis of the camera is either orthogonal or parallel to the ground plane, the parameterisation can be much simplified compared to the general case described by Wiles and Brady. This situation can of course also be achieved if the camera tilt is known with sufficient precision to allow a transformation to, *e.g.*, an overhead view. An approach for this case by Ortín and Montiel parameterises the essential matrix explicitly in the motion parameters, and then estimates the parameters using either a linear three-point method or a non-linear two-point method [18]. Scaramuzza used essentially the same parameterisation of the essential matrix, but combined it with an additional nonholonomic constraint based on the assumption that the local motion is a circular motion [20, 21]. Because of this additional constraint, the local motion can be computed from only one point correspondence, and this allows for an exceptionally efficient outlier removal scheme based on histogram voting.

Since the essential matrix is a homogeneous entity, it does not capture the length of the translation, and the maintaining of a consistent global scale then requires some additional information. One possibility for this, explored by Chen and Liu, is to add a second camera [4]. This allows the length of the local translation to be computed in terms of the distance between the two cameras, and since this remains constant, it provides a way to prevent scale drift.

If the camera is oriented such that it views a reasonable part of the ground plane, an alternative to using the essential matrix is to instead use homographies for the local motion estimation. This has the advantage that the length of the translation between frames can be expressed in terms of the height above the ground plane, which thus defines the global scale. The homography based approach by Liang and Pears is based on an eigendecomposition of the homography matrix, and it is shown that the rotation about

the vertical axis can be determined from the eigenvalues, regardless of the camera tilt [14]. Hajjdiab and Laganère parameterised the homography matrix under the assumption of only one tilt angle, and then transformed the images into a synthetic overhead view to compute the residual rigid body motion in the plane [10].

A more recent homography based method by Wadenbäck and Heyden, which also exploits a decoupling of the camera tilt and the camera motion, uses an alternating iterative estimation scheme to compute the two tilt angles and the three motion parameters [32, 33]. Zienkiewicz and Davison solved the same 5-DoF problem through a joint non-linear optimisation over all five parameters to achieve a dense matching of successive views, with the implementation running on a GPU to reach very high frame rates [39].

Valtonen Örnå and Heyden extended the general 5-DoF situation to handle a binocular setup, where the two cameras are connected by a fixed (but unknown) rigid body motion in 3D, and where the fields of view do not necessarily overlap [28, 29].

Bundle Adjustment is used to optimise a set of structure and motion parameters, and is typically performed over several camera views. Triggs *et al.* give an excellent overview [26]. Since the number of parameters optimised over is in most cases very large, naïve implementations will not work, and care must be taken to exploit the problem structure (*e.g.* the sparsity pattern of the Jacobian).

Generic software packages for bundle adjustment, which use sparsity of the Jacobian matrix together with Schur complementation to speed up the computations, include *SBA* (Sparse Bundle Adjustment) by Lourakis and Argyros, *sSBA* (Sparse Sparse Bundle Adjustment) by Konolige, and *SSBA* (Simple Sparse Bundle Adjustment) by Zach [13, 16, 37].

Additional performance gains may sometimes be obtained through parallelisation. GPU accelerated BA systems using parallelised versions of the Levenberg–Marquardt algorithm [11] and the conjugate gradients method [36] have been presented *e.g.* by Hänsch *et al.* and by Wu *et al.*. More recently, distributed approaches by *e.g.* Eriksson *et al.* and by Zhang *et al.* have employed splitting methods to make very large SfM problems tractable [7, 38].

The present paper extends the sparse bundle adjustment system for the binocular planar motion case by Valtonen Örnå and Wadenbäck. The aim of our approach is to exploit the particular structure in the Jacobian which arises due to the planar motion assumption for the two cameras. We demonstrate how this particular situation can be attacked via the use of nested Schur complementations when solving the normal equations. In comparison to the earlier paper [30], we have significantly extended the experimental evaluation of the system. Additionally, we have investigated the effect of enforcing the planar motion assumption earlier on a local level, by using homographies estimated such that they are compatible with this assumption [27, 31].

3 Theory

3.1 Problem Geometry

The geometrical situation we consider in this paper is that of two cameras which have been rigidly mounted onto a mobile platform. Due to this setup, which is illustrated in Figure 1, the cameras are connected by a rigid body motion which remains constant over time but which is initially not known. Each camera is assumed to be mounted in such a way that it can view a portion of the ground plane, but it is *not* a requirement that the cameras have any portion of their fields of view in common. The world coordinate system is chosen such that the ground plane is positioned at $z = 0$, whereas the cameras move in the planes $z = a$ and $z = b$, respectively. We may also, without loss of generality, assume that the centre of rotation of the mobile platform coincides with the centre of the first camera.

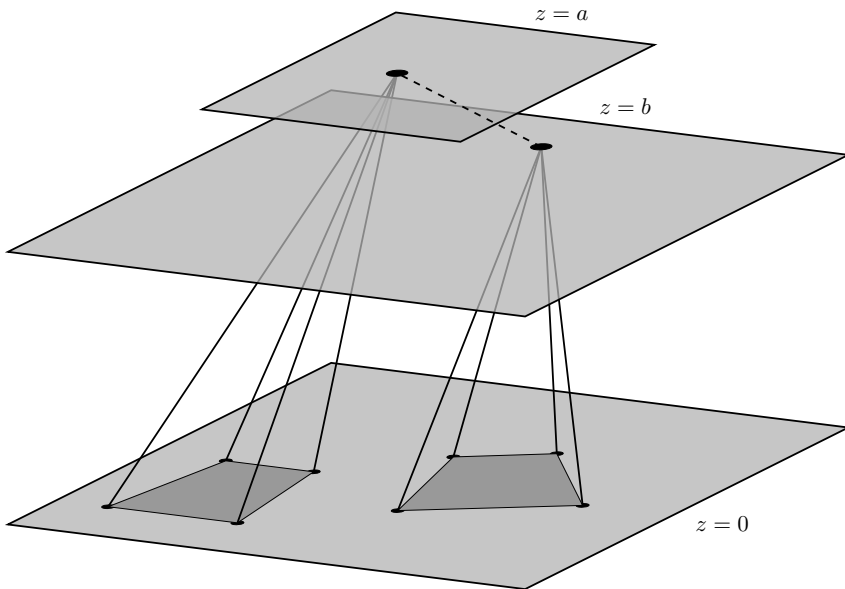


Figure 1: The problem geometry considered in this paper. The cameras are assumed to move in the planes $z = a$ and $z = b$, the relative orientation between them as well as the tilt towards the floor normal is assumed to be constant as the mobile platform moves freely.

3.2 Camera Parameterisation

We shall adopt the camera parameterisation for internally calibrated monocular planar motion that was introduced in [33]. With this parameterisation, the camera matrix associated with the image taken at position j will be

$$\mathbf{P}^{(j)} = \mathbf{R}_{\psi\theta} \mathbf{R}_{\varphi}^{(j)} [\mathbf{I} \mid -\mathbf{t}^{(j)}], \quad (1)$$

where $\mathbf{R}_{\psi\theta}$ is a rotation θ about the y -axis followed by a rotation of ψ about the x -axis. The motion of the mobile platform contains for each frame a rotation $\varphi^{(j)}$ about the z -axis, encoded as $\mathbf{R}_{\varphi}^{(j)}$, and a vector $\mathbf{t}^{(j)}$ for the translational part. The second camera, which is related to the first camera through a constant rigid body motion, uses the parameterisation

$$\mathbf{P}'^{(j)} = \mathbf{R}_{\psi'\theta'} \mathbf{R}_{\eta} \mathbf{T}_{\tau}(b) \mathbf{R}_{\varphi}^{(j)} [\mathbf{I} \mid -\mathbf{t}^{(j)}], \quad (2)$$

introduced in [28]. Here, ψ' and θ' are the tilt angles (defined in the same way as for the first camera), τ is the relative translation between the camera centres and η is the constant rotation about the z -axis relative to the first camera. None of the constant parameters are assumed to be known. The translation matrix $\mathbf{T}_{\tau}(b)$ is defined as $\mathbf{T}_{\tau}(b) = \mathbf{I} - \tau \mathbf{n}^T / b$, where $\tau = (\tau_x, \tau_y, 0)^T$, \mathbf{n} is a floor normal and b is the height above the ground floor. The global scale ambiguity allows us to set $a = 1$ without any loss of generality.

4 Prerequisites

4.1 Geometric Reprojection Error

Consider the pose of the first camera at position j , given by the camera matrix in (1), and let $\hat{\mathbf{x}}_i^{(j)}$ denote the estimated measurement of the scene point \mathbf{X}_i in homogeneous coordinates, *i.e.* $\hat{\mathbf{x}}_i^{(j)} \sim \mathbf{P}^{(j)} \mathbf{X}_i$. Let $\mathbf{x}_i^{(j)}$ denote the measured image point, and define the residual \mathbf{r}_{ij} as $\mathbf{r}_{ij} = \mathbf{x}_i^{(j)} - \hat{\mathbf{x}}_i^{(j)}$, where $\hat{\mathbf{x}}_i^{(j)}$ is the inhomogeneous representation of $\hat{\mathbf{x}}_i^{(j)}$. Analogously to the first camera, define the residual \mathbf{r}'_{ij} for the image of \mathbf{X}_i in the second camera. Given N stereo camera locations and M scene points, we seek to minimise the geometric reprojection error E given by

$$E(\boldsymbol{\beta}) = \sum_{i=1}^N \sum_{j=1}^M \|\mathbf{r}_{ij}\|_2^2 + \|\mathbf{r}'_{ij}\|_2^2, \quad (3)$$

where $\boldsymbol{\beta}$ is the parameter vector consisting of the camera parameters and the scene points.

4.2 The Levenberg–Marquardt Algorithm

For bundle adjustment it is common to use the Levenberg–Marquardt algorithm (LM) which solves the augmented normal equations

$$\left(\mathbf{J}^T \mathbf{J} + \mu \mathbf{I}\right) \delta = \mathbf{J}^T \boldsymbol{\varepsilon}, \quad (4)$$

where \mathbf{J} is the Jacobian of the cost function, $\boldsymbol{\varepsilon}$ the residual vector, and $\mu \geq 0$ is the *damping parameter*. The reader is referred to [16, 26] for more details regarding the LM algorithm and its application to bundle adjustment. There are other options to the LM algorithm, *e.g.* the dog-leg solver [15] and preconditioned CG [3]; however, LM is one of the most commonly used algorithms today, and is used in modern systems such as SBA [16] and sSBA [13]. Note, however, that SBA assumes the camera parameters for each camera to be decoupled, which is not the case for this specific problem geometry.

4.3 Obtaining an Initial Solution for the Camera Parameters

Homographies can be estimated in a number of different ways; however, the classical approach is to compute point correspondences from matching robust feature points in subsequent images. Popular feature extraction algorithms include SIFT [17] and SURF [2], but many more are available and implemented in various computer vision software. When the putative point correspondences have been matched a popular choice is to use RANSAC (or similar frameworks) to robustly estimate a homography. Such an approach is suitable in order to discard mismatched feature points. A well-known method is the Direct Linear Transform (DLT); however, it requires four point correspondences, and does not generate a homography compatible with the general planar motion model. A good rule of thumb is to use a minimal amount of point correspondences, since the probability of finding a set of points containing only inliers decreases with each additional point that is used. However, as *e.g.* Pham *et al.* point out, for very severely noisy data it may in some cases still be preferable to use a non-minimal set [19].

In [31] a minimal solver compatible with the general planar motion model was studied. It was shown that a homography compatible with the general planar motion model must fulfil 11 quartic constraints, and that, a minimal solver only requires 2.5 point correspondences. In a recent paper, a variety of different non-minimal polynomial solvers are considered, partly because of execution time, but also because of sensitivity to noise [27]. These non-minimal solvers enforce a subset of the necessary and sufficient conditions for compatibility with the general planar motion model, thus enforcing a weaker form of it. By accurately making a trade-off between fitting the model constraints (*i.e.* using more model constraints) and tuning to data (*i.e.* using more point correspondences), one can increase the performance for noisy data. It is important to note that the assumption of

constant tilt parameters cannot be enforced by only considering a single homography, and, therefore, pre-optimisation in an early step of the complete SfM pipeline is not guaranteed to yield better performance.

Once the homographies are obtained, one may enforce the constant tilt constraint by employing the method proposed by Wadenbäck and Heyden [32], to obtain a good initial solution for the monocular case. The method starts by computing the overhead tilt $\mathbf{R}_{\psi\theta}$ from an arbitrary number of homographies, followed by estimating the translation and orientation about the floor normal.

The method by Valtonen Örnberg and Heyden [28] extended the method to include the stereo case, and starts off by treating the two stereo trajectories individually, and estimates the tilt parameters by employing the monocular method described in the previous paragraph. Once the monocular parameters are known for the individual tracks, the relative pose can be extracted by minimising an algebraic error in the relative translation between the cameras, followed by estimating the relative orientation about the floor normal.

4.4 Obtaining an Initial Solution for the Scene Points

Linear triangulation of scene points does not guarantee that all points lie in a plane, and the resulting initial solution would not be compatible with the general planar motion model. In order to obtain a physically meaningful solution we make use of the fact that there is a homography relating the measured points and the ground plane positioned at $z = 0$.

Given a camera \mathbf{P} , an image point \mathbf{x} and the corresponding scene point $\mathbf{X} \sim (X, Y, 0, 1)^T$, they are related by $\mathbf{x} \sim \mathbf{P}\mathbf{X} = \mathbf{H}\tilde{\mathbf{X}}$, where \mathbf{H} is the sought homography. By denoting the i :th column of \mathbf{P} by \mathbf{P}_i , it may be expressed as $\mathbf{H} = [\mathbf{P}_1 \ \mathbf{P}_2 \ \mathbf{P}_4]$, where $\tilde{\mathbf{X}} \sim (X, Y, 1)^T$ contains the unknown scene point coordinates. It follows that the corresponding scene point can be extracted from $\tilde{\mathbf{X}} \sim \mathbf{H}^{-1}\mathbf{x}$.

In the presence of noise, using more than one camera results in different scene points, which all will be projected onto the plane $z = 0$. In order to triangulate the points we compute the centre mass; such an approach is computationally inexpensive, however, it is not robust to outliers, which have to be excluded in order to get a reliable result.

5 Planar Motion Bundle Adjustment

5.1 Block Structure of the Jacobian

Denote the unknown and constant parameters for the first camera path by $\gamma = (\psi, \theta)$ and the second camera path by $\gamma' = (\psi', \theta', \tau_x, \tau_y, b, \eta)$. Furthermore, let the non-constant parameters for position j be denoted by $\xi_j = (\varphi^{(j)}, t_x^{(j)}, t_y^{(j)})$. Given N stereo camera positions and M scene points, the following, highly structured Jacobian J , is obtained

$$J = \begin{bmatrix} \Gamma_{11} & & A_{11} & & & & & & B_{11} \\ \vdots & & & & & & & & \\ \Gamma_{1N} & & & & & & A_{1N} & & B_{1N} \\ \vdots & & & & & & & & \\ \Gamma_{M1} & & & & & & & & B_{M1} \\ \vdots & & & & & & & & \\ \Gamma_{MN} & & & & & & & & B_{MN} \\ & & \Gamma'_{11} & & A'_{11} & & & & B'_{11} \\ & & \vdots & & & & & & \\ & & \Gamma'_{1N} & & & & A'_{1N} & & B'_{1N} \\ & & \vdots & & & & & & \\ & & \Gamma'_{M1} & & A'_{M1} & & & & B'_{M1} \\ & & \vdots & & & & & & \\ & & \Gamma'_{MN} & & & & A'_{MN} & & B'_{MN} \end{bmatrix}, \quad (5)$$

where we use the following notation for the derivative blocks

$$\begin{aligned} A_{ij} &= \frac{\partial r_{ij}}{\partial \xi_j}, & B_{ij} &= \frac{\partial r_{ij}}{\partial \tilde{X}_i}, & \Gamma_{ij} &= \frac{\partial r_{ij}}{\partial \gamma}, \\ A'_{ij} &= \frac{\partial r'_{ij}}{\partial \xi_j}, & B'_{ij} &= \frac{\partial r'_{ij}}{\partial \tilde{X}_i}, & \Gamma'_{ij} &= \frac{\partial r'_{ij}}{\partial \gamma'}, \end{aligned} \quad (6)$$

where $\tilde{X}_i = (X_i, Y_i)$ are the unknown scene coordinates. This can be written in a more compact manner as

$$J = \begin{bmatrix} \Gamma & \mathbf{0} & A & B \\ \mathbf{0} & \Gamma' & A' & B' \end{bmatrix}. \quad (7)$$

5.2 Utilising the Sparse Structure

In SfM, the number of scene points is often significantly larger than the number of cameras, which makes Schur complementation tractable, and can significantly decrease

the execution time. Standard Schur complementation is, however, not directly applicable due to the constant parameters giving rise to the blocks $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$. We will, however, show in this section, that it is indeed possible to use *nested Schur complements*, *i.e.* to recursively apply Schur complements to different parts, and that, in fact, several of the intermediate computations can be stored, thus drastically decreasing the computational time. First, note that the approximate Hessian $\mathbf{J}^T \mathbf{J}$, in compact form, can be written

$$\mathbf{J}^T \mathbf{J} = \begin{bmatrix} \mathbf{C} & \mathbf{E} \\ \mathbf{E}^T & \mathbf{D} \end{bmatrix}. \quad (8)$$

Here the contribution from the constant parameters are stored in \mathbf{C} , the contribution from the nonconstant parameters and the scene points are stored in \mathbf{D} , and the mixed contributions are stored in \mathbf{E} . Furthermore, the matrix \mathbf{D} can be written as

$$\mathbf{D} = \begin{bmatrix} \mathbf{U} & \mathbf{W} \\ \mathbf{W}^T & \mathbf{V} \end{bmatrix}, \quad (9)$$

with block diagonal matrices $\mathbf{U} = \text{diag}(\mathbf{U}_1, \dots, \mathbf{U}_N)$ and $\mathbf{V} = \text{diag}(\mathbf{V}_1, \dots, \mathbf{V}_M)$, where

$$\begin{aligned} \mathbf{U}_j &= \sum_{i=1}^M \mathbf{A}_{ij}^T \mathbf{A}_{ij} + \mathbf{A}'_{ij}{}^T \mathbf{A}'_{ij}, \\ \mathbf{V}_i &= \sum_{j=1}^N \mathbf{B}_{ij}^T \mathbf{B}_{ij} + \mathbf{B}'_{ij}{}^T \mathbf{B}'_{ij}, \\ \mathbf{W}_{ij} &= \mathbf{A}_{ij}^T \mathbf{B}_{ij} + \mathbf{A}'_{ij}{}^T \mathbf{B}'_{ij}. \end{aligned} \quad (10)$$

First, note that the system $(\mathbf{D} + \mu \mathbf{I})\boldsymbol{\delta} = \boldsymbol{\varepsilon}$, where \mathbf{D} is defined as in (9), is not affected by the constant parameters. Such a system reduces to that of the unconstrained case, which can be solved using standard SfM frameworks, such as SBA, or other packages utilising Schur complementation.

We will now show how to efficiently treat the decomposition of (8) as nested Schur complements, by reducing the problem to a series of subproblems of the form used in SBA and other computer vision software packages. In order to do so, consider the augmented normal equations (4) in block form

$$\begin{bmatrix} \mathbf{C}^* & \mathbf{E} \\ \mathbf{E}^T & \mathbf{D}^* \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_c \\ \boldsymbol{\delta}_d \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varepsilon}_c \\ \boldsymbol{\varepsilon}_d \end{bmatrix}, \quad (11)$$

where $\mathbf{C}^* = \mathbf{C} + \mu \mathbf{I}$ and $\mathbf{D}^* = \mathbf{D} + \mu \mathbf{I}$ denote the augmented matrices, with the added contribution from the damping factor μ , as in (4). Now, utilising Schur comple-

mentation yields

$$\begin{bmatrix} \mathbf{C}^* - \mathbf{E}\mathbf{D}^{*-1}\mathbf{E}^T & \mathbf{0} \\ \mathbf{E}^T & \mathbf{D}^* \end{bmatrix} \begin{bmatrix} \delta_c \\ \delta_d \end{bmatrix} = \begin{bmatrix} \varepsilon_c - \mathbf{E}\mathbf{D}^{*-1}\varepsilon_d \\ \varepsilon_d \end{bmatrix}. \quad (12)$$

Let us take a step back and reflect over the consequences of the above equation. First, note that \mathbf{D}^{*-1} is present in (12) twice, and is infeasible to compute explicitly. This can be avoided by introducing the auxiliary variable δ_{aux} , defined as

$$\mathbf{D}^* \delta_{\text{aux}} = \varepsilon_d. \quad (13)$$

Again, such a system is not affected by the constraints of the constant parameters, and can be solved with standard computer vision software. Furthermore, we may introduce Δ_{aux} and solve the system $\mathbf{D}^* \Delta_{\text{aux}} = \mathbf{E}^T$ in a similar manner by iterating over the columns of \mathbf{E}^T . Since the number of constant parameters are low, such an approach is highly feasible, but the performance can be further boosted by storing the Schur complement and the intermediate matrices not depending on the right-hand side, from the previous computations of obtaining δ_{aux} from (13).

When the auxiliary variables have been obtained, we proceed to compute δ_c from

$$(\mathbf{C}^* - \mathbf{E}\Delta_{\text{aux}}) \delta_c = \varepsilon_c - \mathbf{E}\delta_{\text{aux}}, \quad (14)$$

and, lastly, δ_d by back-substitution

$$\mathbf{D}^* \delta_d = \varepsilon_d - \mathbf{E}^T \delta_c. \quad (15)$$

Again, by storing the computation of the Schur complement and intermediate matrices, these can be reused to solve (15) efficiently.

6 Experiments

6.1 Initial Solution

The inter-image homographies were estimated using the MSAC algorithm [25] from point correspondences by extracting SURF keypoints and applying a KNN algorithm to establish the matches. In the first experiment, we use the standard DLT solver, the minimal 2.5 pt solver [31] and the four different polynomial solvers studied in [27].

In all experiments we use all available homographies, and extract the monocular parameters using the method proposed in [32]. Similarly, the binocular parameters were extracted using [28]. When all motion parameters have been estimated the camera path is reconstructed by aligning the first camera position to the origin, and use the estimated camera poses to triangulate the scene points as in Section 4.4.

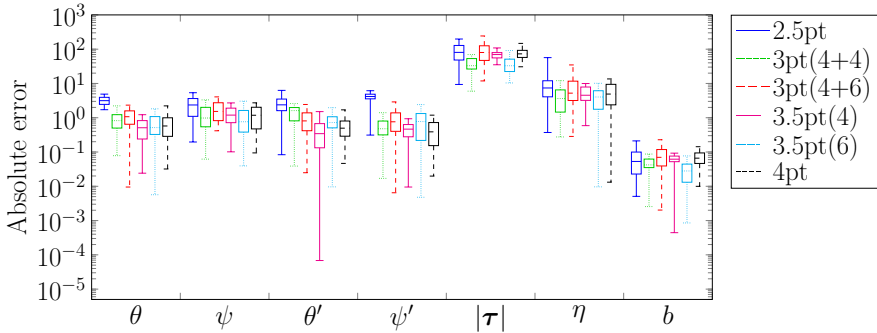


Figure 2: Errors before applying BA. The angles are measured in degrees, and the translation in pixels.

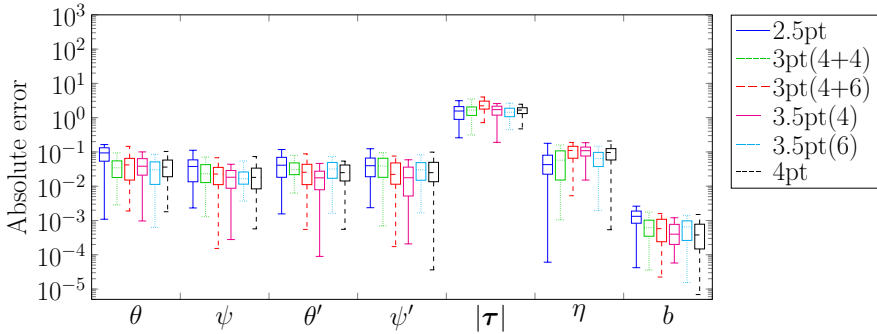


Figure 3: Errors after applying BA. The angles are measured in degrees, and the translation in pixels.

6.2 Impact of Pre-Processing Steps

In this section we work with synthetic data in order to have access to accurate ground truth data. We generate an image sequence from a high-resolution image, depicting a floor, which is the typical use case for the algorithm. This is done by constructing a path compatible with the general planar motion model, and project that part of the floor through the camera and extract the corresponding image. The resulting image is 400×400 pixels, and all cameras are set to a field of view of 90 degrees, with parameters $\psi = -2^\circ$, $\theta = -4^\circ$, $\psi' = 6^\circ$, $\theta' = 4^\circ$, $\tau = (0 \ 400)$, $\eta = 20^\circ$ and $b = 1$. In total, the image sequence consists of 20 images. Lastly, to simulate image noise, we add Gaussian noise with a standard deviation of five pixels, where the pixel depth allows 256 different

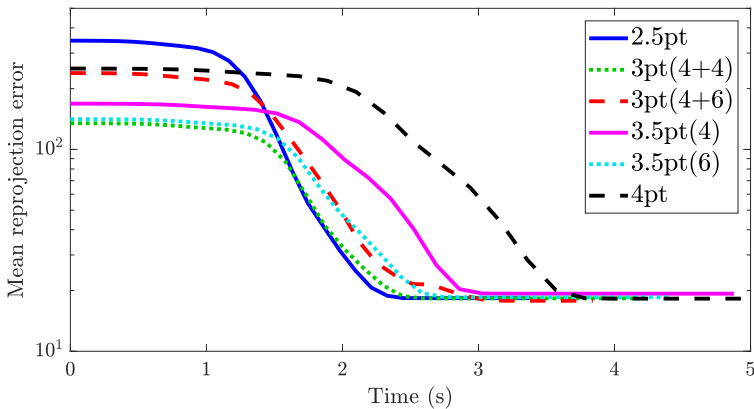


Figure 4: Mean reprojection error vs execution time (s) over 50 iterations.

intensities per channel.

In order to study the difference in accuracy for the constant parameters, we proceed by obtaining homographies as described in Section 6.1, using the minimal 2.5 point solver [31], four non-minimal solvers [27] and the DLT equations (4 point). The accuracy, over 50 iterations, is reported before BA, in Figure 2, and after BA, in Figure 3. In general, the overall performance of the solvers are almost equal; however, some tendencies are present. The minimal solver performs worse than the other before BA, but this deviation is smaller after BA, although present. One possible explanation is that the general planar motion model is enforced too early in the pipeline — in fact, since it is enforced between two consecutive image pairs only, it does not guarantee that the overhead tilt is constant throughout the entire sequence, and thus, in the presence of noise, the error propagates differently, compared to the other methods that partially (non-minimal) or completely (DLT) tune to the data.

Overall, the performance is acceptable after BA, regardless of how the homographies are obtained. Hence, the differentiating factors come down to convergence rates. For the same problem instances as in the previous section we also save the convergence history in terms of the mean reprojection error and the execution time in seconds. The results are shown in Figure 4. It is clear that the execution time for reaching convergence increase with the number of point correspondences required by the polynomial solvers. This suggests that one can make a trade-off between speed and accuracy when designing a planar motion compatible BA framework by choosing different solvers, in order to suit ones specific needs. Note, however, that the implementation used in this paper is a native



Figure 5: Images from the KITTI Visual Odometry / SLAM benchmark, Sequence 01 (left) and 03 (right). Since the algorithm is homography-based the images are cropped *a priori* in order to contain a significant portion of planar or near planar surface. Such an assumption is not valid on all sequences of the dataset, however, certain cases, such as the highway of Sequence 01 (left) is a good candidate. There are several examples where occlusions occur, such as the car in Sequence 03 (right). These situations typically occur at crossroads and turns. Image credit: KITTI dataset [9].

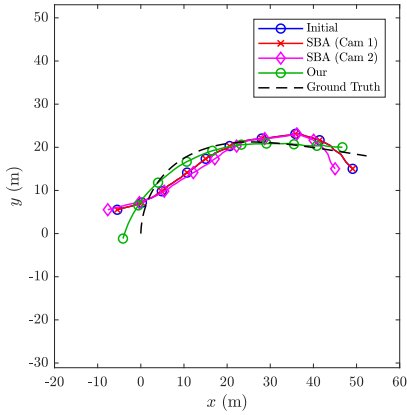
Matlab implementation, and that the absolute timings can be greatly improved by careful implementation; however, the relative execution time between the solvers will be similar.

6.3 Bundle Adjustment Comparison

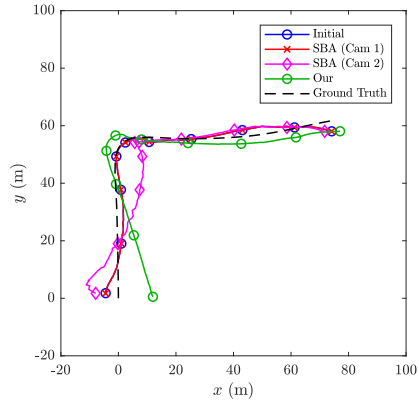
In this section we compare the qualitative difference between enforcing the general planar motion model versus the general unconstrained six degree of freedom model on a real dataset. Currently, there is not a good or well-established dataset compatible with the general planar motion model, and as a substitute, we use the KITTI Visual Odometry / SLAM benchmark [9]. Since many sequences or subsequences depict urban environments with paved roads, the general planar motion model can roughly be applied. In case of clear violation of the general planar motion model, we proceed to use only subsequences where the model is applicable. As we are only interested by the road in front of the vehicle, and not the sky and other objects by the roadside, we proceed to crop a part of the image prior to estimating the homography. An example of this is shown in Figure 5.

We use SBA [16] to enforce the general 6-DoF model from the initial trajectory obtained using the traditional 4-point DLT solver, and from the same trajectory our proposed BA algorithm is used. The same thresholds for absolute and relative errors, termination control and damping factors are used for both methods. Furthermore, we do not match features between the stereo views, in order to demonstrate that enforcing the model is enough to increase the overall performance. The results are shown in Figure 6.

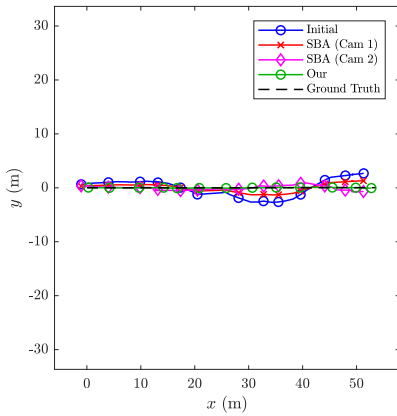
In most cases it is favourable to impose the proposed method compared to the general 6-DoF method, using SBA. Furthermore, note that irregularities that are present in the initial trajectory is often transferred to the solutions obtained by SBA, thus producing physically improbable solutions. These irregularities are rarely seen using the proposed method, which results in smooth realistic trajectories under general conditions, regardless



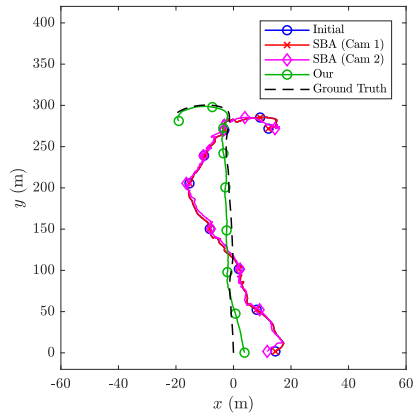
(a) Sequence 01 (60 images).



(b) Sequence 03 (200 images).



(c) Sequence 04 (40 images).



(d) Sequence 06 (330 images).

Figure 6: Estimated trajectories of subsequences of Sequence 01, 03, 04 and 06. In order to align the estimated paths with the ground truth, Procrustes analysis has been carried out. *N.B.* the different aspect ratio in (c), which is intentionally added in order to clearly visualise the difference.

of whether the initial solution contains irregularities or not.

In fact, it is interesting to see what happens in cases where the general planar motion model is violated. Such an instance occurs in Figure 6b depicting Sequence 03, and is due to the car approaching a crossroads, where a passing vehicle enters the field of view. The observed car, and the surroundings, are highly non-planar; one would, perhaps, expect such a clear violation to result in completely unreliable output, however, the only inconsistency in comparison to the ground truth, is that the resulting turn is too sharp, and the remaining path is consistent with the ground truth. This is not true for the general 6-DoF model, where several obvious inconsistencies are present.

7 Conclusion

In this paper a novel bundle adjustment method has been devised, which enforces the general planar motion model. We provide an efficient implementation scheme that exploits the sparse structure of the Jacobian, and, additionally, avoids recomputing unnecessary quantities, making it highly attractive for real-time computations.

The performance of different polynomial solvers are studied, in terms of both accuracy and speed, taking the entire bundle adjustment framework into account. We discuss how enforcing different polynomial constraints, through planar motion compatible homography solvers, in an early part of the bundle adjustment framework affect the end results. Furthermore, we discuss which trade-offs between speed and accuracy that can be made to suit ones specific priorities.

The proposed method has been tested on real data and was compared to state-of-the-art methods for sparse bundle adjustment, for which it performs well, and gives physically accurate solutions, despite some model assumptions not being fulfilled.

Acknowledgements

This work has been funded by the Swedish Research Council through grant no. 2015-05639 “Visual SLAM based on Planar Homographies”.

References

- [1] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski. “Building Rome in a Day”. In: *Communications of the ACM* 54.10 (Oct. 2011), pp. 105–112.

-
- [2] H. Bay, T. Tuytelaars, and L. Van Gool. “SURF: Speeded Up Robust Features”. In: *European Conference on Computer Vision (ECCV)*. Graz, Austria, May 2006, pp. 404–417.
- [3] M. Byröd and K. Åström. “Conjugate Gradient Bundle Adjustment”. In: *European Conference on Computer Vision (ECCV)*. Heraklion, Crete, Greece, Sept. 2010, pp. 114–127.
- [4] T. Chen and Y.-H. Liu. “A robust approach for structure from planar motion by stereo image sequences”. In: *Machine Vision and Applications (MVA) 17.3* (Aug. 2006), pp. 197–209.
- [5] A. J. Davison. “Real-Time Simultaneous Localisation and Mapping with a Single Camera”. In: *International Conference on Computer Vision (ICCV)*. Nice, France, Oct. 2003, pp. 1403–1410.
- [6] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. “MonoSLAM: Real-Time Single Camera SLAM”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 29.6 (June 2007), pp. 1052–1067.
- [7] A. Eriksson, J. Bastian, T. Chin, and M. Isaksson. “A Consensus-Based Framework for Distributed Bundle Adjustment”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA, June 2016, pp. 1754–1762.
- [8] J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, and M. Pollefeys. “Building Rome on a Cloudless Day”. In: *European Conference on Computer Vision (ECCV)*. Heraklion, Crete, Greece, Sept. 2010, pp. 368–381.
- [9] A. Geiger, P. Lenz, and R. Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Providence, RI, USA, June 2012, pp. 3354–3361.
- [10] H. Hajjdiab and R. Laganière. “Vision-based Multi-Robot Simultaneous Localization and Mapping”. In: *Canadian Conference on Computer and Robot Vision (CRV)*. London, ON, Canada, May 2004, pp. 155–162.
- [11] R. Hänsch, I. Drude, and O. Hellwich. “Modern Methods of Bundle Adjustment on the GPU”. In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Congress)*. Prague, Czech Republic, July 2016, pp. 43–50.
- [12] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
- [13] K. Konolige. “Sparse Sparse Bundle Adjustment”. In: *British Machine Vision Conference*. Aberystwyth, Wales, Aug. 2010.

- [14] B. Liang and N. Pears. “Visual Navigation using Planar Homographies”. In: *International Conference on Robotics and Automation (ICRA)*. Washington, DC, USA, May 2002, pp. 205–210.
- [15] M. Lourakis and A. Argyros. “Is Levenberg-Marquardt the Most Efficient Optimization Algorithm for Implementing Bundle Adjustment?.” In: *Proceedings of the Tenth IEEE International Conference on Computer Vision*. Vol. 2. Jan. 2005, pp. 1526–1531.
- [16] M. Lourakis and A. Argyros. “SBA: A Software Package for Generic Sparse Bundle Adjustment”. In: *ACM Trans. Math. Softw.* 36.1 (Mar. 2009), 2:1–2:30.
- [17] D. G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision (IJCV)* 60.2 (Nov. 2004), pp. 91–110.
- [18] D. Ortín and J. M. M. Montiel. “Indoor robot motion based on monocular images”. In: *Robotica* 19.3 (May 2001), pp. 331–342.
- [19] T. T. Pham, T.-J. Chin, J. Yu, and D. Suter. “The Random Cluster Model for Robust Geometric Fitting”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 36.8 (Aug. 2014), pp. 1658–1671.
- [20] D. Scaramuzza. “1-Point-RANSAC Structure from Motion for Vehicle-Mounted Cameras by Exploiting Non-holonomic Constraints”. In: *International Journal of Computer Vision (IJCV)* 95.1 (Oct. 2011), pp. 74–85.
- [21] D. Scaramuzza. “Performance Evaluation of 1-Point-RANSAC Visual Odometry”. In: *Journal of Field Robotics (JFR)* 28.5 (Sept. 2011), pp. 792–811.
- [22] N. Snavely, S. M. Seitz, and R. Szeliski. “Modeling the World from Internet Photo Collections”. In: *International Journal of Computer Vision (IJCV)* 80.2 (Nov. 2008), pp. 189–210.
- [23] P. Sturm. “A Historical Survey of Geometric Computer Vision”. In: *Proceedings of the 14th International Conference on Computer Analysis of Images and Patterns (CAIP)*. Vol. 6854. Lecture Notes in Computer Science. Seville, Spain: Springer-Verlag, Aug. 2011, pp. 1–8. ISBN: 978-3-642-23671-6.
- [24] R. Szeliski. *Computer Vision: Applications and Algorithms*. London, England, UK: Springer-Verlag, 2011.
- [25] P. H. S. Torr and A. Zisserman. “MLESAC: A New Robust Estimator with Application to Estimating Image Geometry”. In: *Computer Vision and Image Understanding (CVIU)* 78.1 (2000), pp. 138–156.
- [26] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. “Bundle Adjustment — A Modern Synthesis”. In: *International Workshop on Vision Algorithms — Vision Algorithms: Theory and Practice*. Corfu, Greece, Sept. 1999, pp. 298–372.

-
- [27] M. Valtonen Örnthag. “Fast Non-minimal Solvers for Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 40–51.
- [28] M. Valtonen Örnthag and A. Heyden. “Generalisation of Parameter Recovery in Binocular Vision for a Planar Scene”. In: *International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI)*. Montréal, Canada, May 2018, pp. 37–42.
- [29] M. Valtonen Örnthag and A. Heyden. “Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Funchal, Madeira, Portugal, Jan. 2018, pp. 568–575.
- [30] M. Valtonen Örnthag and M. Wadenbäck. “Planar Motion Bundle Adjustment”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Prague, Czech Republic, Feb. 2019, pp. 24–31.
- [31] M. Wadenbäck, K. Åström, and A. Heyden. “Recovering Planar Motion from Homographies Obtained using a 2.5-Point Solver for a Polynomial System”. In: *International Conference on Image Processing (ICIP)*. Phoenix, AZ, USA, Sept. 2016, pp. 2966–2970.
- [32] M. Wadenbäck and A. Heyden. “Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Lisbon, Portugal, Jan. 2014, pp. 635–639.
- [33] M. Wadenbäck and A. Heyden. “Planar Motion and Hand-Eye Calibration using Inter-Image Homographies from a Planar Scene”. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*. Barcelona, Spain, 2013, pp. 164–168.
- [34] C. Wiles and M. Brady. “Closing the Loop on Multiple Motions”. In: *Proceedings of the Fifth IEEE International Conference on Computer Vision (ICCV)*. Cambridge, MA, USA: IEEE Computer Society, June 1995, pp. 308–313. ISBN: 0-8186-7042-8.
- [35] C. Wiles and M. Brady. “Ground Plane Motion Camera Models”. In: *Proceedings of the 4th European Conference on Computer Vision (ECCV)*. Vol. 1065. Lecture Notes in Computer Science. Cambridge, England, UK: Springer-Verlag, Apr. 1996, pp. 238–247. ISBN: 3-540-61123-1.

- [36] C. Wu, S. Agarwal, B. Curless, and S. M. Seitz. “Multicore Bundle Adjustment”. In: *Computer Vision and Pattern Recognition (CVPR)*. Providence, RI, USA, June 2011, pp. 3057–3064.
- [37] C. Zach. “Robust Bundle Adjustment Revisited”. In: *European Conference on Computer Vision (ECCV)*. Zurich, Switzerland, Sept. 2014, pp. 772–787.
- [38] R. Zhang, S. Zhu, T. Fang, and L. Quan. “Distributed Very Large Scale Bundle Adjustment by Global Camera Consensus”. In: *International Conference on Computer Vision (ICCV)*. Venice, Italy, Oct. 2017, pp. 29–38.
- [39] J. Zienkiewicz and A. J. Davison. “Extrinsics Autocalibration for Dense Planar Visual Odometry”. In: *Journal of Field Robotics (JFR)* 32.5 (Aug. 2015), pp. 803–825.

Minimal Solvers for Indoor UAV Positioning

MARCUS VALTONEN ÖRNHAG¹, PATRIK PERSSON¹, MÅRTEN WADENBÄCK²,
KALLE ÅSTRÖM¹ AND ANDERS HEYDEN¹

¹*Centre for Mathematical Sciences, Lund University*

²*Department of Electrical Engineering, Linköping University*

Abstract: In this paper we consider a collection of relative pose problems which arise naturally in applications for visual indoor navigation using unmanned aerial vehicles (UAVs). We focus on cases where additional information from an onboard IMU is available and thus provides a partial extrinsic calibration through the gravitational vector. The solvers are designed for a partially calibrated camera, for a variety of realistic indoor scenarios, which makes it possible to navigate using images of the ground floor. Current state-of-the-art solvers use more general assumptions, such as using arbitrary planar structures; however, these solvers do not yield adequate reconstructions for real scenes, nor do they perform fast enough to be incorporated in real-time systems.

We show that the proposed solvers enjoy better numerical stability, are faster, and require fewer point correspondences, compared to state-of-the-art approaches. These properties are vital components for robust navigation in real-time systems, and we demonstrate on both synthetic and real data that our method outperforms other solvers, and yields superior motion estimation¹.

1 Introduction

One of the lessons which have been learned in computer vision is the importance of leveraging prior knowledge pertaining to the specific vision task at hand. For geometrical computer vision problems, this often means introducing constraints encoding the prior knowledge already at the modeling stage. Successfully making use of such prior information in a vision system can have numerous direct benefits, such as improved robustness, better accuracy, and faster performance. However, constraining the solution space will in many cases come at the cost of requiring significantly more complex algorithms. A familiar illustration of this is the computation of fundamental or essential matrices in epipolar geometry; the simplicity of the seven-point [10] or eight-point [29] algorithms stands in stark contrast to the complexity of the five-point algorithm [32].

¹Code available at: https://github.com/marcusvaltonen/minimal_indoor_uav.

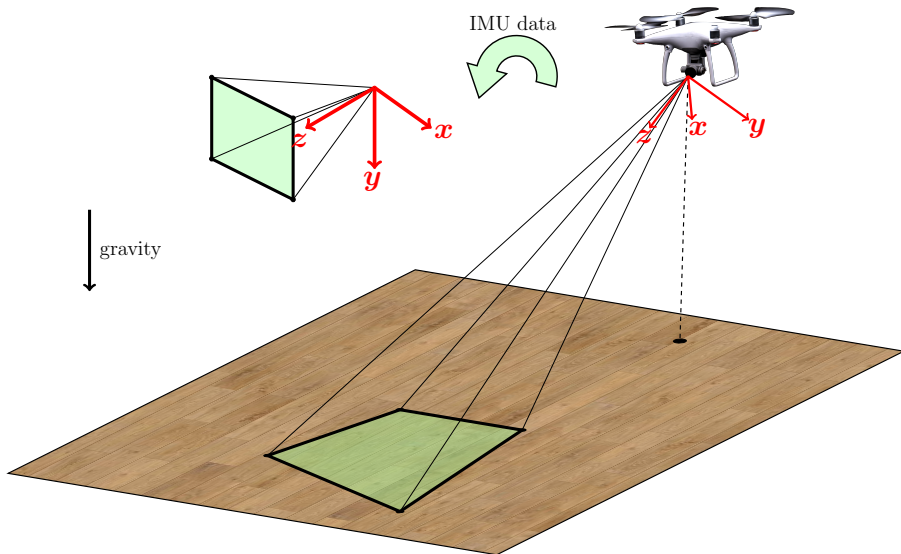


Figure 1: The pitch and roll angles can be estimated from the IMU data, leaving the unknown yaw angle to be determined.

In particular over the past two decades, the embracement of approaches based on algebraic geometry has allowed many constrained geometrical problems to be solved with a minimal (or near-minimal) amount of data. This has notably been the case for notorious problems such as five-point essential matrix estimation [28, 32, 38] or optimal three-view triangulation [5, 20, 39], but also for pose estimation problems involving some kind of geometric distortion such as radial distortion [11, 14, 15, 18, 24, 25, 34, 40] or refractive distortion in underwater problems [8, 33]. These successes have been enabled chiefly thanks to improvements in the computational algorithms themselves, but to a lesser extent also by improvements in hardware which have made it feasible to solve increasingly large systems.

Inspired primarily by recent papers by Saurer *et al.* [35] and by Ding *et al.* [6], we aim to create a homography based indoor positioning system for UAVs, that utilizes data from an onboard IMU. One benefit of using the IMU data is that two degrees of freedom, the roll and the pitch, are removed from the positioning problem, only leaving a single rotational parameter—the yaw angle—to determine. An illustration of the envisaged situation is provided in Figure 1.

Our main contributions are fourfold:

1. We incorporate the known IMU data for positioning using the ground floor and a partially calibrated camera with unknown focal length,
2. The resulting solvers are orders of magnitude faster than the current state-of-the-art,
3. We demonstrate through numerical experiments that our solvers are better than or on par with existing solvers with respect to accuracy and runtime, and
4. We run the solvers in real-time on a UAV system, demonstrating that the derived solvers are feasible for practical real-world situations.

2 Related Work

Pose estimation with a known direction For many practical pose estimation problems, there are simple ways which allow the extraction of one particular direction, thus giving some of the pose parameters for free. As shown by Kalantari *et al.* [12], this could happen *e.g.* by detection in the image of a horizon line or vanishing points, or through external sources such as the gravitational vector from an IMU. Provided a known direction and 2D–3D correspondences, Kukulova *et al.* gave a closed-form solution to the absolute pose problem for a calibrated or partially calibrated (unknown focal length) camera [17].

Homography-based relative pose, where the plane normal is taken as the known direction, has also been considered in the literature. Minimal solvers for the calibrated case, including using the ground floor (2pt) or unknown vertical plane (2.5pt) and an arbitrary plane (3pt), were treated in [36] and [35]. In Ding *et al.* the 3pt case was considered, and also included some partially calibrated cases [6], *e.g.* with either one or two unknown focal lengths.

Estimation of focal lengths Hartley considered the problem of estimating the essential matrix and focal lengths from point correspondences, under the assumption that all other intrinsic camera parameters were known [9]. The method is based on the eight-point algorithm [29] followed by a series of algebraic manipulations of the fundamental matrix to extract the focal lengths. Under the same assumptions, Bougnoux derived a closed-form formula for computing the focal lengths [1]. With the introduction of polynomial solvers for various problems in computer vision, people have proposed solvers which give directly the essential matrix and the focal lengths, without the necessity to go via the fundamental matrix. One of the early such algorithms was presented by Li [27], who solved the problem with six point correspondences (the minimal case). A recent paper

by Kukulova *et al.* [19] contains, as an application, a state-of-the-art solver to the six-point essential matrix and focal length estimation problem. In the experiments, we will compare our proposed solvers against this last solver by Kukulova *et al.*

Solving multivariate polynomial equations Solving systems of multivariate polynomial equations numerically is usually done through the *action matrix method* [4, 16, 30]. This works by first expanding the equations into an *elimination template*, which is then reduced to an eigenvalue problem, where the matrix in question is known as the *action matrix*. The speed of the solver depends on (i) how fast the coefficients of the polynomials can be calculated, (ii) the size of the elimination template, and (iii) the number of solutions. For most cases the size of the elimination template is the dominant factor. The numerical stability does also depend on the design, although in a more complicated way.

Finding a reasonably sized and numerically stable elimination template for a given problem is a highly non-trivial task, although it has been greatly simplified through the introduction of automatic generators. One early such automatic generator by Kukulova *et al.* [16] used a heuristic approach to expand the system of equations until a valid elimination template was obtained, and then proceeded with successively removing one row at a time from the elimination template to prune redundancies. Larsson *et al.* [22] exploited the inherent relations between the equations to directly compute a set of monomials with which to expand the equations in order to yield a valid elimination template.

In addition to elimination template generation, there have been several improvements to the numerical accuracy, numerical stability, and speed of polynomial solvers. In [3] and [4], Byröd *et al.* proposed performing the reduction to the action matrix by means of either a QR decomposition or a singular value decomposition, together with an adaptive scheme for pruning columns which are deemed unnecessary. Naroditsky and Daniilidis suggested elimination template trimming based on certain algebraic conditions [31]. However, as it turns out, there are situations where excessive reduction in template size comes at the cost of inferior numerical stability [2]. Kuang and Åström suggested evaluating many reduction schemes for a large number of random problem instances in order to ensure good numerical properties of the resulting template [13].

3 Incorporating the IMU Data

We follow the approach used in [6]. A general homography \mathbf{H} has eight degrees of freedom, fulfilling the Direct Linear Transform (DLT) constraint,

$$\mathbf{x}_2 \sim \mathbf{H}\mathbf{x}_1, \tag{1}$$

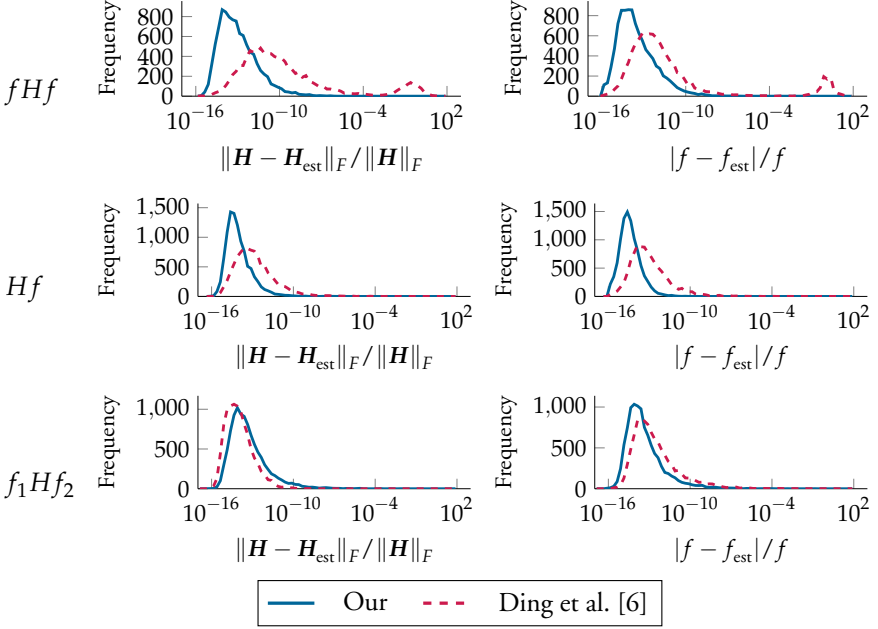


Figure 2: Error histogram for 10,000 randomly generated problem instances for the partially calibrated cases. Top to bottom: fHf , Hf , f_1Hf_2 .

for two point correspondences $\mathbf{x}_1 \leftrightarrow \mathbf{x}_2$ on a common scene plane. The relation (1) yields two linearly independent equations, hence a minimum of four point correspondences are necessary in order to estimate the homography. In the calibrated case, with intrinsic parameters encoded in the calibration matrices \mathbf{K}_1 and \mathbf{K}_2 , respectively, one obtains

$$\mathbf{H}_{\text{euc}} \sim \mathbf{K}_2 \mathbf{H} \mathbf{K}_1^{-1}, \quad (2)$$

where the Euclidean homography \mathbf{H}_{euc} can be written as

$$\mathbf{H}_{\text{euc}} \sim \mathbf{R} + \frac{1}{d} \mathbf{t} \mathbf{n}^T, \quad (3)$$

where \mathbf{R} is the relative rotation between the views, \mathbf{n} is the plane normal, and \mathbf{t} is the relative translation. The depth parameter d is the distance from the first camera center to the scene plane.

We consider the case of a common reference direction, assumed to be aligned with the gravitational direction, which eliminates two degrees of freedom, see Figure 1. There-

fore, after a suitable change of coordinates, we may assume that

$$\mathbf{H}_y \sim \mathbf{R}_y + \frac{1}{d'} \mathbf{t}' \mathbf{n}'^T, \quad (4)$$

where \mathbf{R}_y is a rotation about the y -axis (gravitational direction). Hence, the DLT equations (1) can be written as

$$\mathbf{R}_2^T \mathbf{K}_2^{-1} \mathbf{x}_2 \sim \mathbf{H}_y \mathbf{R}_1^T \mathbf{K}_1^{-1} \mathbf{x}_1. \quad (5)$$

The known matrices \mathbf{R}_1 and \mathbf{R}_2 are given by the IMU data, and consist of the pitch and roll angles for the first and second camera positions.

The relation between the original homography \mathbf{H} , from (1), and \mathbf{H}_y , from (4), is thus given by

$$\mathbf{H}_y \sim \mathbf{R}_2^T \mathbf{K}_2^{-1} \mathbf{H} \mathbf{K}_1 \mathbf{R}_1. \quad (6)$$

Furthermore, the relative rotation \mathbf{R} and the relative translation \mathbf{t} , are given by

$$\mathbf{R} = \mathbf{R}_2 \mathbf{R}_y \mathbf{R}_1^T \quad \text{and} \quad \frac{\mathbf{t}}{d'} = \mathbf{R}_2 \frac{\mathbf{t}'}{d'}, \quad (7)$$

with the modified plane normal $\mathbf{n}' = \mathbf{R}_1 \mathbf{n}$.

3.1 Navigation Using the Ground Plane

By introducing the auxiliary variables

$$\mathbf{y}_i = \mathbf{R}_j^T \mathbf{K}_j^{-1} \mathbf{x}_i, \quad (8)$$

one may reduce (5) to $\mathbf{y}_2 \sim \mathbf{H}_y \mathbf{y}_1$. For an arbitrary plane normal \mathbf{n}' , the matrix \mathbf{H}_y , defined as in (4), has 6 degrees of freedom (DoF). However, if we constrain ourselves to navigating using the ground plane, then the plane normal is uniquely defined (up to scale), with $\mathbf{n}' = (0, 1, 0)^T$. Note, that we can only recover the translation up to scale, hence we may assume that the depth $d' = 1$. Parameterizing the rotation matrix as

$$\mathbf{R}_y = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, \quad (9)$$

we may write

$$\mathbf{H}_y = \begin{bmatrix} h_1 & h_3 & h_2 \\ 0 & h_4 & 0 \\ -h_2 & h_5 & h_1 \end{bmatrix}, \quad (10)$$

where \mathbf{R} and \mathbf{t} can be extracted directly through the entries h_i , given by

$$\mathbf{R} = \begin{bmatrix} h_1 & 0 & h_2 \\ 0 & 1 & 0 \\ -h_2 & 0 & h_1 \end{bmatrix} \text{ and } \mathbf{t} = \begin{bmatrix} h_3 \\ h_4 - 1 \\ h_5 \end{bmatrix}, \quad (11)$$

where the trigonometric constraint $h_1^2 + h_2^2 = 1$ must be enforced to get a valid rotation matrix.

3.2 The Calibrated Case

In [35], the authors construct a minimal solver for the calibrated case, with 4 DoF (three translation components, and the unknown rotation about the ground floor normal). As a consequence, one only needs two point correspondences, which give rise to four linearly independent equations. By parameterizing \mathbf{H}_y as in (4), it is possible to form the linear system²

$$\mathbf{A}\mathbf{h} = \mathbf{0}, \quad (12)$$

where \mathbf{A} is a 4×5 matrix and \mathbf{h} contains the h_i . For non-degenerate configurations, the matrix \mathbf{A} has a one-dimensional nullspace, which can be obtained using singular value decomposition (SVD). One may fix the scale of \mathbf{H}_y , in order to acquire valid rotation parameters. This is achieved by using the trigonometric constraint introduced through the parameterization of the rotation matrix, $h_1^2 + h_2^2 = 1$, as in (11). This approach leaves two possible solutions [35].

4 Partially Calibrated Cases for Ground Plane Navigation

We will extend the ground plane solver to three partially calibrated cases, also considered in [6].

4.1 Equal and Unknown Focal Length (fHf , 2.5-point)

Parameterize the inverse of the unknown calibration matrix as $\mathbf{K}^{-1} = \text{diag}(1, 1, w)$, and consider the rectified points (8), which now depend linearly on the unknown parameter w . Parameterizing \mathbf{H}_y as in (10), it is clear that the equations obtained from (5)

²In [35] the camera is aligned with the z -axis, but the same procedure can be repeated using the y -axis instead.

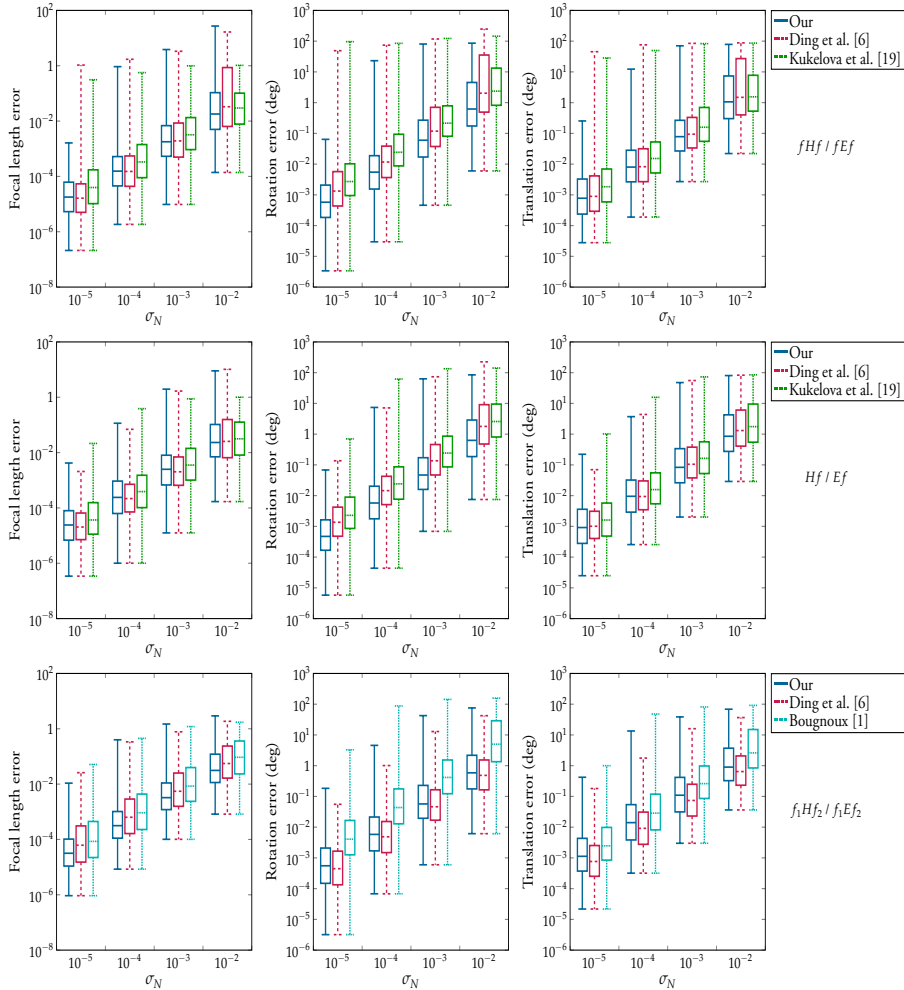


Figure 3: Noise sensitivity comparison for Gaussian noise with standard deviation σ_N . For each noise level 1,000 random problem instances were generated.

are quadratic in h_i , $i = 1, \dots, 5$ and w . Using 2.5-point correspondences, *i.e.* using 3-point correspondences and discarding one of the DLT equations, together with the constraint $h_1^2 + h_2^2 = 1$, gives six equations and six unknowns. This system of equations has infinitely many solutions, if we allow $w = 0$. Such solutions, however, do not yield geometrically meaningful reconstructions, and should therefore be excluded. This can

be achieved using saturation, through the method suggested in [23].

To generate a polynomial solver, we use the automatic generator proposed in [21], from which we have 14 solutions, and an elimination template of size 10×17 . Exploiting symmetries, the solutions can be obtained by solving a 7×7 eigenvalue problem.

4.2 One Unknown Focal Length (Hf , 2.5-point)

We may use the same approach as in the previous case, but may pre-compute $\mathbf{y}_1 = \mathbf{R}_1^T \mathbf{K}_1^{-1} \mathbf{x}_i$, to reduce the number of unknown coefficients in the elimination template. Due to this approach, we are able to reduce the eigenvalue problem to a 4×4 matrix³, despite having the same amount of solutions and elimination template

4.3 Different and Unknown Focal Lengths ($f_1 H f_2$, 3-point)

Using the same approach as previously for this case resulted in a large elimination template, and in order to reduce the template size we opted for a different approach. Similar to [6] we parameterize the nullspace of the homography (1). In general, three point correspondences yield six linearly independent equations, hence the corresponding nullspace is of dimension 3 and can be parameterized using

$$\mathbf{H} = \alpha_0 \mathbf{H}_0 + \alpha_1 \mathbf{H}_1 + \alpha_2 \mathbf{H}_2. \quad (13)$$

We may fix the scale by letting $\alpha_0 = 1$. As in the previous case, we may parameterize the calibration matrices $\mathbf{K}_1 = \text{diag}(1, 1, w_1)$ and $\mathbf{K}_2^{-1} = \text{diag}(1, 1, w_2)$. Inserting this into (6) yields a parameterization of \mathbf{H}_y in the unknowns $\alpha_1, \alpha_2, w_1, w_2$ of cubic degree.

Finally, given

$$\mathbf{H}_y = \begin{bmatrix} \hat{h}_1 & \hat{h}_2 & \hat{h}_3 \\ \hat{h}_4 & \hat{h}_5 & \hat{h}_6 \\ \hat{h}_7 & \hat{h}_8 & \hat{h}_9 \end{bmatrix}, \quad (14)$$

and comparing to (10), we obtain the four equations

$$\hat{h}_1 - \hat{h}_9 = 0, \quad \hat{h}_3 + \hat{h}_7 = 0, \quad \hat{h}_4 = \hat{h}_6 = 0. \quad (15)$$

In conclusion, we get four cubic equations in four unknowns. Using the automatic generator, and carefully selecting a basis [26], we are able to find an elimination template of 10×15 , with five solutions, which are obtained by solving a 5×5 eigenvalue problem.

³In fact, one may use the quartic formula for root finding instead, to gain a bit of extra performance.

4.4 Synthetic Experiments

Numerical Stability and Noise Sensitivity

In this section the proposed solvers are tested on synthetic data. We generate points on the ground plane $y = 0$ and compare the proposed solvers with the corresponding solvers by Ding *et al.* [6]⁴. We only compare these solvers as they both estimate homographies, as compared to the methods by Kukelova *et al.* [19] and Bougnoux [1], which estimates the fundamental matrix. Furthermore, with noise-free data, the latter methods degenerate for planar correspondences.

We generate all solvers in C++, and measure the error for the obtained homography and the focal length(s) over 10,000 randomly generated problem instances. The problem instances are obtained by generating scene points $[x_i, 0, z_i, 1]^T$, where x_i, z_i are chosen from a random distribution with zero mean and unit variance, then projected through cameras $P_i = [R_i | t_i]$, where $R_i = R_{\text{imu}}^{(i)} R_z^{(i)}$. The translation component was generated at random with zero mean and unit variance. When measuring the error, the homographies are normalized to have last entry equal to 1. To increase numerical stability, we use the normalization technique proposed by Ding *et al.* [6]. The results are shown in Figure 2. For the partially calibrated cases with unknown and equal focal length (fHf), and one unknown focal length (Hf) the proposed solvers are significantly more stable. Furthermore, in the case of unknown and (possibly) different focal lengths (f_1Hf_2) the method by Ding *et al.* [6] uses 4 point correspondences, hence the homography estimation is a regular 4-point DLT system obtained using SVD, with known stable properties.

5 Experiments

To mimic real data, we distort the image point correspondences by adding zero mean Gaussian noise with a standard deviation σ_N , and vary the noise level. To be able to compare the same quantity, the obtained homographies and fundamental matrices, are decomposed to relative translation and orientation. We proceed as in the previous experiment by generating ground truth data, but also include non-planar point correspondences for the 6-point and 7-point methods, in order for them to not degenerate.

⁴Code shared by the authors.

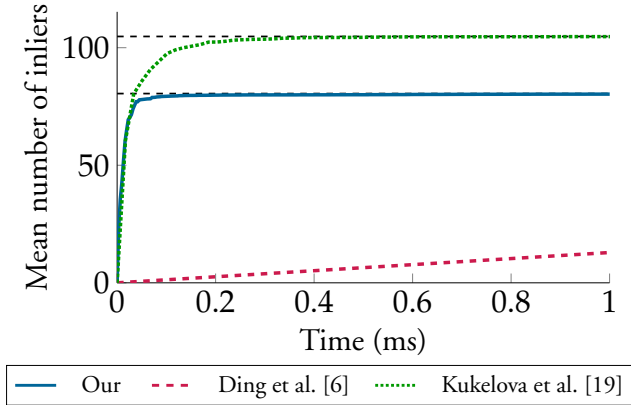


Figure 4: Number of RANSAC inliers vs. time for the different methods. The horizontal dashed lines are the theoretical maximal limits for the homography based methods (lower) and the method by Kukelova *et al.* (upper).

We define the errors as in [6, 35], namely

$$\begin{aligned}
 e_R &= \arccos\left(\frac{\text{tr}(\mathbf{R}_{\text{GT}}\mathbf{R}_{\text{est}}^T) - 1}{2}\right), \\
 e_t &= \arccos\left(\frac{\mathbf{t}_{\text{GT}}^T \mathbf{t}_{\text{est}}}{\|\mathbf{t}_{\text{GT}}\| \|\mathbf{t}_{\text{est}}\|}\right), \\
 e_f &= \frac{|f_{\text{GT}} - f_{\text{est}}|}{f_{\text{GT}}}.
 \end{aligned} \tag{16}$$

In Figure 3 we show the results for the noise sensitivity experiments. We note that the proposed solvers perform consistent to SOTA solvers in all cases. We emphasize, however, that, for the synthetic data experiments, the main benefit of using our method is the execution time, which we show in the next section.

Speed Evaluation

The experiments were run on a laptop with an Intel Core i5-6200U 2.30GHz CPU using C++ implementations based on the Eigen linear algebra library [7]. The same optimization flags were used for all solvers. The mean runtime for one hypothesis estimation is shown in Table 1. We note that the proposed methods are faster than [19], which is partly due to the number of operations required to compute the coefficients

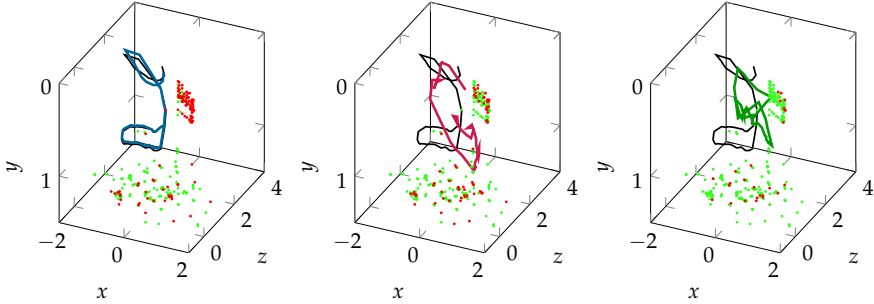


Figure 5: Drone experiment ground truth trajectories (black) for the *indoor* sequence, and the estimated trajectories for the different solvers. Green dots indicate inliers (from at least one frame) and red dots denote outliers (all frames). *Left to right*: Our, Ding *et al.* [6] and Kukulova *et al.* [19].

in the elimination template is significantly less in our case, as well as the corresponding eigenvalue problems being smaller. Furthermore, we note a speed-up of more than $75\times$ for our fHf solver compared to the SOTA solver by Ding *et al.* [6]. Furthermore, we argue that this is the most relevant case for indoor UAV navigation, as most consumer-grade UAVs have fixed focal length; however, when considering more than one drone, the other cases are relevant. For the case of Hf , one could have a calibrated camera on one of the UAVs, as a reference, thus not having to calibrate all remaining UAVs, potentially eliminating exhaustive calibration procedures. For the f_1Hf_2 case we may consider two drones (with possibly different focal lengths) covering the same area, *e.g.* drone swarms [37], or drones with varifocal optics.

Apart from being significantly faster than the SOTA solvers, our approach benefits from requiring one less point correspondence. In practice, when one uses a robust framework, such as RANSAC, the number of iterations required depends on the number of points to select, hence the practical speed-up is greater than $75\times$, see Table 2.

Furthermore, the 2.5-point and 3.5-point methods both benefit from being able to do a consistency check of the putative solutions, by using the third or fourth point correspondence, respectively, whereas other methods need to estimate the hypothesis every iteration. This optimization step was implemented for the fHf solvers, and we compare the number of inliers vs total execution time, see Figure 4, in a complete RANSAC framework. In this experiment 100 points were generated on the ground plane, as well as 30 non-planar correspondences, in order to simulate a possible real-life scenario. Noise was added, and 20 % of the points were scrambled to simulate outliers. Note, that the method by Kukulova *et al.* can make use of all points, including the non-planar correspondences, hence the theoretical maximal limit is higher than for the homography

based methods. As is evident, our method is superior in terms of reaching its theoretical maximal limit fastest, and significantly faster than the SOTA solver [6].

Table 1: Mean execution time for 10,000 randomly generated problems in C++. All solvers were implemented using Eigen [7] and compiled in g++ with the -O3 optimization flag.

Case	Author	Execution time (μ s)
fHf	Our	14
	Ding <i>et al.</i> [6]	1052
fEf	Kukelova <i>et al.</i> [19]	103
Hf	Our	5
	Ding <i>et al.</i> [6]	124
Ef	Kukelova <i>et al.</i> [19]	25
f_1Hf_2	Our	9
	Ding <i>et al.</i> [6]	47
f_1Ef_2	Bougnoux [1]	27

Table 2: Mean execution time for reaching at least 95 % inlier ratio in a RANSAC loop, based on 100 randomly generated problems in C++.

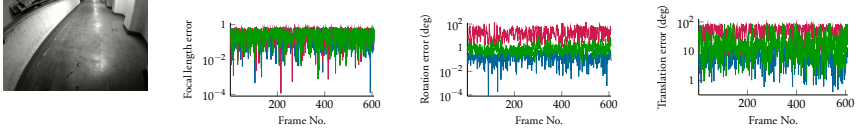
Case	Author	Execution time (ms)
fHf	Our	0.035
	Ding <i>et al.</i> [6]	19.32
fEf	Kukelova <i>et al.</i> [19]	0.118

5.1 Real Data

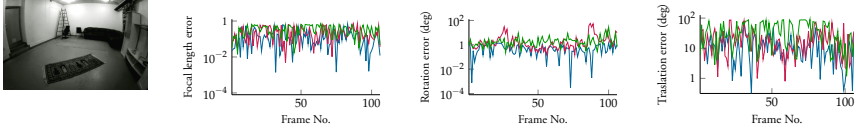
The real data was captured using a monochrome global shutter camera (OV9281) with resolution 480×640 and an inertial measurement unit (MPU-9250). The extracted feature locations were undistorted to remove fish-eye effects. Ground truth and pairwise feature matches were generated by a simultaneous localization and mapping system, where both the re-projection and IMU error were minimized; this is in order to create a globally consistent solution in metric scale. No assumptions about the scene structure were made, which lead to some matches not belonging to the ground plane, resulting in natural outliers, for the proposed method.

The data consists of both indoor and outdoor sequences containing mostly planar surfaces. The dataset contains shorter sequences (containing fewer than 50 images), as well as longer ones (containing more than 600 images), with varying kinds of motions.

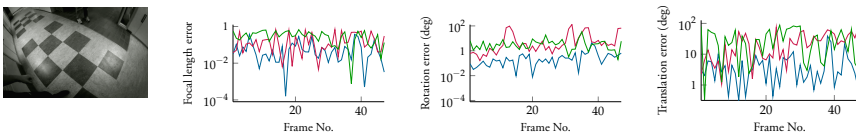
Basement (610 frames, 2978 keypoints)



Carpet (107 frames, 400 keypoints)



Indoor (48 frames, 184 keypoints)



Outdoor (601 frames, 3659 keypoints)

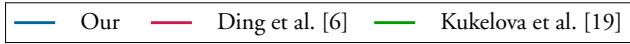
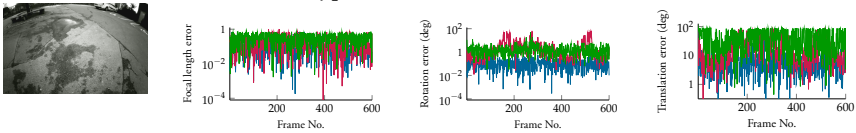


Figure 6: Drone experiment: (Left to right) Example image of the scene from the UAV. Focal length error, rotation error, translation error (as defined in (16)).

Examples of input images can be seen in the left-most column of Figure 6. Here, we also present the focal length, rotation and translation errors compared to the ground truth data for all frames. The estimated poses were obtained using 200 RANSAC iterations per method and frame for the equal and unknown focal length solvers.

To visualize the qualitative difference between the methods, we show the estimated trajectories for the *indoor* sequence in Figure 5. The initial pose was synchronized with the ground truth pose and the length of the translations scaled to match the ground truth translation vector for each frame. From here we converted the relative poses to the absolute poses and display the estimated trajectories together with the ground truth trajectories individually.

We also include green dots indicating feature points that have been matched (at least once) as an inlier in the sequence. The red dots are feature points that have consistently

been rejected as outliers. For the proposed method a red cluster of rejected outliers is clearly visible, and these are all feature point of a door, hence do not belong to the ground plane. It is interesting to see that the method by Ding *et al.* [6] uses these points as well when trying to estimate the relative pose. This, of course, is valid given the more general assumption used, by allowing arbitrary plane normals; however, it is readily seen from the estimated trajectory alone, that this approach is not robust enough for real-life applications.

6 Conclusions

In this paper we have presented new minimal solvers to perform homography based indoor positioning. The solvers can utilize IMU data to obtain a partial extrinsic calibration. We have evaluated the solvers on both synthetic and real world data, where we have shown that they are robust to both Gaussian noise, as well as noise present in real world data. Comparing with the current state of the art, our solvers are on par with SOTA [6] on synthetic data and outperforms them on real world data, where the solvers can be seen to follow ground truth more accurately. However, the main benefit of the solvers compared to the current state of the art is the execution speed, which is several orders of magnitude faster. In addition to this the solvers require fewer points compared to their counterparts. These attributes make the solvers viable for real-time systems, such as UAVs, requiring many RANSAC iterations to ensure robustness and therefore fast execution to meet the short time budget per frame.

Acknowledgements

We thank Ding *et al.* for providing their implementations. This work was partially supported by the Swedish Research Council (grant no. 2015-05639), the strategic research projects ELLIIT and eSENCE, the Swedish Foundation for Strategic Research project, Semantic Mapping and Visual Navigation for Smart Robots (grant no. RIT15-0038), and Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

References

- [1] S. Bougnoux. “From Projective to Euclidean Space Under any Practical Situation, a Criticism of Self-Calibration”. In: *International Conference on Computer Vision (ICCV)*. 1998, pp. 790–798.

- [2] M. Byröd. “Numerical Methods for Geometric Vision: From Minimal to Large Scale Problems”. PhD thesis. Lund University, 2010.
- [3] M. Byröd, K. Josephson, and K. Åström. “A Column-Pivoting Based Strategy for Monomial Ordering in Numerical Gröbner Basis Calculations”. In: *Proceedings of the 10th European Conference on Computer Vision (ECCV)*. Vol. 5305. Lecture Notes in Computer Science. Marseille, France: Springer-Verlag, Oct. 2008, pp. 130–143. ISBN: 978-3-540-88692-1.
- [4] M. Byröd, K. Josephson, and K. Åström. “Fast and Stable Polynomial Equation Solving and Its Application to Computer Vision”. In: *International Journal of Computer Vision* 84.3 (Sept. 2009), pp. 237–256.
- [5] M. Byröd, K. Josephson, and K. Åström. “Fast Optimal Three View Triangulation”. In: *Proceedings of the 8th Asian Conference on Computer Vision (ACCV)*. Tokyo, Japan, 2007, pp. 549–559.
- [6] Y. Ding, J. Yang, J. Ponce, and H. Kong. “An Efficient Solution to the Homography-Based Relative Pose Problem With a Common Reference Direction”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2019, pp. 1655–1664.
- [7] G. Guennebaud, B. Jacob, et al. *Eigen v3*. <http://eigen.tuxfamily.org>. 2010.
- [8] S. Haner and K. Åström. “Absolute Pose for Cameras Under Flat Refractive Interfaces”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA, June 2015, pp. 1428–1436.
- [9] R. I. Hartley. “Estimation of relative camera positions for uncalibrated cameras”. In: *European Conference on Computer Vision (ECCV)*. Santa Margherita Ligure, Italy, May 1992, pp. 579–587.
- [10] R. I. Hartley. “Projective Reconstruction and Invariants from Multiple Images”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16.10 (Oct. 1994), pp. 1036–1041.
- [11] K. Josephson and M. Byröd. “Pose Estimation with Radial Distortion and Unknown Focal Length”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Miami, FL, USA: IEEE Computer Society, June 2009, pp. 2419–2426. ISBN: 978-1-4244-3991-1.
- [12] M. Kalantari, A. Hashemi, F. Jung, and J.-P. Guedon. “A New Solution to the Relative Orientation Problem Using Only 3 Points and the Vertical Direction”. In: *Journal of Mathematical Imaging and Vision* 39 (Mar. 2011), pp. 259–268.

-
- [13] Y. Kuang and K. Åström. “Numerically Stable Optimization of Polynomial Solvers for Minimal Problems”. In: *European Conference on Computer Vision (ECCV)*. Florence, Italy, 2012, pp. 100–113.
- [14] Y. Kuang, J. E. Solem, F. Kahl, and K. Åström. “Minimal Solvers for Relative Pose with a Single Unknown Radial Distortion”. eng. In: *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE - Institute of Electrical and Electronics Engineers Inc., 2014, pp. 33–40.
- [15] Z. Kukelova, J. Heller, M. Bujnak, and T. Pajdla. “Radial distortion homography”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 639–647.
- [16] Z. Kukelova, M. Bujnak, and T. Pajdla. “Automatic generator of minimal problem solvers”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2008), pp. 302–315.
- [17] Z. Kukelova, M. Bujnak, and T. Pajdla. “Closed-Form Solutions to Minimal Absolute Pose Problems with Known Vertical Direction”. In: *Asian Conference on Computer Vision (ACCV)*. Queenstown, New Zealand, Nov. 2010.
- [18] Z. Kukelova, M. Byröd, K. Josephson, T. Pajdla, and K. Åström. “Fast and Robust Numerical Solutions to Minimal Problems for Cameras with Radial Distortion”. In: *Computer Vision and Image Understanding* 114.2 (2010), pp. 234–244.
- [19] Z. Kukelova, J. Kileel, B. Sturmfels, and T. Pajdla. “A Clever Elimination Strategy for Efficient Minimal Solvers”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 3605–3614.
- [20] Z. Kukelova, T. Pajdla, and M. Bujnak. “Fast and Stable Algebraic Solution to L2 Three-View Triangulation”. In: *2013 International Conference on 3D Vision (3DV 2013)*. June 2013, pp. 326–333.
- [21] V. Larsson and K. Åström. “Uncovering Symmetries in Polynomial Systems”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2016), pp. 252–267.
- [22] V. Larsson, K. Åström, and M. Oskarsson. “Efficient Solvers for Minimal Problems by Syzygy-based Reduction”. In: *Computer Vision and Pattern Recognition (CVPR)* (July 2017), pp. 2383–2392.
- [23] V. Larsson, K. Åström, and M. Oskarsson. “Polynomial Solvers for Saturated Ideals”. In: *International Conference on Computer Vision (ICCV)* (Oct. 2017), pp. 2307–2316.
- [24] V. Larsson, Z. Kukelova, and Y. Zheng. “Making Minimal Solvers for Absolute Pose Estimation Compact and Robust”. In: *International Conference on Computer Vision (ICCV)*. Venice, Italy, Oct. 2017, pp. 2335–2343.

- [25] V. Larsson, Z. Kukelova, and Y. Zheng. “Camera Pose Estimation with Unknown Principal Point”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 2984–2992.
- [26] V. Larsson, M. Oskarsson, K. Åström, A. Wallis, Z. Kukelova, and T. Pajdla. “Beyond Gröbner Bases: Basis Selection for Minimal Solvers”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 3945–3954.
- [27] H. Li. “A Simple Solution to the Six-Point Two-View Focal-Length Problem”. In: *European Conference on Computer Vision (ECCV)*. Graz, Austria, May 2006, pp. 200–213.
- [28] H. Li and R. I. Hartley. “Five-Point Motion Estimation Made Easy”. In: *Proceedings of the 18th International Conference on Pattern Recognition (ICPR)*. Vol. 1. Hong Kong (SAR): IEEE, Aug. 2006, pp. 630–633. ISBN: 0-7695-2521-0.
- [29] H. C. Longuet-Higgins. “A computer algorithm for reconstructing a scene from two projections”. In: *Nature* 293 (Sept. 1981), pp. 133–135.
- [30] H. M. Möller and H. J. Stetter. “Multivariate polynomial equations with multiple zeros solved by matrix eigenproblems”. In: *Numerische Mathematik* 70.3 (1995), pp. 311–329.
- [31] O. Naroditsky and K. Daniilidis. “Optimizing polynomial solvers for minimal geometry problems”. In: *2011 International Conference on Computer Vision*. Nov. 2011, pp. 975–982.
- [32] D. Nistér. “An Efficient Solution to the Five-Point Relative Pose Problem”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.6 (June 2004), pp. 756–770.
- [33] T. Palmér, K. Åström, and J.-M. Frahm. “The Misty Three Point Algorithm for Relative Pose”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA, July 2017, pp. 4551–4559.
- [34] J. Pritts, Z. Kukelova, V. Larsson, and O. Chum. “Radially-Distorted Conjugate Translations”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.
- [35] O. Saurer, P. Vasseur, R. Boutteau, C. Demonceaux, M. Pollefeys, and F. Fraundorfer. “Homography Based Egomotion Estimation with a Common Direction”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.2 (Feb. 2017), pp. 327–341. ISSN: 1939-3539.

-
- [36] O. Saurer, F. Fraundorfer, and M. Pollefeys. “Homography based visual odometry with known vertical direction and weak Manhattan world assumption”. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vilamoura, Portugal, 2012.
- [37] F. Schilling, J. Lecoeur, F. Schiano, and D. Floreano. “Learning Vision-Based Flight in Drone Swarms by Imitation”. In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 4523–4530.
- [38] H. Stewénius, C. Engels, and D. Nistér. “Recent developments on direct relative orientation”. In: *Journal of Photogrammetry and Remote Sensing* 60.4 (June 2006), pp. 284–294.
- [39] H. Stewénius, F. Schaffalitzky, and D. Nistér. “How Hard is 3-view Triangulation Really?” In: *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV’05)*. Vol. 1. Beijing, China: IEEE Computer Society, Oct. 2005, pp. 686–693.
- [40] C. Wu. “P3.5P: Pose estimation with unknown focal length”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA, June 2015, pp. 2440–2448.

Efficient Real-Time Radial Distortion Correction for UAVs

MARCUS VALTONEN ÖRNHAG¹, PATRIK PERSSON¹, MÅRTEN WADENBÄCK²,
KALLE ÅSTRÖM¹ AND ANDERS HEYDEN¹

¹*Centre for Mathematical Sciences, Lund University*

²*Department of Electrical Engineering, Linköping University*

Abstract: In this paper we present a novel algorithm for onboard radial distortion correction for unmanned aerial vehicles (UAVs) equipped with an inertial measurement unit (IMU), that runs in real-time. This approach makes calibration procedures redundant, thus allowing for exchange of optics extemporaneously. By utilizing the IMU data, the cameras can be aligned with the gravity direction. This allows us to work with fewer degrees of freedom, and opens up for further intrinsic calibration. We propose a fast and robust minimal solver for simultaneously estimating the focal length, radial distortion profile and motion parameters from homographies. The proposed solver is tested on both synthetic and real data, and perform better or on par with state-of-the-art methods relying on pre-calibration procedures. Code available at: <https://github.com/marcusvaltonen/HomLib>.¹

1 Introduction

In epipolar geometry, the relative pose of two uncalibrated camera views is encoded algebraically as the fundamental matrix F concomitant with the two views. When trying to estimate F from point correspondences, it is well-known that the minimal case—*i.e.* the smallest number of point correspondences for which there exists at most finitely many solutions—uses seven point correspondences [18]. By using eight point correspondences instead of seven, the estimation problem results in a system of eight linear equations, which can be solved fast and in a numerically robust manner using the singular value decomposition (SVD) [19]. To solve the minimal case, *i.e.* using only seven point correspondences, one must conjoin the seven linear equations with one cubic equation emanating from the rank constraint $\det F = 0$. In the case of calibrated cameras, the minimal case involves only five point correspondences; however, the corresponding system of polynomial equations now contains ten cubic equations, and the complexity of the solver increases further [40].

¹This work was supported by the Swedish Research Council (grant no. 2015-05639), the strategic research projects ELLIIT and eSENCE, the Swedish Foundation for Strategic Research project, Semantic Mapping and Visual Navigation for Smart Robots (grant no. RIT15-0038), and Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

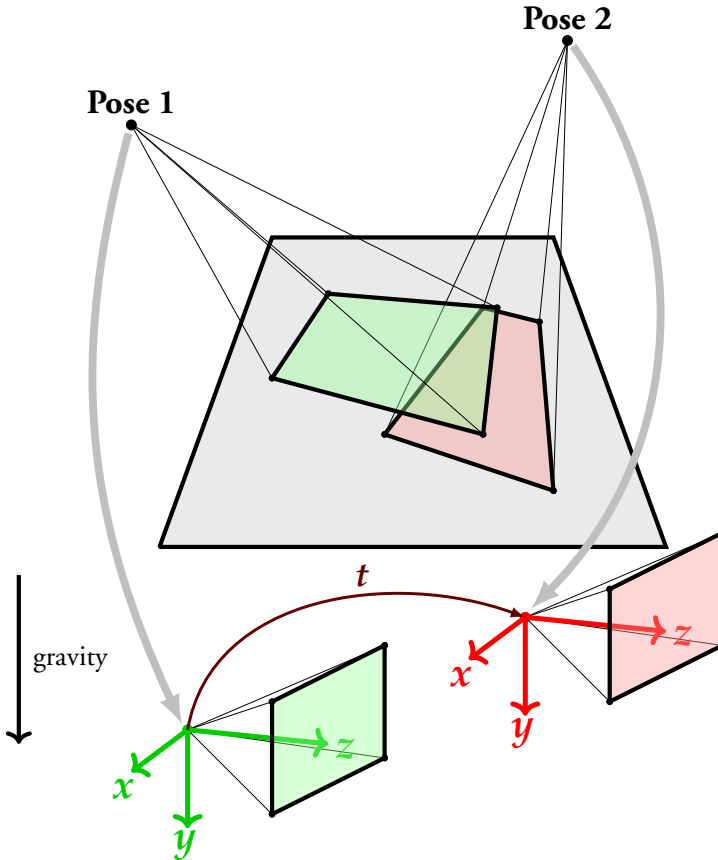


Figure 1: Utilizing the IMU data it is possible to align the camera views, only leaving an unknown translation t . This assumes that the IMU drift is negligible, which is a realistic assumption if the measurements are not taken too far apart in time.

There are several benefits of reducing the number of point correspondences used to estimate the motion parameters. In most cases, this comes at the cost of increased complexity of the system to be solved. Solving systems of polynomial equations numerically, in a sufficiently fast and robust way, is a challenging task. One popular method, sometimes referred to as the *action matrix method*, works if there are finitely many solutions [10, 38]. The system of polynomial equations defines an ideal, for which a Gröbner basis can be computed, leading to an elimination template, where the solutions to the original problem are obtained by solving an eigenvalue problem [5, 8, 24]. This process

has been automated by several authors, with the automatic solver by Kukelova *et al.* [24] as one of the first. Recent advances use syzygies to make the elimination template smaller [30], as well as discarding spurious solutions by saturating the ideal [31]. Using Gröbner bases is not the only option; it has for example been shown that other bases can yield better performance [34], and that overcomplete spanning sets sometimes give better numerical stability [8]. Recently, alternative methods relying on resultants show promising results [1, 2].

Apart from adding extra constraints to the motion of the cameras, one may add scene requirements to reduce the number of necessary point correspondences. A classic example is when the point correspondences lie on a plane, in which case they are related through a homography. In applications where a planar environment is known to exist, such as indoor environments, the minimal number of point correspondences for the uncalibrated case is reduced from seven to four, and the corresponding system of equations—known as the Direct Linear Transform (DLT) equations—is linear in the entries of the homography, and can thus be solved using SVD.

If available, additional input data can be obtained from auxiliary sensors. In this paper we will consider UAVs equipped with an IMU, from which the gravity direction can be obtained. This, in turn, is assumed to be aligned with the ground plane normal.

Many commercially available UAVs are equipped with a camera that suffers from radial distortion to some degree. In order for the pinhole camera model to apply, such distortions must be compensated for, which is usually done in a pre-calibration process involving a calibration target. In contrast, we investigate a process for onboard radial distortion auto-calibration, *i.e.* a method capable of computing the radial distortion profile (and focal length) of the optics as well as the motion parameters, without a specific calibration target, thus eliminating the pre-calibration process. This enables the user of the UAV to exchange optics, without the need of intermediate calibration procedures, which may not be feasible without a calibration target. The main contributions of this paper are:

- (i) a novel polynomial solver for simultaneous estimation of radial distortion profile, focal length and motion parameters, suitable for real-time applications,
- (ii) new insights in how to handle IMU drift, and
- (iii) extensive validation on synthetic and real data on a UAV system demonstrating the applicability of the proposed method.

While the algorithms proposed in this paper works for any application where a camera and IMU are available, we will use them exclusively for UAV positioning.

2 Related Work

In most Simultaneous Localization and Mapping (SLAM) frameworks the distortion profile is pre-calibrated using a calibration target. This requires extra off-line processing, as well as scene requirements. For general scenes, there are a number of algorithms for simultaneously estimating the distortion profile and the motion parameters. Some authors propose methods based on large-scale optimization (bundle adjustment) [37], while others suggest using polynomial solvers [6, 7, 20, 22, 23, 25, 26, 29, 32, 33, 41, 42, 47]. A polynomial solver for the minimal case, *i.e.* the smallest number of point correspondences, is referred to as a minimal solver. There are several reasons to prefer minimal solvers, as they accurately encode intrinsic constraints, and transfer such properties to the final solution. Furthermore, they are suitable for robust estimation frameworks, *e.g.* RANSAC, as the number of necessary iterations (to obtain an inlier set with a pre-defined probability) is minimized.

There exists a number of different models for estimating the distortion profile. One classic approach, that is still frequently used in applications is the Brown–Conrady model [4]; however, although exceptions do exist [35], the division model by Fitzgibbon [12] is almost universally used in the construction of minimal solvers that deal with radial distortion. One reason for this is that the distortion profile can be accurately estimated using fewer parameters, which is consistent with the general theory behind minimal solvers. Other parametric models, recently *e.g.* [45], have been proposed, but are not suitable for minimal solvers for the same reason.

There are several methods that leverage the IMU data—or, simply, rely on the mechanical setup to be accurate enough—to assume a motion model with a known reference direction [11, 13–16, 21, 39, 43, 44, 46, 48]. None of the mentioned papers, however, include simultaneous radial distortion correction, while only a handful consider the case of unknown focal length [11, 14, 16, 48]. To the best of our knowledge, we propose the first ever simultaneous distortion correction, focal length and motion estimation algorithm utilizing IMU data.

3 Embracing the IMU Drift

Prevailing methods have been conceived under the assumption that only two angles can be compensated for using the IMU data, which is true under general conditions. The drift in the yaw angle, however, is often very small for consecutive frames. The idea is that we can disregard the error for the yaw angle initially, and instead correct for it later in the pipeline when enough time has passed for the drift to make a noticeable impact. This makes the equations significantly easier to handle, and allows for further intrinsic

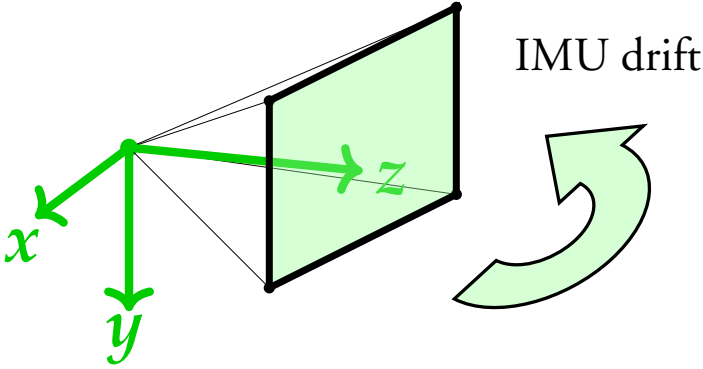


Figure 2: The pitch and roll angles can be accurately estimated using an IMU; however, the yaw angle (about the y -axis, also the gravity direction) often suffers from a drift that accumulates over time. By fusing angular velocity and accelerometer measurements, the drift is negligible for small time frames.

calibration, such as radial distortion correction.

Using [36], the orientation can be estimated both robustly and efficiently by fusing angular velocity and accelerometer measurements to form a single estimate of the orientation. The filter operates by integrating gyroscopic data and compensating for bias and integration errors by using the orientation that can be observed from the accelerometer. The rotation about the gravitational direction, however, is not observable and an inevitable drift will accumulate, see Figure 2. The drift is typically very small for short time frames since the sensor noise of the gyroscope is usually very low and the bias changes slowly.

3.1 New Assumptions on the Homography

Assume the reference direction is known, and aligned with the gravitational direction, chosen as the y -axis. Then, after a suitable change of coordinates, we may assume that

$$\mathbf{H}_y \sim \mathbf{I} + \frac{1}{d} \mathbf{t} \mathbf{n}^T, \quad (1)$$

where \mathbf{I} is the identity matrix, \mathbf{t} is the translation vector and \mathbf{n} is the unit normal of the plane, see Figure 1. We will assume that the plane normal is aligned with the gravitational direction, which is a valid assumption when using the ground floor, thus $\mathbf{n} = [0, 1, 0]^T$. To ease notation, define

$$\mathbf{y}_i^{(j)} := \mathbf{R}_j^T \mathbf{K}^{-1} \mathbf{x}_i, \quad (2)$$

where \mathbf{R}_j is the rotation between the two coordinate systems (given by the IMU) and $\mathbf{K} = \text{diag}(f, f, 1)$ is the calibration matrix, where f is the focal length, which is assumed to be constant. Then for two point correspondences $\mathbf{x}_1 \leftrightarrow \mathbf{x}_2$ the DLT equations can be written as

$$\mathbf{y}_2^{(2)} \times \mathbf{H}_y \mathbf{y}_1^{(1)} = \mathbf{0}. \quad (3)$$

The relation between the general (uncalibrated) homography \mathbf{H} and \mathbf{H}_y is thus given by

$$\mathbf{H}_y \sim \mathbf{R}_2^T \mathbf{K}^{-1} \mathbf{H} \mathbf{K} \mathbf{R}_1, \quad (4)$$

where $\mathbf{x}_2 \sim \mathbf{H} \mathbf{x}_1$. From this, the relative rotation \mathbf{R}_{rel} and the direction of the relative translation \mathbf{t}_{rel} can be extracted, and are given by

$$\mathbf{R}_{\text{rel}} = \mathbf{R}_2 \mathbf{R}_1^T \quad \text{and} \quad \mathbf{t}_{\text{rel}} \sim \mathbf{R}_2 \mathbf{t}. \quad (5)$$

Due to the global scale ambiguity, we may assume $d = 1$, and write

$$\mathbf{H}_y = \begin{bmatrix} 1 & h_1 & 0 \\ 0 & h_2 & 0 \\ 0 & h_3 & 1 \end{bmatrix}, \quad (6)$$

where \mathbf{t} can be extracted directly through the entries h_i , given by

$$\mathbf{t} = \begin{bmatrix} h_1 \\ h_2 - 1 \\ h_3 \end{bmatrix}. \quad (7)$$

In order to apply the pinhole camera model, radially distorted feature points must be rectified. Assuming the distortion can be modeled by the division model [12], using only a single distortion parameter λ , the distorted (measured) image point \mathbf{x}_i in camera i obeys the relationship

$$\mathbf{x}_i^u = \varphi(\mathbf{x}_i, \lambda) = \begin{bmatrix} x_i \\ y_i \\ 1 + \lambda(x_i^2 + y_i^2) \end{bmatrix}, \quad (8)$$

where $\mathbf{x}_i = [x_i, y_i, 1]^T$, and \mathbf{x}_i^u are the undistorted image points compatible with the pinhole camera model. Here we implicitly assume that the distortion center is at the center of the image. The modified DLT equations, can therefore be written as

$$\varphi(\mathbf{x}_i, \lambda) \times \mathbf{H} \varphi(\mathbf{x}_j, \lambda) = \mathbf{0}, \quad (9)$$

for two point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}_j$.

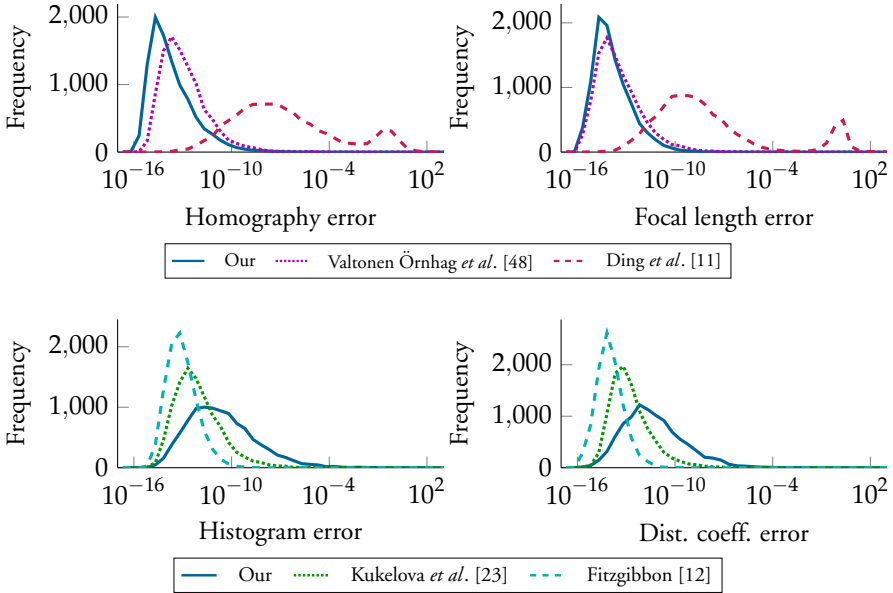


Figure 3: Error histogram for 10,000 randomly generated problem instances for the proposed solvers: (left) fHf and (right) $frHfr$. The solver [23] estimates two focal lengths, and we calculate the error for both and report the geometric mean. Most solvers have an acceptable error distribution, since in practice it rarely has an impact if the error is of the magnitude 10^{-14} or 10^{-10} .

3.2 Benefits of this Approach

The homography described in Appendix 3.1 is greatly simplified compared to a general homography and has fewer parameters that need to be determined. In the case of unknown radial distortion profile, the competing methods [12, 23] return a general homography, *i.e.* with eight degrees of freedom. Unless one makes assumptions about the motion of the cameras—for example that it consists only of pure rotations—it is not possible to extract the motion parameters, even in the partially calibrated case. To see this, note that a Euclidean homography

$$H_{\text{euc}} \sim \mathbf{R} + \mathbf{t}\mathbf{n}^T, \quad (10)$$

has eight degrees of freedom—three in \mathbf{R} , three in \mathbf{t} and two in \mathbf{n} (since the length of \mathbf{n} is arbitrary). This is to be compared to a general homography that also has eight degrees of freedom. We conclude that a partially calibrated homography on the form $\mathbf{K}H_{\text{euc}}\mathbf{K}^{-1}$

must have nine degrees of freedom (the focal length f in \mathbf{K} and the eight from \mathbf{H}_{unc}), hence is over-parametrized, *i.e.* there exists a one-dimensional family of possible decompositions. For this reason, we cannot extract the pose of the methods [12, 23], unless we assume that we know the focal length *a priori*, or constrain the motion. This, in itself, makes the methods infeasible to include in a SLAM framework, where we want to estimate the camera positions.

4 Polynomial Solvers

In this section we present two-sided solvers, *i.e.* when the same intrinsic parameters (focal length and/or radial distortion parameter) are assumed for both cameras.

4.1 Calibrated Case (1.5-point)

This case does not model an unknown focal length or distortion parameter, and is essentially the same approach as in [15], but is given here for completeness. Given 1.5 point correspondences it is possible to form the linear system $\mathbf{A}\mathbf{h} = \mathbf{b}$, where \mathbf{A} is a 3×3 matrix and \mathbf{h} contains the h_i from (6). For non-degenerate configurations, the matrix \mathbf{A} has full rank, and the solution can be obtained immediately as $\mathbf{h} = \mathbf{A}^{-1}\mathbf{b}$. This is a very fast solver, since it is linear and can be solved without SVD.

4.2 Equal and Unknown Focal Length (fHf , 2-point)

Parameterize the inverse of the unknown calibration matrix as $\mathbf{K}^{-1} = \text{diag}(1, 1, w)$, and consider the rectified points (2), which now depend linearly on the unknown parameter w . Parameterizing \mathbf{H}_y as in (6), it is clear that the equations obtained from (3) are linear in h_1, h_2 and h_3 and quadratic in w . This system of equations has infinitely many solutions, if we allow $w = 0$. Such solutions, however, do not yield geometrically meaningful reconstructions, and should therefore be excluded. This can be achieved using saturation, through the method suggested in [31].

We exploit the linear relation of h_1, h_2, h_3 , making it possible to write the equations as

$$\mathbf{M} \begin{bmatrix} \mathbf{h} \\ 1 \end{bmatrix} = \mathbf{0}, \quad (11)$$

where \mathbf{M} is a 4×4 matrix depending on w , and \mathbf{h} is the vector containing the elements h_i . Thus, we may consider finding a non-trivial nullspace of \mathbf{M} , which exists if and only if $\det \mathbf{M} = 0$. This equation reduces to a sextic polynomial in the unknown w , thus has six solutions, which can be found using a simple root finding algorithm (action

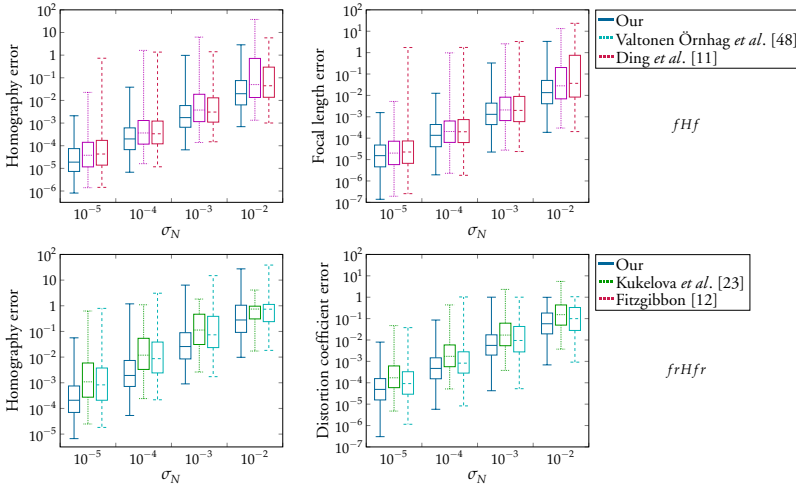


Figure 4: Noise sensitivity comparison for Gaussian noise with standard deviation σ_N . For each noise level 1,000 random problem instances were generated. The geometric mean error is shown for [23].

matrix method is not necessary). Since we know from before that the original problem has four solutions, we conclude that two spurious solutions have been added; in fact, these can easily be disregarded as a pre-processing step, as they correspond to nullspace basis vectors with last element equal to zero. Numerical tests confirm that this is the case.

When the (up to) four possible real solutions of w have been obtained, the unknowns h_1 , h_2 and h_3 can be obtained using SVD.

4.3 Equal and Unknown Focal Length and Radial Distortion Coefficient ($frHfr$, 2.5-point)

Let us now consider the case with equal and unknown focal length and radial distortion coefficient. We use the division model introduced in [12], with a single distortion parameter λ .

Given two point correspondences $x_1 \leftrightarrow x_2$, the modified DLT equations (9) hold true. Building an elimination template from these equations yields a large and numerically unstable solver, and therefore, we reparameterize the problem. Applying $H_y = I + tn^T$ to (4), we get

$$K^{-1}HK \sim R_2 H_y R_1^T = R_2 R_1^T + R_2 t n^T R_1^T. \quad (12)$$

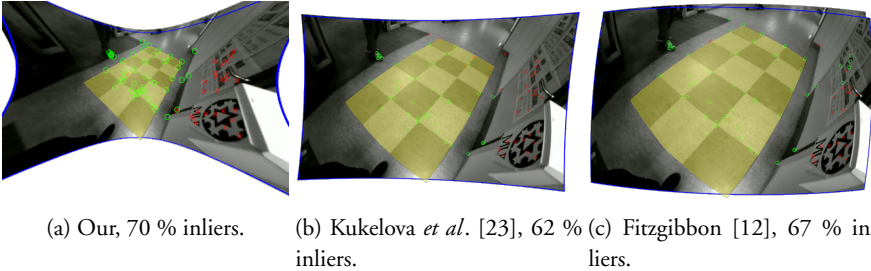


Figure 5: Selection of panoramas created with the competing methods. The blue frame is added for visualization, as well as inliers (green circles) and outliers (red crosses). Note that none of the methods require a checkerboard to be visible in the scene, but is simply chosen to ease the ocular inspection of the stitching. A correct rectification will map physically straight lines to straight lines, *i.e.* the yellow area should be a quadrilateral. Only our method is capable of producing this result.

Denote the elements of \mathbf{R}_1 by r_{ij} . Introducing $\hat{\mathbf{R}} = \mathbf{R}_2 \mathbf{R}_1^T$, $\hat{\mathbf{t}} = \mathbf{R}_2 \mathbf{t}$ and $\hat{\mathbf{n}} = \mathbf{R}_1 \mathbf{n} = [r_{12}, r_{22}, r_{32}]^T$, the general homography can now be written as

$$H \sim \mathbf{K} \left(\hat{\mathbf{R}} + \hat{\mathbf{t}} \hat{\mathbf{n}}^T \right) \mathbf{K}^{-1}. \quad (13)$$

This accomplishes two things: (1) we have replaced several multiplications, (2) we have reduced the number of input data necessary. Analyzing the quotient ring of the corresponding ideal, we conclude that there are three possible solutions.

We parameterize the calibration matrix as $\mathbf{K} = \text{diag}(f, f, 1)$ and its inverse $\mathbf{K}^{-1} = \text{diag}(f^{-1}, f^{-1}, 1)$, respectively. From here on it would be possible to construct an elimination template; however, we may eliminate one variable in order to get a reduced system. Using only the third row of (9), from three point correspondences, one obtains a system on the form $\mathbf{M} \mathbf{v} = \mathbf{0}$, where \mathbf{M} is a 3×9 coefficient matrix, and \mathbf{v} is the vector of monomials, more precisely²

$$\mathbf{v} = [\hat{t}_1 f \lambda \quad \hat{t}_1 f \quad \hat{t}_1 \quad \hat{t}_2 f \lambda \quad \hat{t}_2 f \quad \hat{t}_2 \quad f \lambda \quad f \quad 1]^T. \quad (14)$$

Since \hat{t}_1 and \hat{t}_2 are present in only three monomials, either of the two can be eliminated; we will proceed by eliminating the latter, as it yields a smaller elimination template. After

²It turns out that the third row does not contain any reciprocal f .

Gauss–Jordan elimination, the coefficient matrix is given by

$$\hat{M} = \begin{bmatrix} \hat{t}_2 f \lambda & \hat{t}_2 f & \hat{t}_2 & \hat{t}_1 f \lambda & \hat{t}_1 f & \hat{t}_1 & f \lambda & f & 1 \\ 1 & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & 1 & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & & 1 & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \end{bmatrix}, \quad (15)$$

from which we establish the following relations

$$\begin{aligned} \hat{t}_2 f \lambda + g_1(\hat{t}_1, f, \lambda) &= 0, \\ \hat{t}_2 f + g_2(\hat{t}_1, f, \lambda) &= 0, \\ \hat{t}_2 + g_3(\hat{t}_1, f, \lambda) &= 0, \end{aligned} \quad (16)$$

where g_i are polynomials of three variables. Furthermore, the following constraints must be fulfilled

$$\begin{aligned} g_1(\hat{t}_2, f, \lambda) - \lambda g_2(\hat{t}_2, f, \lambda) &= 0, \\ g_2(\hat{t}_2, f, \lambda) - f g_3(\hat{t}_2, f, \lambda) &= 0, \\ g_1(\hat{t}_2, f, \lambda) - \lambda f g_3(\hat{t}_2, f, \lambda) &= 0. \end{aligned} \quad (17)$$

We can now use the first row of (9), from which we get two equations (which must be multiplied by f to make it polynomial). Together with (17) we have five equations in four unknowns.

To build a solver we saturate f , to remove spurious solutions corresponding to zero focal length. Analyzing the quotient ring we have again three solutions and the basis heuristic [34] yields a template size of 26×29 . Using the hidden variable trick, as in Appendix 4.2, we were able to construct a solver with a template size of 17×20 ; however, this solver was not as numerically stable as the one proposed, nor faster.

5 Experiments

5.1 Numerical Stability and Noise Sensitivity

We compare the proposed methods with other state-of-the-art methods on synthetic data, to evaluate the numerical stability. For the case of unknown focal length we compare to the 2.5 point method [48] and the 3.5 pt method [11], and in the case of unknown radial distortion we compare to the 5 point methods [12, 23]. We generate noise free problem instances, by generating homographies and rotation matrices, and project a random set of points to establish point correspondences. In the case of radial distortion, these points are distorted using the division model. The error histograms are shown in see Figure 3. In the case of unknown focal length our method is superior to the others;

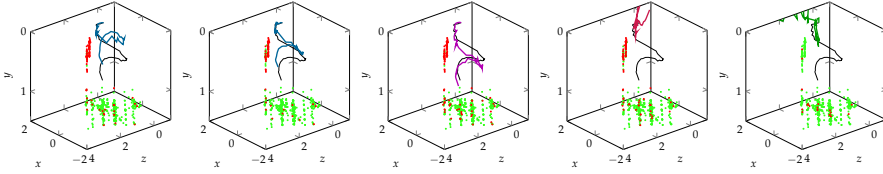


Figure 6: Estimated trajectories from the *Indoor* dataset. From left to right: Our fHf , our $frHfr$, Valtonen Örnthag *et al.* [48], Ding *et al.* [11] and Kukulova *et al.* [27]. The green dots indicate points that have been selected as inliers at least once, and the red points those which have been consistently rejected. Rectified images were used as input to all solvers except for $frHfr$, which received the raw input images.

however, with unknown radial distortion, we are not as stable as others. The accuracy, however, is in the order of 10^{-10} . This is sufficient for most applications. We will see in future experiments, that this does not cause any practical issues. The homography error is measured as the difference between the estimated homography and the ground truth in the Frobenius norm, normalized with the Frobenius norm of the ground truth homography, where the homographies are chosen such that $h_{33} = 1$. The errors of the focal length and radial distortion coefficient are measured as the absolute difference divided by the ground truth value.

Lastly, Gaussian noise is added to the image correspondences, in order to compare the noise sensitivity of the methods. The standard deviation σ_N is varied for a number of different noise levels. For all noise levels, our solvers perform superior to the other methods, for both the case with and without radial distortion, see Figure 4.

5.2 Speed Evaluation

Next we compare the execution time for the considered methods. We compare the mean execution time given a minimal set of point correspondences until the putative homographies, and other parameters are obtained, *i.e.* including all pre-processing and post-processing steps. Furthermore, for the 2.5 and 3.5 point methods, we discard false solutions using the previously unused DLT equation.

As we are interested in performing the computations onboard the UAV, we evaluate the performance on a Raspberry Pi 4, and the mean execution times are listed in Table 1. All solvers are implemented in C++ using Eigen [17] and compiled in g++ with the -O2 optimization flag. Lastly, we list the maximal number of iterations possible on a 30 fps system, which we will use in Appendix 5.3 to compare real-time performance.

Table 1: Mean execution time on a Raspberry Pi 4 for 100,000 randomly generated problem instances in C++. The last column is the maximal number of iterations possible when running 30 fps.

Author	Time (μ s)	No. iter.
Our <i>fHf</i>	215	155
Our <i>frHfr</i>	149	223
Valtonen Örnthag <i>et al.</i> [48]	80	416
Ding <i>et al.</i> [11]	3301	10
Kukelova <i>et al.</i> [27]	371	89
Fitzgibbon [12]	428	77
Kukelova <i>et al.</i> [23]	226	147

5.3 Real Data

In this section, we compare the proposed methods on real data. We use the datasets from [48], captured using a UAV with a monochrome global shutter camera (OV9281) with resolution 640×480 . The UAV is equipped with an inertial measurement unit (MPU-9250). In the experiments with only unknown focal length, the extracted features were undistorted using a pre-calibrated distortion profile (using the OpenCV [3] camera calibration procedure); for the case with unknown radial distortion profile, the raw unprocessed coordinates were used as input.

The ground truth was obtained using a complete SLAM system where the reprojection error and IMU error were minimized. No scene requirements are enforced by the system, hence feature points from non-planar structures will be present—such feature points should be discarded by a robust framework as outliers.

The dataset consists of both indoor and outdoor sequences containing planar surfaces, and includes varying motions and length of sequences. Example images from the sequences are shown in Figure 7.

We exclusively use consecutive keyframes for extracting and matching features. The following keyframe insertion heuristic is used: (i) If the frame has moved more than 2.5% of its median depth from the previous keyframe, then proceed to next step, (ii) If we are tracking 10 % fewer 3D points in the current frame compared to the closest keyframe, or if the uncertainty of the projection of a 3D point is higher than a predefined threshold, then increase new information counter, otherwise, reset counter. (iii) If the new information counter is above 3 and the current frame tracks at least 10 3D points, add frame as keyframe.

We use the IMU filter technique [36], described in Appendix 3, to obtain the estimates. Since these measurements are noisy, we propose to use the novel solvers in a

LO-RANSAC framework [9]. As a first step the solvers are used to discard outliers and in the inner LO loop we propose to optimize over the space of Euclidean homographies with unknown focal length. This refinement step allows for correction of the errors initially caused by the IMU filter. Empirically, we have seen that this improves the accuracy.

For a fair comparison, we simulate a scenario where the UAV uses a frame rate of 30 fps, and limit the number of RANSAC cycles to fit this time frame. For simplicity, we use the values from Table 1 alone, acknowledging there are other parts in the pipeline—such as image capturing, feature extraction and matching, LO cycles, etc.—that would affect a complete system; however, we argue that this overhead time is roughly independent of the solver used.

When working with radial distortion correction, one may choose to minimize the reprojection error in the undistorted image space, or in the distorted image space. In [28], it was shown that it is beneficial to perform triangulation in the distorted image space. Therefore, we chose to measure the reprojection in the distorted space.

Image Stitching

As argued in Appendix 3.2 we cannot decompose the homographies obtained from [12, 23], into a relative pose; however, we can still test the ability of the methods to return an accurate distortion profile.

We simulate a scenario where a UAV is navigating in 30 fps, and limit the number of iterations for each method according to Table 1. We use the same pixel threshold for all methods, for two consecutive keyframes of the *Indoor* sequence. We chose this sequence, because it naturally contains a checkerboard pattern, which facilitates in making an accurate ocular evaluation of the quality of the estimated distortion profile. Physically straight lines should be mapped to straight lines if the rectification is successful.

In Figure 5 we show the results of the estimated distortion profile. We notice that the distortion profile is correct for the proposed method as the yellow area is a quadrilateral, whereas this is not the case for other methods. Furthermore, we note that the method by Kukulova *et al.* [23] does not contain all inliers of the ground plane, and that the method by Fitzgibbon [12] pick incorrect matches of the wall.



Figure 7: Example images from the dataset [48].

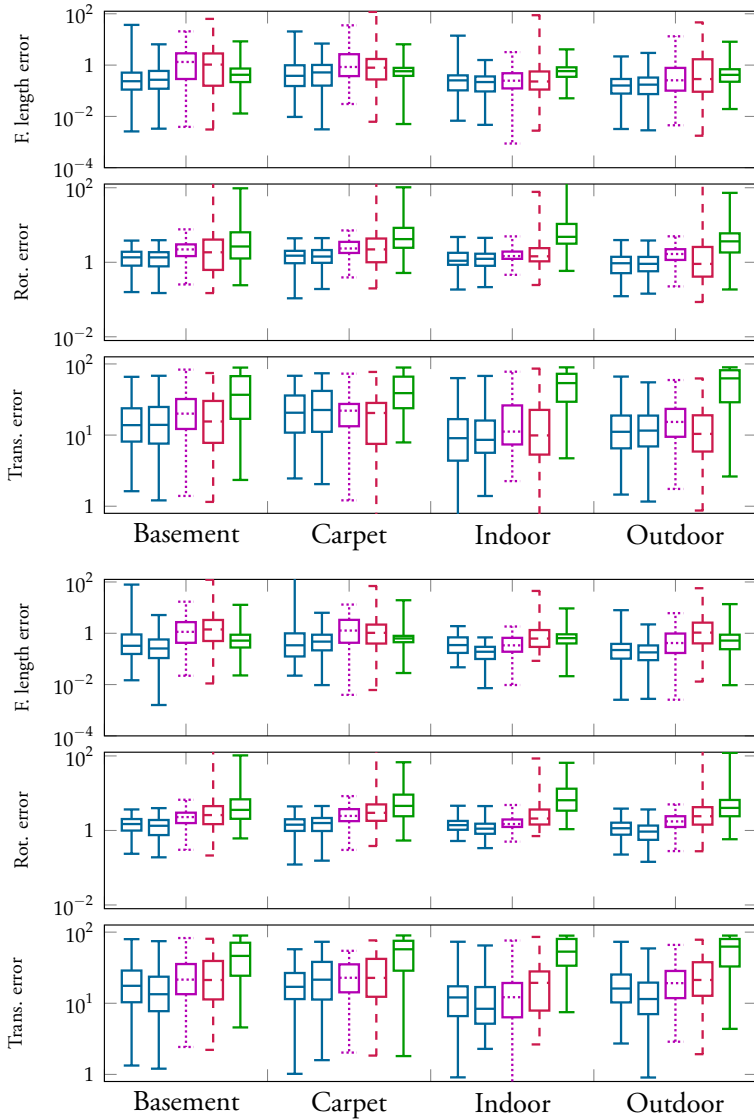


Figure 8: Errors for the different methods—from left to right: fHf , $frHfr$, Valtonen Örn-hag *et al.* [48], Ding *et al.* [11], Kukulova *et al.* [27]—using the metrics (18). (Top) rectified input images were used for all but the $frHfr$. (Bottom) unrectified images were used for all methods.

For the same pair of images we measure the inliers as a function of time, see Figure 9. The only method converging to the correct number of inliers in the allotted time is our method, which it does by a large margin.

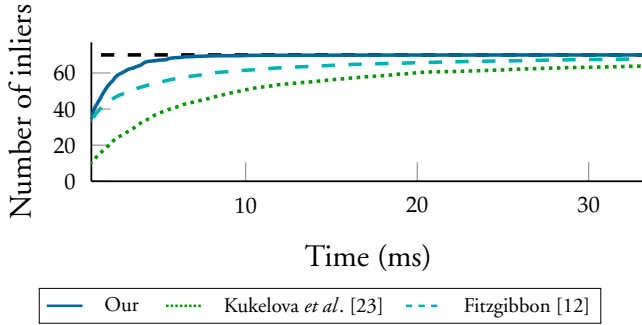


Figure 9: Number of inliers over time (average of 100 tests). The proposed method is the only method to converge to the correct number of inliers (black dashed line) within the 33.3 ms allotted per frame when running at 30 fps.

Pose Estimation

In this final section, we compare both proposed methods to [11, 27, 48]. Note that only one of these methods (our *frHfr*) estimates the radial distortion profile. As argued in Appendix 3.2, the methods [12, 23] cannot estimate the motion parameters without additional requirements that are not applicable for UAVs, hence cannot be compared in this section.

The error metrics are defined as in [11, 43, 48], and are given by

$$\begin{aligned}
 e_R &= \arccos\left(\frac{\text{tr}(\mathbf{R}_{\text{GT}}\mathbf{R}_{\text{est}}^T) - 1}{2}\right), \\
 e_t &= \arccos\left(\frac{\mathbf{t}_{\text{GT}}^T \mathbf{t}_{\text{est}}}{\|\mathbf{t}_{\text{GT}}\| \|\mathbf{t}_{\text{est}}\|}\right), \\
 e_f &= \frac{|f_{\text{GT}} - f_{\text{est}}|}{f_{\text{GT}}}.
 \end{aligned} \tag{18}$$

In Figure 6 we compare the estimated trajectories for all methods. It can be seen that there are only small differences between the methods using pre-calibrated radial distortion profile, and the proposed *frHfr* solver. Furthermore, we measure the errors, according to (18) for all four sequences. In the left part of Figure 8 we use rectified

images for all except the *frHfr* method, which still performs best or on par with the other methods in terms of all errors. In the right part of the figure, we run the same experiment, but all methods are given the raw (unrectified) images as input—here it is clear that our method achieves superior results.

6 Conclusions

We have presented the first ever method capable of simultaneously estimating the distortion profile, focal length and motion parameters from a pair of homographies, while incorporating the IMU data. The method relies on a novel assumption that the IMU data is accurate enough, to disregard the IMU drift for small time frames, allowing for simpler equations and faster solvers. We have shown that this assumption is true on both synthetic and real data, and that the proposed methods are robust. The method has been shown to give accurate reconstructions, and performs on par or better than state-of-the-art methods relying on pre-calibration procedures, while being fast enough for real-time applications.

References

- [1] S. Bhayani, Z. Kukelova, and J. Heikkilä. “A Sparse Resultant Based Method for Efficient Minimal Solvers”. In: *Computer Vision and Pattern Recognition (CVPR)*. June 2020.
- [2] S. Bhayani, Z. Kukelova, and J. Heikkilä. “Computing stable resultant-based minimal solvers by hiding a variable”. In: *CoRR* abs/2007.10100 (2020).
- [3] G. Bradski. “The OpenCV Library”. In: *Dr. Dobb’s Journal of Software Tools* (2000).
- [4] D. C. Brown. “Decentering Distortion of Lenses”. In: *Photogrammetric Engineering* 32 (Mar. 1966), pp. 444–462.
- [5] M. Bujnak, Z. Kukelova, and T. Pajdla. “Making Minimal Solvers Fast”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2012.
- [6] M. Bujnak, Z. Kukelova, and T. Pajdla. “New Efficient Solution to the Absolute Pose Problem for Camera with Unknown Focal Length and Radial Distortion”. In: *Asian Conference on Computer Vision (ACCV)*. Nov. 2011.

- [7] M. Byröd, M. Brown, and alle. Åstrom. “Minimal Solutions for Panoramic Stitching with Radial Distortion”. In: *Proceedings of the British Machine Vision Conference (BMVC)*. London, 2009.
- [8] M. Byröd, K. Josephson, and K. Åström. “Fast and Stable Polynomial Equation Solving and Its Application to Computer Vision”. In: *International Journal of Computer Vision* 84.3 (Sept. 2009), pp. 237–256.
- [9] O. Chum, J. Matas, and J. Kittler. “Locally Optimized RANSAC”. In: *Pattern Recognition*. Springer Berlin Heidelberg, 2003, pp. 236–243.
- [10] D. A. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Second edition. Graduate Texts in Mathematics. Springer-Verlag New York, 2005.
- [11] Y. Ding, J. Yang, J. Ponce, and H. Kong. “An Efficient Solution to the Homography-Based Relative Pose Problem With a Common Reference Direction”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Oct. 2019, pp. 1655–1664.
- [12] A. W. Fitzgibbon. “Simultaneous linear estimation of multiple view geometry and lens distortion”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Dec. 2001.
- [13] F. Fraundorfer, P. Tanskanen, and M. Pollefeys. “A Minimal Case Solution to the Calibrated Relative Pose Problem for the Case of Two Known Orientation Angles”. In: *European Conference on Computer Vision (ECCV)*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 269–282.
- [14] B. Guan, Y. Qifeng, and F. Fraundorfer. “Minimal solutions for the rotational alignment of IMU-camera systems using homography constraints”. English. In: *Computer vision and image understanding* (2018), pp. 79–91. ISSN: 1077-3142.
- [15] B. Guan, P. Vasseur, C. Demonceaux, and F. Fraundorfer. “Visual Odometry Using a Homography Formulation with Decoupled Rotation and Translation Estimation Using Minimal Solutions”. In: *International Conference on Robotics and Automation (ICRA)*. 2018, pp. 2320–2327.
- [16] B. Guan, J. Zhao, Z. Li, F. Sun, and F. Fraundorfer. “Minimal Solutions for Relative Pose With a Single Affine Correspondence”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.
- [17] G. Guennebaud, B. Jacob, et al. *Eigen v3*. <http://eigen.tuxfamily.org>. 2010.
- [18] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second edition. Cambridge University Press, 2004.
- [19] R. I. Hartley. “In Defense of the Eight-Point Algorithm”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.6 (June 1997), pp. 580–593.

-
- [20] K. Josephson and M. Byröd. “Pose Estimation with Radial Distortion and Unknown Focal Length”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Miami, FL, USA: IEEE Computer Society, June 2009, pp. 2419–2426. ISBN: 978-1-4244-3991-1.
- [21] M. Kalantari, A. Hashemi, F. Jung, and J.-P. Guedon. “A New Solution to the Relative Orientation Problem Using Only 3 Points and the Vertical Direction”. In: *Journal of Mathematical Imaging and Vision* 39 (Mar. 2011), pp. 259–268.
- [22] Y. Kuang, J. E. Solem, F. Kahl, and K. Åström. “Minimal Solvers for Relative Pose with a Single Unknown Radial Distortion”. eng. In: *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE - Institute of Electrical and Electronics Engineers Inc., 2014, pp. 33–40.
- [23] Z. Kukelova, J. Heller, M. Bujnak, and T. Pajdla. “Radial distortion homography”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 639–647.
- [24] Z. Kukelova, M. Bujnak, and T. Pajdla. “Automatic generator of minimal problem solvers”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2008), pp. 302–315.
- [25] Z. Kukelova, M. Bujnak, and T. Pajdla. “Real-Time Solution to the Absolute Pose Problem with Unknown Radial Distortion and Focal Length”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Dec. 2013.
- [26] Z. Kukelova, M. Byröd, K. Josephson, T. Pajdla, and K. Åström. “Fast and Robust Numerical Solutions to Minimal Problems for Cameras with Radial Distortion”. In: *Computer Vision and Image Understanding* 114.2 (2010), pp. 234–244.
- [27] Z. Kukelova, J. Kileel, B. Sturmfels, and T. Pajdla. “A Clever Elimination Strategy for Efficient Minimal Solvers”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 3605–3614.
- [28] Z. Kukelova and V. Larsson. “Radial Distortion Triangulation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.
- [29] Z. Kukelova and T. Pajdla. “A Minimal Solution to Radial Distortion Autocalibration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.12 (2011), pp. 2410–2422.
- [30] V. Larsson and K. Åström. “Uncovering Symmetries in Polynomial Systems”. In: *European Conference on Computer Vision (ECCV)* (Oct. 2016), pp. 252–267.

- [31] V. Larsson, K. Åström, and M. Oskarsson. “Polynomial Solvers for Saturated Ideals”. In: *International Conference on Computer Vision (ICCV)* (Oct. 2017), pp. 2307–2316.
- [32] V. Larsson, Z. Kukelova, and Y. Zheng. “Camera Pose Estimation With Unknown Principal Point”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018.
- [33] V. Larsson, Z. Kukelova, and Y. Zheng. “Making minimal solvers for absolute pose estimation compact and robust”. In: *International Conference on Computer Vision (ICCV)* (Oct. 2017), pp. 2335–2343.
- [34] V. Larsson, M. Oskarsson, K. Åström, A. Wallis, Z. Kukelova, and T. Pajdla. “Beyond Gröbner Bases: Basis Selection for Minimal Solvers”. In: *Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 3945–3954.
- [35] V. Larsson, T. Sattler, Z. Kukelova, and M. Pollefeys. “Revisiting Radial Distortion Absolute Pose”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019, pp. 1062–1071.
- [36] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan. “Estimation of IMU and MARG orientation using a gradient descent algorithm”. In: *IEEE International Conference on Rehabilitation Robotics*. 2011, pp. 1–7.
- [37] H. Meuel, S. Ferenz, M. Munderloh, H. Ackermann, and J. Ostermann. “In-loop Radial Distortion Compensation for Long-term Mosaicking of Aerial Videos”. In: *Proc. of the 23rd IEEE International Conference on Image Processing (ICIP)* (Sept. 2016), pp. 2961–2965.
- [38] H. M. Möller and H. J. Stetter. “Multivariate polynomial equations with multiple zeros solved by matrix eigenproblems”. In: *Numerische Mathematik* 70.3 (1995), pp. 311–329.
- [39] O. Naroditsky, X. Zhou, J. Gallier, S. Roumeliotis, and K. Daniilidis. “Two Efficient Solutions for Visual Odometry Using Directional Correspondence”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (Nov. 2011), pp. 818–24.
- [40] D. Nistér. “An Efficient Solution to the Five-Point Relative Pose Problem”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.6 (June 2004), pp. 756–770.
- [41] J. Pritts, Z. Kukelova, V. Larsson, and O. Chum. “Radially-Distorted Conjugate Translations”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

-
- [42] J. Pritts, Z. Kukulova, V. Larsson, and O. Chum. “Rectification from Radially-Distorted Scales”. In: *Asian Conference of Computer Vision (ACCV)*. 2018, pp. 36–52.
- [43] O. Saurer, P. Vasseur, R. Boutteau, C. Demonceaux, M. Pollefeys, and F. Fraundorfer. “Homography Based Egomotion Estimation with a Common Direction”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.2 (Feb. 2017), pp. 327–341. ISSN: 1939-3539.
- [44] O. Saurer, F. Fraundorfer, and M. Pollefeys. “Homography based visual odometry with known vertical direction and weak Manhattan world assumption”. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vilamoura, Portugal, 2012.
- [45] T. Schops, V. Larsson, M. Pollefeys, and T. Sattler. “Why Having 10,000 Parameters in Your Camera Model Is Better Than Twelve”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020, pp. 2535–2544.
- [46] C. Sweeney, J. Flynn, and M. Turk. “Solving for Relative Pose with a Partially Known Rotation is a Quadratic Eigenvalue Problem”. In: *2014 2nd International Conference on 3D Vision*. Vol. 1. 2014, pp. 483–490.
- [47] M. Valtonen Örnthag. “Radially Distorted Planar Motion Compatible Homographies”. In: *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. Valletta, Malta, Feb. 2020, pp. 568–575.
- [48] M. Valtonen Örnthag, P. Persson, M. Wadenbäck, K. Åström, and A. Heyden. “Minimal Solvers for Indoor UAV Positioning”. In: *CoRR* abs/2003.07111 (2020).

Bilinear Parameterization for Differentiable Rank-Regularization

MARCUS VALTONEN ÖRNHAG¹, CARL OLSSON^{1,2} AND ANDERS HEYDEN¹

¹*Centre for Mathematical Sciences, Lund University*

²*Department of Electrical Engineering, Chalmers University of Technology*

Abstract: Low rank approximation is a commonly occurring problem in many computer vision and machine learning applications. There are two common ways of optimizing the resulting models. Either the set of matrices with a given rank can be explicitly parametrized using a bilinear factorization, or low rank can be implicitly enforced using regularization terms penalizing non-zero singular values. While the former approach results in differentiable problems that can be efficiently optimized using local quadratic approximation, the latter is typically not differentiable (sometimes even discontinuous) and requires first order subgradient or splitting methods. It is well known that gradient based methods exhibit slow convergence for ill-conditioned problems.

In this paper we show how many non-differentiable regularization methods can be reformulated into smooth objectives using bilinear parameterization. This allows us to use standard second order methods, such as Levenberg–Marquardt (LM) and Variable Projection (VarPro), to achieve accurate solutions for ill-conditioned cases. We show on several real and synthetic experiments that our second order formulation converges to substantially more accurate solutions than competing state-of-the-art methods.

1 Introduction

Low rank models have been applied to numerous vision applications ranging from high level shape and deformation to pixel appearance models [2, 6, 11, 21, 22, 48, 50, 52]. When the sought rank is known, a commonly occurring formulation is the least squares minimization

$$\min_{\text{rank}(\mathbf{X}) \leq r} \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2, \quad (1)$$

where $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ is a linear operator, and $\|\cdot\|$ is the standard Euclidean vector norm. In general, this is a difficult non-convex problem and some versions are even known to be NP-hard [26]. In structure from motion, a popular approach [7] is to optimize over a bilinear factorization $\mathbf{X} = \mathbf{B}\mathbf{C}^T$, where \mathbf{B} is $m \times r$ and \mathbf{C} is $n \times r$, and solve

$$\min_{\mathbf{B}, \mathbf{C}} \|\mathcal{A}\mathbf{B}\mathbf{C}^T - \mathbf{b}\|^2. \quad (2)$$

Since the rank is bounded by the number of columns in \mathbf{B} and \mathbf{C} this approach explicitly parametrizes the set of matrices of rank r . While bilinear approaches often perform well [16, 29] they can have local minima [7]. Recent works [29–32] have, however, shown that properly implemented, LM and VarPro approaches are remarkably robust to local minima, achieve quadratic convergence and give impressive reconstruction results. Recently [3, 23, 24] was able to give conditions which guarantee that there are no "spurious" local minimizers (meaning that all local minimizers are close to or identical to the global solution). They use the notion of restricted isometry property (RIP) [45] which assumes that the operator \mathcal{A} fulfills

$$(1 - \delta_r) \|\mathbf{X}\|_F^2 \leq \|\mathcal{A}\mathbf{X}\|^2 \leq (1 + \delta_r) \|\mathbf{X}\|_F^2, \quad (3)$$

with $0 \leq \delta_r < 1$, if $\text{rank}(\mathbf{X}) \leq r$. If the isometry constant δ_r is sufficiently small [3, 23, 24] prove that every local minimizer is optimal (or near optimal). Similarly, for the matrix completion problem [23] showed that there are no spurious local minima under uniformly distributed missing data. While the above theoretical assumptions generally do not hold for computer vision problems such as structure from motion, these results still give some intuition as to why bilinear parameterization often works well.

An alternative approach is to optimize directly over the entries of \mathbf{X} and penalize high rank using regularization terms. Applying a robust function f to the singular values $\sigma_i(\mathbf{X}) = 1, \dots, N = \min(m, n)$ results in a low-rank inducing objective

$$\min_{\mathbf{X}} \mathcal{R}(\mathbf{X}) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2, \quad (4)$$

where $\mathcal{R}(\mathbf{X}) = \sum_{i=1}^N f(\sigma_i(\mathbf{X}))$. Besides controlling the rank of the solution the generality of the function f offers increased modeling capability compared to (1) and can for example be used to add priors on the size of the non-zero singular values.

The most popular regularization approach is undoubtedly the nuclear norm, $f(\sigma_i(\mathbf{X})) = \sigma_i(\mathbf{X})$, due to its convexity [9, 10, 18, 44, 45]. Under the RIP assumption exact or approximate recovery with the nuclear norm can then be guaranteed [10, 45]. On the other hand, since it penalizes large singular values, it suffers from a shrinking bias [8, 11, 36]. Ideally f should penalize small singular values (assumed to stem from measurement noise) harder than the large ones. Therefore non-increasing derivatives on $[0, \infty)$, or concavity, has been shown to give stronger relaxations [12, 27, 33, 37, 39, 43, 47]. These non-convex formulations usually only come with local convergence guarantees. Two exceptions are [36, 40] which gave optimality guarantees for (4) with $f = f_\mu$ as in (8).

The regularization term is generally not differentiable as a function of \mathbf{X} . Thus, optimization methods based on local quadratic approximation become infeasible. Figure 1 gives a simple illustration of a 1-dimensional example of how non-differentiability

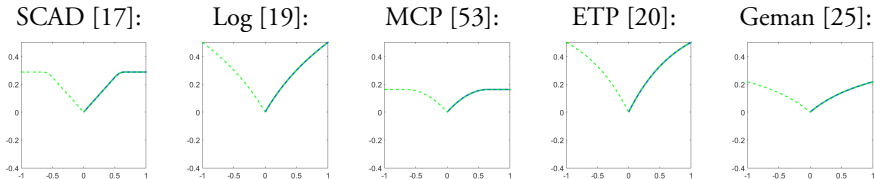


Figure 1: A few commonly occurring robust penalties of the form $f(\sigma)$, with $\sigma \in [0, \infty)$ and f differentiable everywhere (blue graph). The green dashed graph shows how non-differentiability occurs at the origin when applying the penalty to a 1×1 matrix $x \in \mathbb{R}$. In this case $\sigma(x) = |x|$ and therefore $f(\sigma(x)) = f(|x|)$. Note also that (8) is a special case of MCP.

occurs at the origin. In addition it is well known that the singular values become non-differentiable functions of the matrix elements when they are non distinct. To circumvent these issues subgradient and splitting methods are often employed [12, 27, 36, 38, 47]. It is well known from basic optimization theory (*e.g.* [5]) that gradient based methods exhibit slow convergence for ill-conditioned problems. It has also been observed (*e.g.* [4]) that splitting methods rapidly reduce the objective value the first couple of iterations, while convergence to the exact solution can be slow. In this paper we show that there are computer vision problems where these approaches make very little improvements at all, returning a solution that is far from optimal. In contrast, bilinear formulations with either LM or VarPro can be made to yield accurate results in few iterations [29].

An alternative approach that unifies bilinear parameterization with regularization approaches is based on the observation [45] that the nuclear norm $\|\mathbf{X}\|_*$ of a matrix \mathbf{X} can be expressed as $\|\mathbf{X}\|_* = \min_{\mathbf{B}\mathbf{C}^T = \mathbf{X}} \frac{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2}{2}$. Thus when $f(\sigma_i(\mathbf{X})) = \mu\sigma_i(\mathbf{X})$, where μ is a scalar controlling the strength of the regularization, optimization of (4) can be formulated as

$$\min_{\mathbf{B}, \mathbf{C}} \mu \frac{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2}{2} + \|\mathbf{A}\mathbf{B}\mathbf{C}^T - \mathbf{b}\|^2. \quad (5)$$

Optimizing directly over the factors has the advantages that the number of variables is much smaller and one may add constraints if a particular factorization is sought. Surprisingly, while (5) is non-convex, using the convexity of the underlying regularization problem (4) it can be shown that any local minimizer \mathbf{B}, \mathbf{C} with $\text{rank}(\mathbf{B}\mathbf{C}^T) < k$, where k is the number of columns in \mathbf{B} and \mathbf{C} , is globally optimal [1, 28]. Additionally, the objective function is two times differentiable and second order methods can be employed.

In this paper we develop new regularizing terms that, similar to (5), work on the bilinear factors. However, in contrast to previous approaches we investigate formulations

that exhibit less shrinking bias and go beyond convex penalties. Specifically, we prove that $\mathcal{R}(\mathbf{X}) = \min_{\mathbf{X}=\mathbf{BC}^T} \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$, where

$$\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) = \sum_{i=1}^k f \left(\frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2} \right), \quad (6)$$

k is the number of columns, and \mathbf{B}_i and \mathbf{C}_i are the i :th columns of \mathbf{B} and \mathbf{C} , respectively. The result holds for a general class of concave penalty functions f , a few of which are illustrated in Figure 1. In view of the above result, we propose to minimize

$$\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}\mathbf{BC}^T - \mathbf{b}\|^2. \quad (7)$$

Rather than resorting to splitting or subgradient methods we present an algorithm that uses a quadratic approximation of the objective. Under the assumption that f is differentiable, we show that our quadratic approximation reduces to a weighted version of (5) to which we can apply VarPro. We show on several computer vision problems that our approach outperforms state-of-the-art methods such as [4, 12, 27, 46, 47].

While our problem is non-convex (both in the \mathbf{X} parameterization (4) and in the \mathbf{B}, \mathbf{C} parameterization (7)) we show that in some cases it is still possible to give global optimality guarantees. Building on the results of [40] we characterize the local minima of the new formulation with the choice

$$f(x) = f_\mu(x) := \mu - \max(\sqrt{\mu} - x, 0)^2. \quad (8)$$

Specifically, for this choice, we give conditions that ensure that when a RIP constraint [45] holds a local minimizer of (7) is a global solution of both

$$\min_{\text{rank}(\mathbf{X}) \leq r} \mathcal{R}(\mathbf{X}) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2, \quad (9)$$

where $\mathcal{R}(\mathbf{X}) = \sum_i f_\mu(\sigma_i(\mathbf{X}))$, and

$$\min_{\text{rank}(\mathbf{X}) \leq r} \mu \text{rank}(\mathbf{X}) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2. \quad (10)$$

In summary our main contributions are:

- A new stronger non-convex regularization term for bilinear parameterizations with less/no shrinking bias.
- A new iteratively reweighted VarPro algorithm optimizing accurate quadratic approximations.
- Theoretical conditions that guarantee optimal recovery under the RIP constraint.
- An experimental evaluation that shows that our methods outperforms state-of-the-art methods on several real computer vision problems.

1.1 Related Work

Our work is very much inspired by a recent series of papers by Hong *et al.* [29–32] which show that bilinear formulations can be made remarkably robust to local minima, and achieve impressive reconstruction results for uncalibrated structure from motion problems, using the so called VarPro method. Our work represents an attempt to unify this line of work with regularization based alternatives, leveraging the benefits of them both.

An approach that is closely related to ours is that of [8] which uses (5) to unify the use of a regularized objective and factorization. They show that if the obtained solution has lower rank than its number of columns it is globally optimal. In practice [8] observes that the shrinking bias of the nuclear norm makes it too weak to enforce a low rank when the data is noisy. Therefore, a “continuation” approach where the size of the factorization is gradually reduced is proposed. While this yields solutions with lower rank, the optimality guarantees no longer apply.

Bach *et al.* [1] showed that

$$\|\mathbf{X}\|_{s,t} := \min_{\mathbf{X}=\mathbf{BC}^T} \sum_{i=1}^k \frac{\|\mathbf{B}_i\|_s^2 + \|\mathbf{C}_i\|_t^2}{2}, \quad (11)$$

is convex for any choice of vector norms $\|\cdot\|_s$ and $\|\cdot\|_t$. In [28] it was shown that a more general class of 2-homogeneous factor penalties result in a convex regularization similar to (11). The property that a local minimizer \mathbf{B} , \mathbf{C} with $\text{rank}(\mathbf{BC}^T) < k$, is also extended to this case. Still, because of convexity, it is clear that these formulations will suffer from a similar shrinking bias as the nuclear norm. Shang *et al.* [46] showed that penalization with the Schatten semi-norms $\|\mathbf{X}\|_q = \sqrt[q]{\sum_{i=1}^N \sigma_i(\mathbf{X})^q}$, for $q = 1/2$ and $2/3$, can be achieved using a convex penalty on the factors \mathbf{B} and \mathbf{C} . A generalization to general values of q is given in [51]. While this reduces shrinking bias to some extent, it results in a non-differentiable and non-convex formulation that is optimized with ADMM. In [34] a bilinear framework on the class of semi-definite matrices was proposed for a general class of low-rank inducing penalties.

Valtonen Örnhaug *et al.* [42] considered a framework similar to the one we propose; however, the rank is assumed to be known a priori. Furthermore, they do not show equivalence between the proposed bilinear regularizer and the corresponding original fixed rank regularizer from [36].

It is important to note that many of the above methods that are considered state-of-the-art have been developed for low level vision tasks such as image denoising, inpainting, alignment and background subtraction. The ground truth for these models are often of higher rank than models in *e.g.* structure from motion, making it possible to obtain

good results with weaker regularization. Additionally, as we will see in the experiments, more difficult data terms prevent rapid convergence of the splitting methods they often employ.

2 Non-Convex Penalties and Shrinking Bias

In this section we will show how to formulate regularization terms of the type

$$\mathcal{R}(\mathbf{X}) = \sum_{i=1}^N f(\sigma_i(\mathbf{X})), \quad (12)$$

by penalizing the factors of the factorization $\mathbf{X} = \mathbf{B}\mathbf{C}^T$. We assume that \mathbf{B} and \mathbf{C} have k columns, making $\sigma_i(\mathbf{X}) = 0$ if $i > k$ and $\text{rank}(\mathbf{X}) \leq k$. Note, however, that we are aiming to achieve a lower rank using the regularization term. In many applications, the sought rank is unknown and should be determined by the regularization. We therefore set k large enough not to exclude the optimal solution. As we shall see in Section 3, this ability to over-parameterize can be used to ensure optimality.

Theorem 1. *If f is concave, non-decreasing on $[0, \infty)$ and $f(0) = 0$ then*

$$\mathcal{R}(\mathbf{X}) = \min_{\mathbf{B}\mathbf{C}^T = \mathbf{X}} \sum_{i=1}^k f(\|\mathbf{B}_i\| \|\mathbf{C}_i\|), \quad (13)$$

where \mathbf{B}_i and \mathbf{C}_i , $i = 1, \dots, k$ are the columns of \mathbf{B} and \mathbf{C} respectively.

Proof. The result is a consequence of the fact that \mathcal{R} will fulfill a triangle inequality $\mathcal{R}(\mathbf{X} + \mathbf{Y}) \leq \mathcal{R}(\mathbf{X}) + \mathcal{R}(\mathbf{Y})$ under the assumptions on f . This is clear from Theorem 4.4 in [49] which shows that

$$\sum_{i=1}^N f(\sigma_i(\mathbf{X} + \mathbf{Y})) \leq \sum_{i=1}^N (f(\sigma_i(\mathbf{X})) + f(\sigma_i(\mathbf{Y}))). \quad (14)$$

Applying this to $\mathbf{X} = \mathbf{B}\mathbf{C}^T = \sum_{i=1}^k \mathbf{B}_i\mathbf{C}_i^T$ we see that

$$\mathcal{R}(\mathbf{X}) = \mathcal{R}\left(\sum_{i=1}^k \mathbf{B}_i\mathbf{C}_i^T\right) \leq \sum_{i=1}^k \mathcal{R}(\mathbf{B}_i\mathbf{C}_i^T). \quad (15)$$

Since $\text{rank}(\mathbf{B}_i\mathbf{C}_i^T) = 1$ we also have

$$\mathcal{R}(\mathbf{B}_i\mathbf{C}_i^T) = f(\sigma_1(\mathbf{B}_i\mathbf{C}_i^T)) = f(\|\mathbf{B}_i\mathbf{C}_i^T\|_F). \quad (16)$$

Lastly, since $\|\mathbf{B}_i \mathbf{C}_i^T\|_F = \|\mathbf{B}_i\| \|\mathbf{C}_i\|$ we get

$$\mathcal{R}(\mathbf{X}) \leq \sum_{i=1}^k f(\|\mathbf{B}_i\| \|\mathbf{C}_i\|). \quad (17)$$

To see that equality can be achieved, let $\mathbf{B}_i = \sqrt{\sigma_i(\mathbf{X})} \mathbf{U}_i$ and $\mathbf{C}_i = \sqrt{\sigma_i(\mathbf{X})} \mathbf{V}_i$, where $\mathbf{X} = \sum_{i=1}^k \sigma_i(\mathbf{X}) \mathbf{U}_i \mathbf{V}_i^T$ is the SVD of \mathbf{X} . Then, $\mathbf{B} \mathbf{C}^T = \mathbf{X}$ and $f(\|\mathbf{B}_i\| \|\mathbf{C}_i\|) = f(\sigma_i(\mathbf{X}))$. \square

While the above result allows optimization over the factors \mathbf{B} and \mathbf{C} we note that it yields an objective that is non-differentiable at $\|\mathbf{B}_i\| \|\mathbf{C}_i\| = 0$. Next we reformulate the objective to achieve a differentiable problem formulation.

Corollary 1. *Under the assumptions of Theorem 1, it follows that $\mathcal{R}(\mathbf{X}) = \min_{\mathbf{X}=\mathbf{B}\mathbf{C}^T} \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$, where*

$$\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) = \sum_{i=1}^k f\left(\frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2}\right). \quad (18)$$

If f is differentiable then $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$ is also differentiable.

Proof. By the rule of arithmetic and geometric means

$$\|\mathbf{B}_i\| \|\mathbf{C}_i\| \leq \frac{1}{2}(\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2), \quad (19)$$

with equality if $\|\mathbf{B}_i\| = \|\mathbf{C}_i\|$ which is achieved when $\mathbf{B}_i = \sqrt{\sigma_i(\mathbf{X})} \mathbf{U}_i$ and $\mathbf{C}_i = \sqrt{\sigma_i(\mathbf{X})} \mathbf{V}_i$. Since f is assumed to be non-decreasing, it follows from (13), that $\mathcal{R}(\mathbf{X}) = \min_{\mathbf{X}=\mathbf{B}\mathbf{C}^T} \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$. The differentiability of $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$ is now trivially checked using the chain rule. \square

We are particularly interested in the case (8) since, with this choice, it is known that the global minimizer of (4) is the same as that of $\mu \text{rank}(\mathbf{X}) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2$ if $\|\mathcal{A}\| < 1$, see [13] for a proof. Note that f_μ is a special case of the MCP class [53]. With this choice $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$ is differentiable and the second derivatives are also defined almost everywhere except in the transition $\frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2} = \sqrt{\mu}$ where the function switches from quadratic to constant.

We conclude this section by comparing the shrinking bias of our approach and three others that can also be optimized over the factorization. Theorem 1 makes it possible to compute the global optimizer of $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) + \|\mathbf{B}\mathbf{C}^T - \mathbf{X}_0\|_F^2$ since the equivalent problem $\mathcal{R}(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2$ has closed form solution in the \mathbf{X} -parameterization. It

Table 1: Distance to ground truth (normalized) mean valued over 20 problem instances for different percentages of missing data, missing data patterns and noise levels σ . Best results are marked in bold.

Missing data (%)	PCP [9]	WNNM [27]	Unifying [8]	LpSq [38]	S12L12 [46]	S23L23 [46]	IRNN [12]	APGL [47]	$\ \cdot \ _*$ [4]	\mathcal{R} [36]	Our
Uniform ($\sigma = 0.0$)	0	0.0000	0.0000	0.0000	0.0002	0.0002	0.0000	0.0000	0.1727	0.0000	0.0000
	10	0.0885	0.0028	0.0713	0.0213	0.0309	0.0071	0.0000	0.1998	0.0000	0.0000
	20	0.2720	0.2220	0.1491	0.0170	0.0412	0.0209	0.0000	0.2223	0.0128	0.0000
	30	0.7404	0.4787	0.7499	0.0003	0.0818	0.0895	0.0000	0.0014	0.2897	0.2346
	40	1.0000	0.6097	0.9553	0.1083	0.1666	0.1360	0.0000	0.0017	0.3374	0.2198
50	1.0000	0.7170	1.0000	0.0315	0.1376	0.1001	0.0003	0.0301	0.4266	0.2930	0.0000
Tracking ($\sigma = 0.0$)	0	0.0000	0.0000	0.0000	0.0000	0.0002	0.0002	0.0000	0.1810	0.0000	0.0000
	10	0.3160	0.2734	0.1534	0.0839	0.1296	0.1233	0.0772	0.0834	0.2193	0.0793
	20	0.4877	0.4499	0.3017	0.1650	0.2389	0.2456	0.1010	0.1786	0.3436	0.2494
	30	0.5821	0.5395	0.5486	0.2520	0.3289	0.3160	0.1189	0.2572	0.4299	0.3421
	40	0.7072	0.6317	0.7376	0.2853	0.4084	0.4110	0.1417	0.2913	0.4825	0.5004
50	0.8125	0.7257	0.9521	0.4178	0.4267	0.4335	0.2466	0.4047	0.5754	0.6503	
Tracking ($\sigma = 0.1$)	0	0.0409	0.0207	0.0407	0.0450	0.0437	0.0435	0.0448	0.0191	0.1581	0.0166
	10	0.3157	0.2734	0.1585	0.0848	0.0529	0.0518	0.0625	0.0696	0.2312	0.0488
	20	0.4771	0.4338	0.3480	0.1394	0.0995	0.0982	0.1090	0.1188	0.3109	0.2071
	30	0.5801	0.5225	0.4726	0.2026	0.2468	0.2592	0.1646	0.1993	0.3820	0.3465
	40	0.7122	0.6148	0.8638	0.2225	0.3292	0.3252	0.1357	0.2110	0.4800	0.4599
50	0.7591	0.6819	0.9216	0.4105	0.4883	0.4811	0.3342	0.3639	0.5652	0.5930	

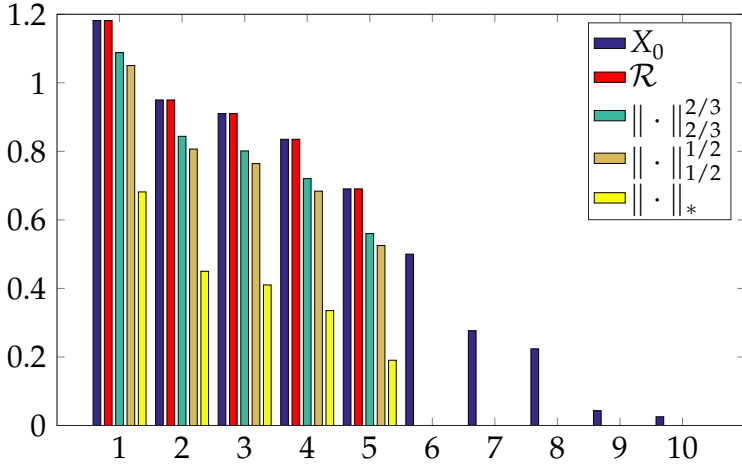


Figure 2: Singular values obtained when minimizing $\|\mathbf{X} - \mathbf{X}_0\|_F^2$ with the four regularizers $\mathcal{R}(\mathbf{X})$ with $f = f_\mu$, $\|\mathbf{X}\|_{1/2}^{1/2}$, $\|\mathbf{X}\|_{2/3}^{2/3}$ and $\|\mathbf{X}\|_*$. Large singular values are left unchanged by \mathcal{R} .

is shown in [36] that with $f = f_\mu$ the solution is obtained by thresholding the singular values at $\sqrt{\mu}$. Similarly, closed form solutions are also available when regularizing $\|\mathbf{X} - \mathbf{X}_0\|_F^2$ with $\|\cdot\|_{1/2}$, $\|\cdot\|_{2/3}$ and $\|\cdot\|_*$ [46]. In Figure 2 we show the singular values obtained when regularizing $\|\mathbf{X} - \mathbf{X}_0\|_F^2$ with these four options, and for comparison the singular values of \mathbf{X}_0 . For all methods we have selected regularization weights as small as possible so that the five smallest singular values are completely suppressed, which minimizes the bias. While all choices, except \mathcal{R} , subtract a part from the singular values that should be retained, the Schatten norms reduce the bias significantly compared to the nuclear norm. For the Schatten norms the bias is larger for singular values that are close to the threshold since the derivative of σ^q , $0 < q < 1$, decreases with increasing σ . For problem instances where there is a clear separation in size between singular values that should be retained and those that should be suppressed, it is likely that this can be done with negligible bias. Since $f'_\mu(\sigma) = 0$ when $\sigma \geq \sqrt{\mu}$ this method does not affect the first five singular values.

3 Overparameterization and Optimality

The results of the previous section show that a global optimizer (\mathbf{B}, \mathbf{C}) of (7) gives a solution $\mathbf{B}\mathbf{C}^T$ which is globally optimal in (4). On the other hand, optimizing (7) over

\mathbf{B} and \mathbf{C} introduces additional stationary points, due to the non-linear parameterization, that are not present in (4). One such point is $(\mathbf{B}, \mathbf{C}) = (0, 0)$ where the gradients of $\|\mathbf{ABC}^T - \mathbf{b}\|^2$ with respect to \mathbf{B} and \mathbf{C} vanish (in contrast to the gradient w.r.t. \mathbf{X}). In this section we show that by overparametrizing, in the sense that we use \mathbf{B} and \mathbf{C} with more columns than the rank of the solution we seek, it is still possible to use properties of (4) to show optimality in (7). We will exclusively use f_μ from (8), assume that \mathbf{B} and \mathbf{C} have $2k$ columns and study locally optimal solutions with $\text{rank}(\mathbf{BC}^T) < k$. The size of \mathbf{B} and \mathbf{C} makes it possible to parametrize line segments between such points and utilize convexity properties, see proof of Theorem 3. The following result (which is proven in the supplementary material) gives conditions that ensure that local minimality in (7) implies that (4) grows in all “low rank” directions.

Theorem 2. *Assume that $(\bar{\mathbf{B}}, \bar{\mathbf{C}}) \in \mathbb{R}^{m \times 2k} \times \mathbb{R}^{n \times 2k}$, where $\bar{\mathbf{B}} = \mathbf{U}\sqrt{\bar{\Sigma}}$ and $\bar{\mathbf{C}} = \mathbf{V}\sqrt{\bar{\Sigma}}$, and $\bar{\mathbf{X}} = \mathbf{U}\bar{\Sigma}\mathbf{V}^T$, is a local minimizer of (7) with $\text{rank}(\bar{\mathbf{X}}) < k$ and let $\mathcal{N}(\mathbf{X}) = \mathcal{R}(\mathbf{X}) + \|\mathbf{AX} - \mathbf{b}\|^2$. Then $\mathcal{R}(\bar{\mathbf{X}}) = \tilde{\mathcal{R}}(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ and the directional derivatives $\mathcal{N}'_{\Delta\mathbf{X}}(\bar{\mathbf{X}})$, where $\Delta\mathbf{X} = \tilde{\mathbf{X}} - \bar{\mathbf{X}}$ and $\text{rank}(\tilde{\mathbf{X}}) \leq k$, are non-negative.*

Note that there can be local minimizers for which $\tilde{\mathcal{R}}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) > \mathcal{R}(\bar{\mathbf{B}}\bar{\mathbf{C}}^T)$ since $\tilde{\mathcal{R}}$ is non-convex. From an algorithmic point of view we can, however, escape such points by taking the current iterate and recompute the factorization of $\bar{\mathbf{B}}\bar{\mathbf{C}}^T$ using SVD. If the SVD of $\bar{\mathbf{B}}\bar{\mathbf{C}}^T = \sum_{i=1}^r \sigma_i \mathbf{U}_i \mathbf{V}_i^T$ we update $\bar{\mathbf{B}}$ and $\bar{\mathbf{C}}$ to $\bar{\mathbf{B}}_i = \sqrt{\sigma_i} \mathbf{U}_i$ and $\bar{\mathbf{C}}_i = \sqrt{\sigma_i} \mathbf{V}_i$, which we know reduces the energy and gives $\tilde{\mathcal{R}}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathcal{R}(\bar{\mathbf{B}}\bar{\mathbf{C}}^T)$.

Theorem 2 allows us to derive optimality conditions using the properties of (4). As a simple example, consider the case where $\|\mathbf{AX}\|^2 \geq \|\mathbf{X}\|^2$, which makes (4) convex [13], and let \mathbf{B} and \mathbf{C} have $2k$ columns. Suppose that we find a local minimizer $(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ fulfilling the assumptions of Theorem 2. Then the derivative along a line segment towards any other low rank matrix is non-decreasing, and therefore $\bar{\mathbf{B}}\bar{\mathbf{C}}^T$ is the global optimum of (4) over the set of matrices with $\text{rank} \leq k$ by convexity.

Below we give a result that goes beyond convexity and applies to the important class [45] of problems that obey the RIP constraint (3). Let \mathcal{A}^* denote the adjoint operator of \mathcal{A} , then:

Theorem 3. *Assume that $(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ is a local minimizer of (7), fulfilling the assumptions of Theorem 2. If the singular values of $\mathbf{Z} = (\mathbf{I} - \mathcal{A}^* \mathcal{A})\bar{\mathbf{B}}\bar{\mathbf{C}}^T + \mathcal{A}^* \mathbf{b}$ fulfill $\sigma_i(\mathbf{Z}) \notin [(1 - \delta_{2k})\sqrt{\mu}, \frac{\sqrt{\mu}}{(1 - \delta_{2k})}]$ then $\bar{\mathbf{B}}\bar{\mathbf{C}}^T$ is the solution of (9) and (10).*

The proof builds on the results of [40] and is given in the supplementary material. The assumption that the singular values of \mathbf{Z} are not too close to the threshold $\sqrt{\mu}$ is a natural restriction which is valid when the noise level is not too large. In case of exact data, i.e. $\mathbf{b} = \mathcal{A}\mathbf{X}_0$, where $\text{rank}(\mathbf{X}_0) = r$ it is trivially fulfilled for any choice of μ

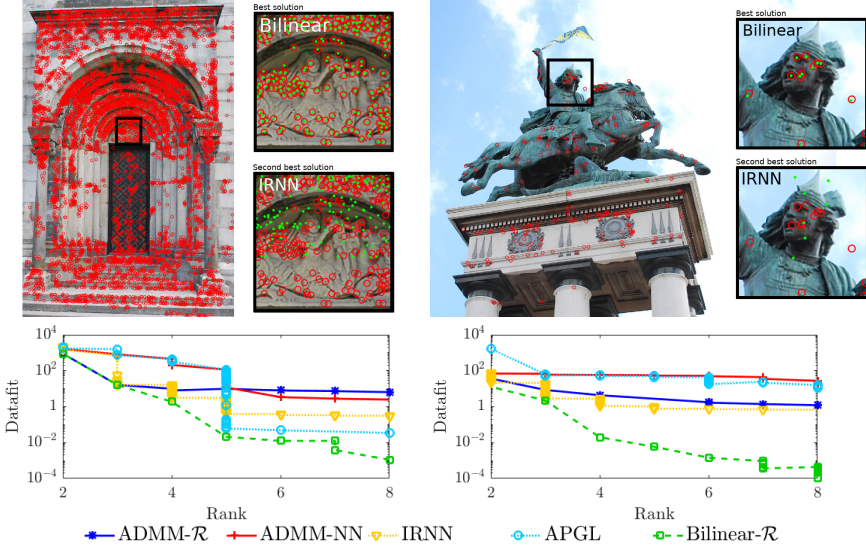


Figure 3: Comparison of reprojection error obtained using the bilinear formulation and ADMM, for datasets *Door* and *Vercingetorix* [41]. The red circles mark the feature points and the green dots the projected image points obtained from the different methods. The best rank 4 solution for the respective method was used. The control parameter $\eta = 0.5$ in both experiments.

such that $\sqrt{\mu} < (1 - \delta_{2k})\sigma_r(\mathbf{X}_0)$ since we have $\mathbf{Z} = \mathbf{X}_0$. For additional details on \mathbf{Z} 's dependence on noise see [14].

The above result is similar in spirit to those of [28, 45], which show that, in the convex case, having $2k$ columns and rank $2k - 1$ is enough to ensure that a local minimizer is global. For the proof in our non-convex case we need rank at most $k - 1$. Presently, it is not clear if our assumption can be relaxed to match that of the convex case or not.

4 An Iterative Reweighted VarPro Algorithm

In this section we give a brief overview of our algorithm for minimizing (7). A more detailed description is given in the supplementary material.

Given a current iterate, $\mathbf{B}^{(t)}$ and $\mathbf{C}^{(t)}$, the first step of our algorithm is to replace the term $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$ with a quadratic function. To do this we note that by the Taylor expansion $f(x) \approx f(x_0) + f'(x_0)(x - x_0)$, minimizing $f(x)$ and $f'(x_0)x$ around x_0 is roughly the same (ignoring constants). Inserting $x_0 = \frac{\|\mathbf{B}_i^{(t)}\|^2 + \|\mathbf{C}_i^{(t)}\|^2}{2}$ and $x =$

$\frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2}$ now gives our approximation

$$\sum_{i=1}^k w_i^{(t)} (\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2) + \|\mathcal{A}\mathbf{B}\mathbf{C}^T - b\|^2, \quad (20)$$

where $w_i^{(t)} = \frac{1}{2} f' \left((\|\mathbf{B}_i^{(t)}\|^2 + \|\mathbf{C}_i^{(t)}\|^2) / 2 \right)$. Here $\mathbf{B}_i^{(t)}$ and $\mathbf{C}_i^{(t)}$ are the i :th columns of $\mathbf{B}^{(t)}$ and $\mathbf{C}^{(t)}$, respectively. Minimizing (20) over \mathbf{C} is now a least squares problem with closed form solution. Inserting this solution into the original problem gives a non-linear problem in \mathbf{B} alone, which is what VarPro solves. We use the so called Ruhe and Wedin (RW2) approximation with a dampening term $\lambda \|\mathbf{B} - \mathbf{B}^{(t)}\|_F^2$, see [31] for details. In each step of the VarPro algorithm we update the weights $w_i^{(t)}$.

As previously mentioned, there can be stationary points for which $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) > \mathcal{R}(\mathbf{B}\mathbf{C}^T)$. In each iteration we therefore take the current iterate and recompute the factorization of $\mathbf{B}^{(t)}\mathbf{C}^{(t)T}$ using SVD. If the SVD of $\mathbf{B}^{(t)}\mathbf{C}^{(t)T} = \sum_{i=1}^r \sigma_i \mathbf{U}_i \mathbf{V}_i^T$ we update $\mathbf{B}^{(t)}$ and $\mathbf{C}^{(t)}$ to $\mathbf{B}_i^{(t)} = \sqrt{\sigma_i} \mathbf{U}_i$ and $\mathbf{C}_i^{(t)} = \sqrt{\sigma_i} \mathbf{V}_i$ which we know reduces the energy and gives $\tilde{\mathcal{R}}(\mathbf{B}^{(t)}, \mathbf{C}^{(t)}) = \mathcal{R}(\mathbf{B}^{(t)}\mathbf{C}^{(t)T})$.

Our approach can be seen as iteratively reweighted nuclear norm minimization [12]; however, our bilinear formulation allows us to use quadratic approximation, thus benefiting from second order convergence in the neighborhood of a local minimum.

5 Experiments

In this section we will show the versatility and strength of the proposed method, focusing on computer vision problems. In Section 5.2 we show an example where state-of-the-art methods fail to achieve a value close to global optimality. We include two more examples of real problems, in the supplementary material: background extraction and photometric stereo. In both cases our method shows superior performance. In the main paper we focus on the trade-off between datafit and rank, but show, in the examples in the supplementary material, the added benefits of convergence speed using the proposed method. This is done by minimizing the same energy with ADMM and the proposed method, in which case the splitting schemes can be tediously slow. In all experiments our proposed method is initialized randomly, with zero mean and unit variance.

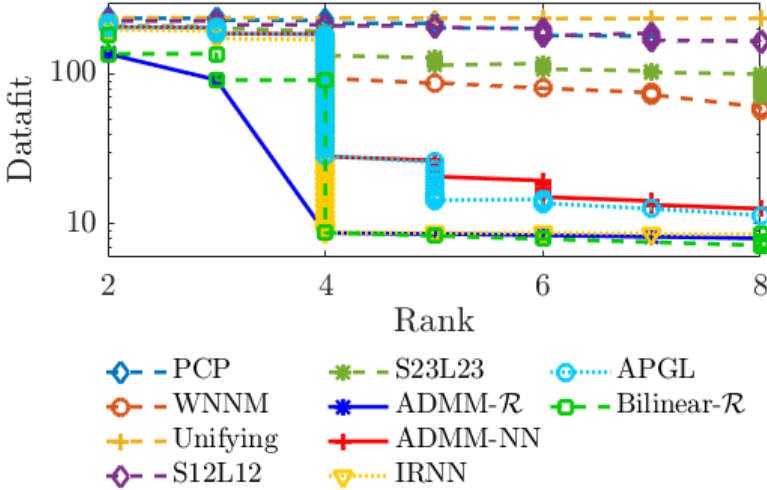


Figure 4: Rank vs datafit for the synthetic experiment in Section 5.1. No true low rank solution using LpSq [38] could be found, regardless of the choice of parameters.

5.1 Synthetic Missing Data Problem

Let \odot denote the Hadamard product, and consider the missing data formulation

$$\min_{\mathbf{X}} \mu \text{rank}(\mathbf{X}) + \|\mathbf{W} \odot (\mathbf{X} - \mathbf{M})\|_F^2, \quad (21)$$

where \mathbf{M} is a measurement matrix and \mathbf{W} a missing data mask with entries $w_{ij} = 1$ if the entry is known, and zero otherwise.

In low-level vision applications such as denoising and image inpainting a uniformly random missing data pattern is often a reasonable approximation of the distribution; however, for structure from motion, the missing data pattern is often highly structured. To this end, we investigate two kinds of patterns: uniformly random and “tracking failure”. In order to construct realistic patterns of tracking failure, we use the method in [35]. This is done by randomly selecting if a track should have missing data (with uniform probability), then select (with uniform probability, starting after the first few frames) in which image tracking failure occurs. If a track is lost, it is not restarted.

We generate random ground truth matrices $\mathbf{M}_0 \in \mathbb{R}^{32 \times 512}$ of rank 4, which can be expressed as $\mathbf{M}_0 = \mathbf{U}\mathbf{V}^T$, where $\mathbf{U} \in \mathbb{R}^{32 \times 4}$ and $\mathbf{V} \in \mathbb{R}^{512 \times 4}$. The entries of \mathbf{U} and \mathbf{V} are normal distributed with zero mean and unit variance. The measurement matrix $\mathbf{M} = \mathbf{M}_0 + \mathbf{N}$, where \mathbf{N} simulates noise and has normal distributed entries with

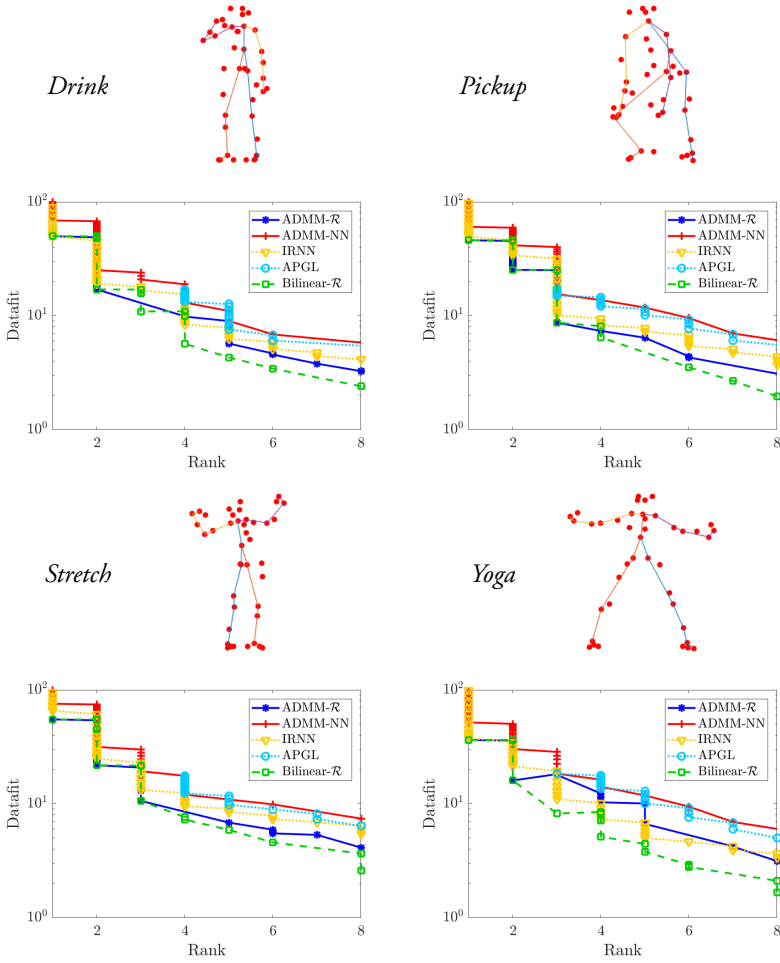


Figure 5: *First and third row*: Example frames from the MOCAP dataset of the *drink*, *pickup*, *stretch* and *yoga* sequences. *Second and last row*: The bilinear method finds the same or a better datafit compared to the other methods for all ranks.

zero mean and variance σ^2 .

Our proposed method is compared to a variety of different methods [4, 8, 9, 12, 27, 36, 38, 46, 47]. For the methods that need an initial estimate of the rank as input, the rank estimation heuristic by Shang *et al.* [46] is used. The regularization parameter is set to $\lambda = \sqrt{\max(m, n)}$, given a sought $m \times n$ matrix, as proposed by [9, 46]. In

case other parameters should be provided, the one recommended from the respective authors have been used. The number of columns, for our proposed method, is set to $k = 8$, *i.e.* twice the rank of the original matrix M_0 . We exclusively use the f_μ regularization (8), and use $\sqrt{\mu} = \lambda$. Since f_μ is a special case of MCP, it is used for IRNN as well. Furthermore, we include the results for regularizing with nuclear norm [4] and f_μ (8) using ADMM, as proposed in [36]. Note that ADMM comes without optimality guarantees, however, it has been shown to work well for several computer vision problems in practice [36, 40]. Several of the compared methods solve the robust PCA problem, thus also include a sparse component, which is not taken into account.

The results are shown in Table 1. Note that most algorithms perform significantly better for the uniformly random missing data pattern, than compared to the structured missing data pattern. Our proposed method outperforms all other methods in this comparison.

Since the final rank of the estimated matrix is not necessarily the same as that of M_0 , we show the rank vs datafit obtained when varying the regularization parameter λ in Figure 4. It is evident from the results that the only candidates that yield an acceptable result for low rank solutions are ADMM with f_μ , IRNN with MCP and our proposed method.

5.2 pOSE: Pseudo Object Space Error

The Pseudo Object Space Error (pOSE) objective combines affine and projective camera models

$$\ell_{\text{OSE}} = \sum_{(i,j) \in \Omega} \|(P_{i,1:2} \tilde{\mathbf{x}}_j - (\mathbf{p}_{i,3}^T \tilde{\mathbf{x}}_j) \mathbf{m}_{i,j})\|^2, \quad (22)$$

$$\ell_{\text{Affine}} = \sum_{(i,j) \in \Omega} \|P_{i,1:2} \tilde{\mathbf{x}}_j - \mathbf{m}_{i,j}\|^2, \quad (23)$$

$$\ell_{\text{pOSE}} = (1 - \eta) \ell_{\text{OSE}} + \eta \ell_{\text{Affine}}, \quad (24)$$

where ℓ_{OSE} is the object space error and ℓ_{Affine} is the affine projection error. Here $P_{i,1:2}$ denotes the first two rows, $\mathbf{p}_{i,3}$ the third row of the i :th camera matrix, and $\tilde{\mathbf{x}}_j$ is the j :th 3D point in homogeneous coordinates. The control parameter $\eta \in [0, 1]$ determines the impact of the respective camera model. This objective was introduced in [30] to be used in a first stage of an initialization-free bundle adjustment pipeline, optimized using VarPro.

The ℓ_{pOSE} objective is linear, and acts on low-rank components \mathbf{P} and \mathbf{X} , which are constrained by $\text{rank}(\mathbf{P}\mathbf{X}^T) = 4$. Instead of enforcing the rank constraint, we replace it

as before with a relaxation. By not enforcing the rank constraint we demonstrate the ability of the methods to make accurate trade-offs between minimizing the rank and fitting the data. Since the objective now becomes more complex, and is no longer compatible with the missing data formulations, only IRNN and APGL are directly applicable, as well as the ADMM approach using f_μ and nuclear norm. We use two real-life datasets with various amounts of camera locations and 3D points: *Door* with 12 images, resulting in seeking a matrix of size 36×8850 and *Vercingetorix* [41] with 69 images, resulting in seeking a matrix of size 207×1148 , both of which have rank 4.¹

As in the synthetic experiment from Section 5.1, the regularization parameter is varied and the resulting rank and datafit is stored and reported in Figure 3. To visualize the results, we considered the best rank 4 approximations, and show the reprojected points and the corresponding measured points obtained from the best method (ours in both cases) and the second best (IRNN in both cases), see Figure 3. As is readily seen by ocular inspection, the rank 4 solution obtained by our proposed method significantly outperforms those of other state-of-the-art methods.

5.3 Non-Rigid Structure from Motion

In this section we test our approach on non-rigid reconstruction (NRSfM) with the CMU Motion Capture (MOCAP) dataset. In NRSfM, the complexity of the deformations are controlled by some mild assumptions of the object shapes. Bregler *et al.* [6] suggested that the set of all possible configurations of the objects are spanned by a low dimensional linear basis of dimension K . In this setting, the non-rigid shapes $\mathbf{X}_i \in \mathbb{R}^{3 \times n}$ can be represented as $\mathbf{X}_i = \sum_{k=1}^K c_{ik} \mathbf{B}_k$, where $\mathbf{B}_k \in \mathbb{R}^{3 \times n}$ are the basis shapes and $c_{ik} \in \mathbb{R}$ the shape coefficients. This way, the matrix \mathbf{X}_i contains the world coordinates of point i , hence the observed image points are given by $\mathbf{x}_i = \mathbf{R}_i \mathbf{X}_i$. We will assume orthographic cameras, *i.e.* $\mathbf{R}_i \in \mathbb{R}^{2 \times 3}$ where $\mathbf{R}_i \mathbf{R}_i^T = \mathbf{I}_2$. As proposed by Dai *et al.* [15], the problem can be turned into a low-rank factorization problem by reshaping and stacking the non-rigid shapes \mathbf{X}_i . Let $\mathbf{X}_i^\sharp \in \mathbb{R}^{1 \times 3n}$ denote the concatenation of the rows in \mathbf{X}_i , and create $\mathbf{X}^\sharp \in \mathbb{R}^{F \times 3n}$ by stacking \mathbf{X}_i^\sharp . This allows us to decompose the matrix \mathbf{X}^\sharp in the low-rank factors $\mathbf{X}^\sharp = \mathbf{C} \mathbf{B}^\sharp$, where $\mathbf{C} \in \mathbb{R}^{F \times K}$ contains the shape coefficients c_{ik} and $\mathbf{B}^\sharp \in \mathbb{R}^{K \times 3n}$ is constructed as \mathbf{X}^\sharp and contains the basis elements.

A suitable objective function is thus given by

$$\mu \text{rank}(\mathbf{X}^\sharp) + \|\mathbf{R} \mathbf{X} - \mathbf{M}\|_F^2, \quad (25)$$

¹The datasets are available here: <http://www.maths.lth.se/matematiklth/personal/calle/dataset/dataset.html>.

where $\mathbf{R} \in \mathbb{R}^{2F \times 3F}$ is a block-diagonal matrix with the camera matrices \mathbf{R}_i on the main diagonal, $\mathbf{X} \in \mathbb{R}^{3F \times n}$ is the concatenation of the 3D points \mathbf{X}_i , and $\mathbf{M} \in \mathbb{R}^{2F \times n}$ is the concatenated observed image points \mathbf{x}_i . By replacing the rank penalty with a relaxation and minimize it using the proposed method and the methods used in the previous section. The regularization parameter is varied for the respective methods in order to obtain a rank 1–8 solution, and the respective datafit is reported in Figure 5, for four different sequences.

In all sequences, the best datafit for each rank level is obtained by our proposed method. IRNN and ADMM using f_μ is able to give the same, or very similar, datafit for lower ranks, but for solutions with rank larger than four our method consistently reports a lower value than the competing state-of-the-art methods.

6 Conclusions

In this paper we presented a unification of bilinear parameterization and rank regularization. Robust penalties for rank regularization has often been used together with splitting schemes, but it has been shown that such methods yield unsatisfactory results for ill-posed problems in several computer vision applications. By using the bilinear formulation, the objective functions become differentiable, and convergence rates in the neighborhood of a local minimum are faster. Furthermore, we showed that theoretical optimality results known from the regularization formulations can be lifted to the bilinear formulation.

Lastly, the generality of the proposed framework allows for a wide range of problems, some of which, have not been amenable by state-of-the-art methods, but have been proven successful using our proposed method.

References

- [1] F. R. Bach. “Convex relaxations of structured matrix factorizations”. In: *CoRR* abs/1309.3117 (2013).
- [2] R. Basri, D. Jacobs, and I. Kemelmacher. “Photometric Stereo with General, Unknown Lighting”. In: *International Journal of Computer Vision* 72.3 (May 2007), pp. 239–257. ISSN: 0920-5691.
- [3] S. Bhojanapalli, B. Neyshabur, and N. Srebro. “Global Optimality of Local Search for Low Rank Matrix Recovery”. In: *Annual Conference in Neural Information Processing Systems (NIPS)*. 2016.

- [4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Found. Trends Mach. Learn.* 3.1 (2011), pp. 1–122.
- [5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [6] C. Bregler, A. Hertzmann, and H. Biermann. “Recovering non-rigid 3D shape from image streams”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2000.
- [7] A. M. Buchanan and A. W. Fitzgibbon. “Damped newton algorithms for matrix factorization with missing data”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2005.
- [8] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [9] E. J. Candès, X. Li, Y. Ma, and J. Wright. “Robust Principal Component Analysis?” In: *J. ACM* 58.3 (2011), 11:1–11:37.
- [10] E. J. Candès and B. Recht. “Exact matrix completion via convex optimization”. In: *Foundations of Computational Mathematics* 9.6 (2009), pp. 717–772.
- [11] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Generalized Nonconvex Nonsmooth Low-Rank Minimization”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014).
- [12] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Nonconvex Nonsmooth Low-Rank Minimization via Iteratively Reweighted Nuclear Norm”. In: *IEEE Transactions on Image Processing* 25 (Oct. 2015).
- [13] M. Carlsson. “On convexification/optimization of functionals including an l2-misfit term”. In: *arXiv preprint arXiv:1609.09378* (2016).
- [14] M. Carlsson, D. Gerosa, and C. Olsson. “An unbiased approach to compressed sensing”. In: *arXiv preprint arXiv:1806.05283* (2018).
- [15] Y. Dai, H. Li, and M. He. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization”. In: *International Journal of Computer Vision* 107.2 (2014), pp. 101–122.
- [16] A. Eriksson and A. Hengel. “Efficient Computation of Robust Weighted Low-Rank Matrix Approximations Using the L_1 Norm”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34.9 (2012), pp. 1681–1690.

-
- [17] J. Fan and R. Li. “Variable selection via nonconcave penalized likelihood and its oracle properties”. In: *Journal of the American Statistical Association* 96.456 (2001), pp. 1348–1360.
- [18] M. Fazel, H. Hindi, and S. P. Boyd. “A rank minimization heuristic with application to minimum order system approximation”. In: *American Control Conference*. 2001.
- [19] J. H. Friedman. “Fast sparse regression and classification”. In: *International Journal of Forecasting* 28.3 (2012), pp. 722–738. ISSN: 0169-2070.
- [20] C. Gao, N. Wang, Q. R. Yu, and Z. Zhang. “A Feasible Nonconvex Relaxation Approach to Feature Selection”. In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2011.
- [21] R. Garg, A. Roussos, and L. Agapito. “A Variational Approach to Video Registration with Subspace Constraints”. In: *International Journal of Computer Vision* 104.3 (2013), pp. 286–314.
- [22] R. Garg, A. Roussos, and L. de Agapito. “Dense Variational Reconstruction of Non-rigid Surfaces from Monocular Video”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.
- [23] R. Ge, C. Jin, and Y. Zheng. “No Spurious Local Minima in Nonconvex Low Rank Problems: A Unified Geometric Analysis”. In: *arXiv preprint arxiv:1704.00708* (2017).
- [24] R. Ge, J. D. Lee, and T. Ma. “Matrix Completion has No Spurious Local Minimum”. In: *Annual Conference on Neural Information Processing Systems (NIPS)*. 2016.
- [25] D. Geman and C. Yang. “Nonlinear Image Recovery with Half-Quadratic Regularization”. In: *IEEE Transactions on Image Processing* 4 (Aug. 1995), pp. 932–946.
- [26] N. Gillis and F. Glinier. “Low-Rank Matrix Approximation with Weights or Missing Data is NP-hard”. In: *SIAM Journal on Matrix Analysis and Applications* 32.4 (2011), pp. 1149–1165.
- [27] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang. “Weighted Nuclear Norm Minimization and Its Applications to Low Level Vision”. In: *International Journal of Computer Vision* 121 (July 2016), pp. 183–208.
- [28] B. D. Haeffele and R. Vidal. “Structured Low-Rank Matrix Factorization: Global Optimality, Algorithms, and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.6 (2020), pp. 1468–1482.

- [29] J. H. Hong and A. Fitzgibbon. “Secrets of Matrix Factorization: Approximations, Numerics, Manifold Optimization and Random Restarts”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.
- [30] J. H. Hong and C. Zach. “pOSE: Pseudo Object Space Error for Initialization-Free Bundle Adjustment”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018, pp. 1876–1885.
- [31] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.
- [32] J. H. Hong, C. Zach, A. Fitzgibbon, and R. Cipolla. “Projective Bundle Adjustment from Arbitrary Initialization Using the Variable Projection Method”. In: *European Conference on Computer Vision (ECCV)*. 2016, pp. 477–493.
- [33] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. “Fast and Accurate Matrix Completion via Truncated Nuclear Norm Regularization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2117–2130.
- [34] M. Krechetov, J. Marecek, Y. Maximov, and M. Takac. “Entropy-Penalized Semidefinite Programming”. In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. International Joint Conferences on Artificial Intelligence Organization, July 2019, pp. 1123–1129.
- [35] V. Larsson and C. Olsson. “Compact Matrix Factorization with Dependent Subspaces”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017, pp. 4361–4370.
- [36] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [37] K. Mohan and M. Fazel. “Iterative reweighted least squares for matrix rank minimization”. In: *Annual Allerton Conference on Communication, Control, and Computing*. 2010, pp. 653–661.
- [38] F. Nie, H. Wang, X. Cai, H. Huang, and C. H. Q. Ding. “Robust Matrix Completion via Joint Schatten p -Norm and l_p -Norm Minimization”. In: *International Conference on Data Mining (ICDM)*. 2012, pp. 566–574.
- [39] T. H. Oh, Y. W. Tai, J. C. Bazin, H. Kim, and I. S. Kweon. “Partial Sum Minimization of Singular Values in Robust PCA: Algorithm and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4 (2016), pp. 744–758.

-
- [40] C. Olsson, M. Carlsson, F. Andersson, and V. Larsson. “Non-Convex Rank/Sparsity Regularization and Local Minima”. In: *Proceedings of the International Conference on Computer Vision* (2017).
- [41] C. Olsson and O. Enqvist. “Stable Structure from Motion for Unordered Image Collections”. In: *Proceedings of the Scandinavian Conference on Image Analysis (SCIA)*. 2011, pp. 524–535.
- [42] M. V. Örnåhag, C. Olsson, and A. Heyden. “Differentiable Fixed-Rank Regularisation using Bilinear Parameterisation”. In: *Proceedings of the British Machine Vision Conference (BMVC)*. 2019.
- [43] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi. “Simultaneously Structured Models With Application to Sparse and Low-Rank Matrices”. In: *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2886–2908.
- [44] S. Oymak, K. Mohan, M. Fazel, and B. Hassibi. “A simplified approach to recovery conditions for low rank matrices”. In: *IEEE International Symposium on Information Theory Proceedings (ISIT)*. 2011, pp. 2318–2322.
- [45] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *SIAM Rev.* 52.3 (Aug. 2010), pp. 471–501.
- [46] F. Shang, J. Cheng, Y. Liu, Z. Luo, and Z. Lin. “Bilinear Factor Matrix Norm Minimization for Robust PCA: Algorithms and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.9 (Sept. 2018), pp. 2066–2080.
- [47] K.-C. Toh and S. Yun. “An Accelerated Proximal Gradient Algorithm for Nuclear Norm Regularized Least Squares Problems”. In: *Pacific Journal of Optimization* 6 (Sept. 2010).
- [48] C. Tomasi and T. Kanade. “Shape and Motion from Image Streams Under Orthography: A Factorization Method”. In: *International Journal of Computer Vision* 9.2 (1992), pp. 137–154.
- [49] M. Uchiyama. “Subadditivity of Eigenvalue Sums”. In: *Proceedings of The American Mathematical Society* 134 (May 2005), pp. 1405–1412.
- [50] N. Wang, T. Yao, J. Wang, and D.-Y. Yeung. “A Probabilistic Approach to Robust Matrix Factorization”. In: *European Conference on Computer Vision (ECCV)*. 2012.
- [51] C. Xu, Z. Lin, and H. Zha. “A Unified Convex Surrogate for the Schatten- p Norm”. In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2017.

- [52] J. Yan and M. Pollefeys. “A Factorization-Based Approach for Articulated Non-rigid Shape, Motion and Kinematic Chain Recovery From Video”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 30.5 (2008), pp. 865–877.
- [53] C.-H. Zhang et al. “Nearly unbiased variable selection under minimax concave penalty”. In: *The Annals of Statistics* 38.2 (2010), pp. 894–942.

Supplementary Material

A Proofs

In this section we present the proofs of Theorems 2 and 3. Our analysis will make use of the differentiable objective

$$\mathcal{D}(\mathbf{B}, \mathbf{C}) := \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}\mathbf{B}\mathbf{C}^T - \mathbf{b}\|^2, \quad (26)$$

the non-convex function

$$\mathcal{N}(\mathbf{X}) := \mathcal{R}(\mathbf{X}) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2, \quad (27)$$

and the convex function

$$\mathcal{C}(\mathbf{X}) = \mathcal{R}(\mathbf{X}) + \|\mathbf{X} - \mathbf{Z}\|_F^2. \quad (28)$$

We will also use the functions

$$\tilde{\mathcal{G}}(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) + \|\mathbf{B}\mathbf{C}^T\|_F^2, \quad (29)$$

$$G(\mathbf{X}) = \mathcal{R}(\mathbf{X}) + \|\mathbf{X}\|_F^2, \quad (30)$$

$$H(\mathbf{X}) = \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2 - \|\mathbf{X}\|_F^2. \quad (31)$$

Note that $\mathcal{D}(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{G}}(\mathbf{B}, \mathbf{C}) + H(\mathbf{B}\mathbf{C}^T)$ and $\mathcal{N}(\mathbf{X}) = G(\mathbf{X}) + H(\mathbf{X})$. Throughout the section we use $f = f_\mu$ with f_μ as in (8) (of the main paper) but for simplicity of notation we will suppress the subscript μ . Furthermore, the subdifferential $\partial G(\mathbf{X})$ of G will be of importance. Let $g(x) = f(|x|) + x^2$. The scalar function g has

$$\partial g(x) = \begin{cases} 2x & |x| \geq \sqrt{\mu} \\ 2\sqrt{\mu}\text{sign}(x) & 0 < |x| \leq \sqrt{\mu} \\ 2\sqrt{\mu}[-1, 1] & x = 0 \end{cases}. \quad (32)$$

The following lemma shows how to compute ∂G for the matrix case using ∂g .

Lemma 1. *The subdifferential of $G(\mathbf{X})$ is given by*

$$\partial G(\mathbf{X}) = \{\mathbf{U}\partial g(\mathbf{\Sigma})\mathbf{V}^T + \mathbf{M} : \sigma_1(\mathbf{M}) \leq 2\sqrt{\mu}, \\ \mathbf{U}^T \mathbf{M} = 0 \text{ and } \mathbf{M}\mathbf{V}^T = 0\} \quad (33)$$

where $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is the SVD and $\partial g(\mathbf{\Sigma})$ is the matrix of same size as $\mathbf{\Sigma}$ with diagonal elements $\partial g(\sigma_i)$.

Next we give the stationary point conditions for \mathcal{D} that are needed for proving Theorem 2.

Lemma 2. *Let $\mathbf{B} = \mathbf{U}\sqrt{\boldsymbol{\Sigma}}$, $\mathbf{C} = \mathbf{V}\sqrt{\boldsymbol{\Sigma}}$ and $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$. If (\mathbf{B}, \mathbf{C}) is a stationary point of \mathcal{D} , then*

$$0 = \mathbf{B}\partial G(\boldsymbol{\Sigma}) + \nabla H(\mathbf{B}\mathbf{C}^T)\mathbf{C}, \quad (34)$$

$$0 = \partial G(\boldsymbol{\Sigma})\mathbf{C}^T + \mathbf{B}^T\nabla H(\mathbf{B}\mathbf{C}^T). \quad (35)$$

We are now ready to prove Theorem 2.

Proof of Theorem 2. Let $\bar{\mathbf{X}} = \bar{\mathbf{B}}\bar{\mathbf{C}}^T$, $\tilde{\mathbf{X}} = \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ and $\Delta\mathbf{X} = \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T - \bar{\mathbf{B}}\bar{\mathbf{C}}^T$. We first note that the limit

$$\mathcal{N}'_{\Delta\mathbf{X}}(\bar{\mathbf{X}}) = \lim_{t \searrow 0} \frac{\mathcal{N}(\bar{\mathbf{X}} + t\Delta\mathbf{X}) - \mathcal{N}(\bar{\mathbf{X}})}{t}, \quad (36)$$

exists since \mathcal{N} is a sum of a finite convex function G and a differentiable function H . Our goal is now to show that the limit is non-negative. Suppose that we can find a factorization $\mathbf{B}(t)\mathbf{C}(t)^T = \bar{\mathbf{X}} + t\Delta\mathbf{X}$, such that $\mathcal{R}(\bar{\mathbf{X}} + t\Delta\mathbf{X}) = \tilde{\mathcal{R}}(\mathbf{B}(t), \mathbf{C}(t))$, $(\mathbf{B}(t), \mathbf{C}(t))$ is continuous and $(\mathbf{B}(0), \mathbf{C}(0)) = (\bar{\mathbf{B}}, \bar{\mathbf{C}})$. Then for small enough t we have

$$\mathcal{N}(\bar{\mathbf{X}} + t\Delta\mathbf{X}) - \mathcal{N}(\bar{\mathbf{X}}) = \mathcal{D}(\mathbf{B}(t), \mathbf{C}(t)) - \mathcal{D}(\bar{\mathbf{B}}, \bar{\mathbf{C}}). \quad (37)$$

This quantity is clearly non-negative since $(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ is a local minimizer of \mathcal{D} , which would prove that the limit (36) is non-negative. It is not difficult to see that this can be done when the two matrices $\bar{\mathbf{X}}$ and $\tilde{\mathbf{X}}$ have singular value decompositions with the same \mathbf{U} and \mathbf{V} . In what follows we will first show that all other cases can be reduced so that the matrices are of this form. When this is done we proceed to construct the factorization $\mathbf{B}(t)\mathbf{C}(t)^T$ which completes the proof.

The directional derivatives can be computed using the sub-differential

$$\mathcal{N}'_{\Delta\mathbf{X}} = \max_{2\mathbf{Z} \in \partial G(\bar{\mathbf{B}}\bar{\mathbf{C}}^T)} \langle 2\mathbf{Z}, \Delta\mathbf{X} \rangle + \langle \nabla H(\bar{\mathbf{B}}\bar{\mathbf{C}}^T), \Delta\mathbf{X} \rangle. \quad (38)$$

By Lemma 1, the first term becomes

$$\begin{aligned} \langle \mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T + \mathbf{M}, \Delta\mathbf{X} \rangle &= \langle \mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T, \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \rangle \\ &\quad + \langle \mathbf{M}, \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \rangle \\ &\quad - \langle \mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T, \bar{\mathbf{B}}\bar{\mathbf{C}}^T \rangle. \end{aligned} \quad (39)$$

The columns of $\tilde{\mathbf{B}}$ can be written as a linear combination of the columns in $\bar{\mathbf{B}}$ and those of a matrix $\bar{\mathbf{B}}_{\perp}$ with at most k columns that are perpendicular to $\bar{\mathbf{B}}$. Similarly, the columns of $\tilde{\mathbf{C}}$ can be written as a linear combination of the columns in $\bar{\mathbf{C}}$ and those of a matrix $\bar{\mathbf{C}}_{\perp}$ with at most k columns that are perpendicular to $\bar{\mathbf{C}}$. Therefore, we may write

$$\begin{aligned}\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T &= [\bar{\mathbf{B}} \quad \bar{\mathbf{B}}_{\perp}] \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{C}}^T \\ \bar{\mathbf{C}}_{\perp}^T \end{bmatrix} \\ &= \bar{\mathbf{B}}\mathbf{K}_{11}\mathbf{C}^T + \bar{\mathbf{B}}\mathbf{K}_{12}\bar{\mathbf{C}}_{\perp}^T \\ &\quad + \bar{\mathbf{B}}_{\perp}\mathbf{K}_{21}\bar{\mathbf{C}}^T + \bar{\mathbf{B}}_{\perp}\mathbf{K}_{22}\bar{\mathbf{C}}_{\perp}^T,\end{aligned}\tag{40}$$

where $\bar{\mathbf{B}}^T\bar{\mathbf{B}}_{\perp} = 0$ and $\bar{\mathbf{C}}^T\bar{\mathbf{C}}_{\perp} = 0$. Our goal is now to show that the terms \mathbf{K}_{12} and \mathbf{K}_{21} and the off diagonal elements of \mathbf{K}_{11} vanish from (38) and can be assumed to be zero.

For the last term of (39) we have

$$\begin{aligned}\langle \mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T, \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \rangle &= \langle \partial G(\boldsymbol{\Sigma}), \mathbf{U}^T\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T\mathbf{V} \rangle \\ &= \langle \partial G(\boldsymbol{\Sigma}), \boldsymbol{\Sigma} \rangle,\end{aligned}\tag{41}$$

which is clearly independent of $\tilde{\mathbf{B}}$ and $\tilde{\mathbf{C}}$. The first term of (39) reduces to

$$\begin{aligned}\langle \mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T, \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \rangle &= \langle \mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T, \bar{\mathbf{B}}\mathbf{K}_{11}\bar{\mathbf{C}}^T \rangle \\ &= \langle \bar{\mathbf{B}}^T\mathbf{U}\partial G(\boldsymbol{\Sigma})\mathbf{V}^T\bar{\mathbf{C}}, \mathbf{K}_{11} \rangle \\ &= \langle \boldsymbol{\Sigma}\partial G(\boldsymbol{\Sigma}), \mathbf{K}_{11} \rangle.\end{aligned}\tag{42}$$

Note that the off diagonal elements of \mathbf{K}_{11} vanish from this expression since $\boldsymbol{\Sigma}\partial G(\boldsymbol{\Sigma})$ is diagonal. Similarly, the second term of (39) reduces to

$$\langle \mathbf{M}, \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \rangle = \langle \mathbf{M}, \bar{\mathbf{B}}_{\perp}\mathbf{K}_{22}\bar{\mathbf{C}}_{\perp}^T \rangle.\tag{43}$$

We now consider the second term of (38)

$$\begin{aligned}\langle \nabla H(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T), \Delta \mathbf{X} \rangle &= \\ \langle \nabla H(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T), \bar{\mathbf{B}}\mathbf{K}_{11}\bar{\mathbf{C}}^T + \bar{\mathbf{B}}\mathbf{K}_{12}\bar{\mathbf{C}}_{\perp}^T \\ &\quad + \bar{\mathbf{B}}_{\perp}\mathbf{K}_{21}\bar{\mathbf{C}}^T + \bar{\mathbf{B}}_{\perp}\mathbf{K}_{22}\bar{\mathbf{C}}_{\perp}^T - \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \rangle.\end{aligned}\tag{44}$$

For the first term we have

$$\begin{aligned}\langle \nabla H(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T), \bar{\mathbf{B}}\mathbf{K}_{11}\bar{\mathbf{C}}^T \rangle &= \langle \nabla H(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T)\tilde{\mathbf{C}}, \bar{\mathbf{B}}\mathbf{K}_{11} \rangle \\ &= -\langle \tilde{\mathbf{B}}\partial G(\boldsymbol{\Sigma}), \bar{\mathbf{B}}\mathbf{K}_{11} \rangle \\ &= -\langle \bar{\mathbf{B}}^T\tilde{\mathbf{B}}\partial G(\boldsymbol{\Sigma}), \mathbf{K}_{11} \rangle \\ &= -\langle \boldsymbol{\Sigma}\partial G(\boldsymbol{\Sigma}), \mathbf{K}_{11} \rangle.\end{aligned}\tag{45}$$

Again the off diagonal elements of \mathbf{K}_{11} vanish. For the second term of (44) we have

$$\begin{aligned} \langle \nabla H(\bar{\mathbf{B}}\bar{\mathbf{C}}^T), \bar{\mathbf{B}}\mathbf{K}_{12}\mathbf{C}_{\perp}^T \rangle &= \langle \mathbf{B}^T \nabla H(\mathbf{B}\mathbf{C}^T), \mathbf{K}_{12}\mathbf{C}_{\perp}^T \rangle \\ &= -\langle \partial G(\boldsymbol{\Sigma})\bar{\mathbf{C}}^T, \mathbf{K}_{12}\bar{\mathbf{C}}_{\perp} \rangle \\ &= -\langle \partial G(\boldsymbol{\Sigma})\bar{\mathbf{C}}^T \bar{\mathbf{C}}_{\perp}, \mathbf{K}_{12} \rangle = 0. \end{aligned} \quad (46)$$

Similarly, the third term is $\langle \nabla H(\bar{\mathbf{B}}\bar{\mathbf{C}}^T), \bar{\mathbf{B}}_{\perp}\mathbf{K}_{21}\bar{\mathbf{C}}^T \rangle = 0$. Thus

$$\begin{aligned} \langle \nabla H(\bar{\mathbf{B}}\bar{\mathbf{C}}^T), \Delta \mathbf{X} \rangle &= \langle \nabla H(\bar{\mathbf{B}}\bar{\mathbf{C}}^T), \bar{\mathbf{B}}_{\perp}^T \mathbf{K}_{22} \bar{\mathbf{C}}_{\perp}^T \rangle \\ &\quad - \langle \boldsymbol{\Sigma} \partial G(\boldsymbol{\Sigma}), \mathbf{K}_{11} \rangle \\ &\quad - \langle \nabla H(\bar{\mathbf{B}}\bar{\mathbf{C}}^T), \bar{\mathbf{B}}\bar{\mathbf{C}}^T \rangle. \end{aligned} \quad (47)$$

Summarizing we see that we have now proven that all the terms in (39) are independent of \mathbf{K}_{12} , \mathbf{K}_{21} as well as the off diagonal terms of \mathbf{K}_{11} . They therefore do not affect the value of $\mathcal{N}'_{\Delta \mathbf{X}}$ and can be assumed to be zero. We can now write $\Delta \mathbf{X}$ as

$$\Delta \mathbf{X} = [\mathbf{U} \quad \mathbf{U}_{\perp}] \begin{bmatrix} (\mathbf{D} - \mathbf{I})\boldsymbol{\Sigma} & 0 \\ 0 & \tilde{\boldsymbol{\Sigma}} \end{bmatrix} \begin{bmatrix} \mathbf{V}^T \\ \mathbf{V}_{\perp}^T \end{bmatrix}, \quad (48)$$

where \mathbf{D} are the diagonal elements of \mathbf{K}_{11} and $\mathbf{U}_{\perp} \tilde{\boldsymbol{\Sigma}} \mathbf{V}_{\perp}^T$ is the SVD of $\bar{\mathbf{B}}_{\perp} \mathbf{K}_{22} \bar{\mathbf{C}}_{\perp}^T$. Note that $\mathbf{U}_{\perp}^T \mathbf{U} = 0$ since \mathbf{U} and \mathbf{U}_{\perp} span orthogonal subspaces. Similarly $\mathbf{V}_{\perp}^T \mathbf{V} = 0$.

We now consider the directional derivative (36) with $\bar{\mathbf{B}} = \mathbf{U}\sqrt{\boldsymbol{\Sigma}}$, $\bar{\mathbf{C}} = \mathbf{V}\sqrt{\boldsymbol{\Sigma}}$. It is clear that for small t the matrix $\tilde{\mathbf{X}} + t\Delta \mathbf{X}$ has the singular value decomposition

$$[\mathbf{U} \quad \mathbf{U}_{\perp}] \begin{bmatrix} ((1-t)\mathbf{I} + t\mathbf{D})\boldsymbol{\Sigma} & 0 \\ 0 & t\tilde{\boldsymbol{\Sigma}} \end{bmatrix} \begin{bmatrix} \mathbf{V}^T \\ \mathbf{V}_{\perp}^T \end{bmatrix}. \quad (49)$$

We now let

$$\mathbf{B}(t) = [\mathbf{U} \quad \mathbf{U}_{\perp}] \sqrt{\begin{bmatrix} ((1-t)\mathbf{I} + t\mathbf{D})\boldsymbol{\Sigma} & 0 \\ 0 & t\tilde{\boldsymbol{\Sigma}} \end{bmatrix}}, \quad (50)$$

$$\mathbf{C}(t) = [\mathbf{V} \quad \mathbf{V}_{\perp}] \sqrt{\begin{bmatrix} ((1-t)\mathbf{I} + t\mathbf{D})\boldsymbol{\Sigma} & 0 \\ 0 & t\tilde{\boldsymbol{\Sigma}} \end{bmatrix}}. \quad (51)$$

Then, we clearly have $\tilde{\mathcal{R}}(\mathbf{B}(t), \mathbf{C}(t)) = \mathcal{R}(\mathbf{X} + t\Delta \mathbf{X})$ for small enough t , which completes the proof. \square

Next we will prove Theorem 3. Our results build on those of [8] and we remind the reader that we exclusively use $f_\mu(\sigma) = \mu - \max(\sqrt{\mu} - \sigma, 0)^2$ throughout this section, but suppress the subscript μ . We will use the fact that the directional derivatives in a local minimum are non-negative for all low rank directions to show that $(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ minimizes the non-convex \mathcal{N} over matrices of rank $< k$ in Theorem 3. For this we will need the following result:

Lemma 3. *If $\bar{\mathbf{X}}$ is a solution to $\min_{\text{rank}(\mathbf{X}) \leq k} \mathcal{C}(\mathbf{X})$ with $\text{rank}(\bar{\mathbf{X}}) < k$ and the singular values of \mathbf{Z} fulfill $\sigma_i(\mathbf{Z}) \notin [(1 - \delta_{2k})\sqrt{\mu}, \frac{\sqrt{\mu}}{(1 - \delta_{2k})}]$ then $\bar{\mathbf{X}}$ also solves $\min_{\mathbf{X}} \mathcal{C}(\mathbf{X})$.*

Proof of Lemma 3. By von Neumann's trace theorem it is easy to see that the problem $\min_{\text{rank}(\mathbf{X}) \leq k} \mathcal{C}(\mathbf{X})$ reduces to a minimization over the singular values of \mathbf{X} . We should thus find $\sigma_i(\bar{\mathbf{X}})$ such that

$$\sum_{i=1}^n \underbrace{-\max(\sqrt{\mu} - \sigma_i(\mathbf{X}), 0)^2 + (\sigma_i(\mathbf{X}) - \sigma_i(\mathbf{Z}))^2}_{:=g_i(\sigma_i(\mathbf{X}))} \quad (52)$$

is minimized and at most k singular values are non-zero. The unconstrained minimizers of g_i can be written down in closed form: If $0 \leq \sqrt{\mu} < \sigma_i(\mathbf{Z})$ then $\sigma_i(\mathbf{X}) = \sigma_i(\mathbf{Z})$ is optimal giving $g_i(\sigma_i(\mathbf{X})) = 0$. If $0 \leq \sigma_i(\mathbf{Z}) < \sqrt{\mu}$ then $\sigma_i(\mathbf{X}) = 0$ is optimal giving $g_i(\sigma_i(\mathbf{X})) = -\mu + \sigma_i(\mathbf{Z})^2$. Hence for any solution of $\min_{\text{rank}(\mathbf{X}) \leq k} \mathcal{C}(\mathbf{X})$ we have $\sigma_i(\mathbf{X}) = 0$ if $0 \leq \sigma_i(\mathbf{Z}) \leq \sqrt{\mu}$. There are now two cases:

1. If $\sigma_{k+1}(\mathbf{Z}) < \sqrt{\mu}$ then the sequence of unconstrained minimizers has at most k non-zero values. Thus, in this case the resulting $\bar{\mathbf{X}}$ solves both $\min_{\mathbf{X}} \mathcal{C}(\mathbf{X})$ and $\min_{\text{rank}(\mathbf{X}) \leq k} \mathcal{C}(\mathbf{X})$.
2. If $\sigma_{k+1} > \sqrt{\mu}$ we will not be able to select $\sigma_i(\mathbf{X}) = \sigma_i(\mathbf{Z})$ for all i where $0 \leq \sqrt{\mu} < \sigma_i(\mathbf{Z})$. Choosing $\sigma_i(\mathbf{X}) = 0$ gives $g_i(0) = -\mu + \sigma_i(\mathbf{Z})^2 < 0$. Since $\sigma_i(\mathbf{Z})$ is decreasing with i it is clear that the smallest value is obtained when selecting $\sigma_i(\mathbf{X}) = \sigma_i(\mathbf{Z})$ for $i = 1, \dots, k$.

We now conclude that if $\text{rank}(\bar{\mathbf{X}}) < k$ then we are in case 1 and therefore $\bar{\mathbf{X}}$ solves the unconstrained problem. \square

We are now ready to give the proof of Theorem 3.

Proof of Theorem 3. Since \mathcal{C} and \mathcal{N} has the same subdifferential (see [6]) at $\bar{\mathbf{X}} = \bar{\mathbf{B}}\bar{\mathbf{C}}^T$ it is clear that the directional derivatives $\mathcal{C}'_{\Delta\mathbf{X}}(\bar{\mathbf{X}}) = \mathcal{N}'_{\Delta\mathbf{X}}(\bar{\mathbf{X}}) \geq 0$, where $\Delta\mathbf{X} =$

$\tilde{\mathbf{X}} - \bar{\mathbf{B}}\bar{\mathbf{C}}^T$ and $\text{rank}(\tilde{\mathbf{X}}) \leq k$. By convexity of \mathcal{C} it is then also clear that

$$\bar{\mathbf{B}}\bar{\mathbf{C}}^T \in \underset{\text{rank}(\mathbf{X}) \leq k}{\text{arg min}} \mathcal{C}(\mathbf{X}). \quad (53)$$

Since $\text{rank}(\bar{\mathbf{B}}\bar{\mathbf{C}}^T) < k$, $\bar{\mathbf{B}}\bar{\mathbf{C}}^T$ is also the unrestricted global minimizer of $\mathcal{C}(\mathbf{X})$ according to Lemma 3. By Lemma 3.1 of [8] it is then a stationary point of $\mathcal{N}(\mathbf{X})$.

What remains now is to prove that $\tilde{\mathbf{X}} = \bar{\mathbf{B}}\bar{\mathbf{C}}^T$ is a global minimizer of \mathcal{N} over all line segments $\tilde{\mathbf{X}} + t\Delta\mathbf{X}$. This can be done by estimating the growth of the directional derivatives along such lines. For this purpose we consider the functions G and H defined as in (30) and (31). Note that $\tilde{\mathbf{X}}$ is a stationary point of $\mathcal{N}(\mathbf{X}) = G(\mathbf{X}) + H(\mathbf{X})$ if and only if $-\nabla H(\tilde{\mathbf{X}}) = 2\mathbf{Z} \in \partial G(\tilde{\mathbf{X}})$.

Since $\nabla H(\tilde{\mathbf{X}} + t\Delta\mathbf{X}) - \nabla H(\tilde{\mathbf{X}}) = t\nabla H(\Delta\mathbf{X}) = 2t(\mathcal{A}^* \mathcal{A}\Delta\mathbf{X} - \Delta\mathbf{X})$ we have

$$\langle \nabla H(\tilde{\mathbf{X}} + t\Delta\mathbf{X}) - \nabla H(\tilde{\mathbf{X}}), t\Delta\mathbf{X} \rangle = 2t^2(\|\mathcal{A}\Delta\mathbf{X}\|_F^2 - \|\Delta\mathbf{X}\|_F^2), \quad (54)$$

and due to RIP $\|\mathcal{A}\Delta\mathbf{X}\|_F^2 - \|\Delta\mathbf{X}\|_F^2 \geq -\delta_{2r}\|\Delta\mathbf{X}\|_F^2$. From Corollary 4.2 of [8] we see that for any $2\mathbf{Z}' \in \partial G(\tilde{\mathbf{X}} + t\Delta\mathbf{X})$ we have

$$\langle \mathbf{Z}' - \mathbf{Z}, t\Delta\mathbf{X} \rangle > t^2\delta_{2r}\|\Delta\mathbf{X}\|_F^2, \quad (55)$$

as long as $t \neq 0$. Since $G'_{\Delta\mathbf{X}}(\mathbf{X}) = \max_{2\mathbf{Z} \in \partial G(\mathbf{X})} \langle 2\mathbf{Z}, \Delta\mathbf{X} \rangle$, $H'_{\Delta\mathbf{X}}(\mathbf{X}) = \langle \nabla H(\mathbf{X}), \Delta\mathbf{X} \rangle$ and $2\mathbf{Z} + \nabla H'(\tilde{\mathbf{X}}) = 0$ we get

$$\mathcal{N}'_{\Delta\mathbf{X}}(\tilde{\mathbf{X}} + t\Delta\mathbf{X}) \geq \langle 2\mathbf{Z}' + \nabla H(\tilde{\mathbf{X}} + t\Delta\mathbf{X}), \Delta\mathbf{X} \rangle > 0 \quad (56)$$

This shows that $\tilde{\mathbf{X}}$ solves (9). That $\tilde{\mathbf{X}}$ also solves (10) is now a consequence of the fact that $\mathcal{R}(\mathbf{X}) \leq \mu \text{rank}(\mathbf{X})$ with equality if \mathbf{X} have no singular values in the interval $(0, \sqrt{\mu}]$. Note that $\tilde{\mathbf{X}}$ is the unrestricted minimizer of $\mathcal{C}(\mathbf{X})$, where the singular values of \mathbf{Z} fulfill $\sigma_i(\mathbf{Z}) \notin \left[(1 - \delta_{2k})\sqrt{\mu}, \frac{\sqrt{\mu}}{1 - \delta_{2k}} \right]$. Since the solution to this problem is hard thresholding $\tilde{\mathbf{X}}$ has no singular values in $\left(0, \frac{\sqrt{\mu}}{1 - \delta_{2k}} \right] \supset (0, \sqrt{\mu}]$. \square

For completeness we give the proofs that were previously omitted.

Proof of Lemma 1. With some abuse of notation we define the function $g : \mathbb{R}^n \rightarrow \mathbb{R}$ by $g(\mathbf{x}) = \sum_{i=1}^n g(x_i)$, where x_i , $i = 1, \dots, n$ are the elements of \mathbf{x} and $g(x) = f(|x|) + x^2$. The function g is an absolutely symmetric convex function and G can be written $G(\mathbf{X}) = g \circ \sigma(\mathbf{X})$, where $\sigma(\mathbf{X})$ is the vector of singular values of \mathbf{X} . Then according to [7] the matrix $\mathbf{Y} \in \partial G(\mathbf{X})$ if and only if $\mathbf{Y} = \mathbf{U}' \text{diag}(\partial g \circ \sigma(\mathbf{X})) \mathbf{V}'^T$ when $\mathbf{X} = \mathbf{U}' \text{diag}(\sigma(\mathbf{X})) \mathbf{V}'^T$. (Here we use the full SVD with square orthogonal

matrices \mathbf{U}' and \mathbf{V}' .) Now given a thin SVD $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ all possible full SVD's of \mathbf{X} can be written

$$\mathbf{X} = [\mathbf{U} \quad \mathbf{U}_\perp] \begin{bmatrix} \mathbf{\Sigma} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}^T \\ \mathbf{V}_\perp^T \end{bmatrix}, \quad (57)$$

where \mathbf{U}_\perp and \mathbf{V}_\perp are singular vectors corresponding to singular values that are zero. Note that \mathbf{U}_\perp and \mathbf{V}_\perp are not uniquely defined since their corresponding singular values are all zero. Therefore we get

$$\begin{aligned} \mathbf{Y} &= [\mathbf{U}' \quad \mathbf{U}_\perp] \begin{bmatrix} \partial g(\mathbf{\Sigma}) & 0 \\ 0 & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{V}'^T \\ \mathbf{V}_\perp^T \end{bmatrix} \\ &= \mathbf{U}' \partial g(\mathbf{\Sigma}) \mathbf{V}'^T + \mathbf{U}_\perp \mathbf{D} \mathbf{V}_\perp^T, \end{aligned} \quad (58)$$

where \mathbf{D} is a diagonal matrix with elements in $2\sqrt{\mu}[-1, 1]$. It is clear that $\sigma_1(\mathbf{U}_\perp \mathbf{D} \mathbf{V}_\perp^T) = \sigma_1(\mathbf{D}) \leq 2\sqrt{\mu}$. Furthermore, since \mathbf{U}_\perp and \mathbf{V}_\perp can be any orthogonal bases of the spaces perpendicular to the column and row spaces of \mathbf{X} , it is clear that any matrix \mathbf{M} fulfilling $\mathbf{U}^T \mathbf{M} = 0$, $\mathbf{M} \mathbf{V} = 0$ and $\sigma_1(\mathbf{M}) \leq 2\sqrt{\mu}$ can be written $\mathbf{M} = \mathbf{U}_\perp \mathbf{D} \mathbf{V}_\perp^T$, hence

$$\begin{aligned} \partial G(\mathbf{X}) &= \{\mathbf{U} \partial g(\mathbf{\Sigma}) \mathbf{V}^T + \mathbf{M} : \sigma_1(\mathbf{M}) \leq 2\sqrt{\mu}, \\ &\quad \mathbf{U}^T \mathbf{M} = 0, \mathbf{M} \mathbf{V} = 0\}. \end{aligned} \quad (59)$$

□

Proof of Lemma 2. The gradients of \tilde{G} are given by

$$\nabla_{\mathbf{B}} \tilde{G}(\mathbf{B}, \mathbf{C}) = \nabla_{\mathbf{B}} (\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})) + \nabla_{\mathbf{B}} (\|\mathbf{B}\mathbf{C}^T\|_F^2). \quad (60)$$

For the first term we get

$$\nabla_{\mathbf{B}_i} \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) = f' \left(\frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2} \right) \mathbf{B}_i. \quad (61)$$

With $\mathbf{B} = \mathbf{U}\sqrt{\mathbf{\Sigma}}$ and $\mathbf{C} = \mathbf{V}\sqrt{\mathbf{\Sigma}}$ we get

$$\nabla_{\mathbf{B}} \tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) = \mathbf{B} \begin{bmatrix} f'(\sigma_1) & 0 & \dots \\ 0 & f'(\sigma_2) & \dots \\ \vdots & \vdots & \ddots \end{bmatrix} = \mathbf{B} f'(\mathbf{\Sigma}), \quad (62)$$

which gives

$$\nabla_{\mathbf{B}} \tilde{G}(\mathbf{B}, \mathbf{C}) = \mathbf{B}f'(\boldsymbol{\Sigma}) + 2\mathbf{B}\mathbf{C}^T\mathbf{C} = \mathbf{B}(f'(\boldsymbol{\Sigma}) + 2\boldsymbol{\Sigma}). \quad (63)$$

For a non-zero σ we have $\partial g(\sigma) = \{f'(\sigma) + 2\sigma\}$ and therefore

$$\nabla_{\mathbf{B}} \tilde{G}(\mathbf{B}, \mathbf{C}) = \mathbf{B}(\partial G(\boldsymbol{\Sigma})), \quad (64)$$

where $g(\mathbf{X}) = \mathcal{R}_\mu(\mathbf{X}) + \|\mathbf{X}\|_F^2$. Similarly we get

$$\nabla_{\mathbf{C}} \tilde{G}(\mathbf{B}, \mathbf{C}) = \mathbf{C}(\partial G(\boldsymbol{\Sigma})). \quad (65)$$

If (\mathbf{B}, \mathbf{C}) is a stationary point then

$$0 = \mathbf{B}\partial G(\boldsymbol{\Sigma}) + \nabla H(\mathbf{B}\mathbf{C}^T)\mathbf{C}, \quad (66)$$

$$0 = \mathbf{C}\partial G(\boldsymbol{\Sigma}) + (\nabla H(\mathbf{B}\mathbf{C}^T))^T\mathbf{B}. \quad (67)$$

The second equation can be re-written to the form stated in the lemma. \square

B Implementation Details

In this section we present some more details on our Iteratively Reweighted VarPro approach. Recall that our approach consists of three main steps. In the first step we make a quadratic approximation (20) of the regularization term by replacing $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C})$ with $\sum_{i=1}^k w_i^{(t)} (\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2)$ as described in Section 4.

In the second step we apply one step of VarPro with the Ruhe Wedin approximation, see [4] for details on the implementation. VarPro uses Jacobians with respect to both the \mathbf{B} and \mathbf{C} parameters. In our case we have two terms that needs to be linearized. The regularization term can be written

$$\|\text{diag}(\mathbf{w}^{(t)})\mathbf{B}\|_F^2 + \|\text{diag}(\mathbf{w}^{(t)})\mathbf{C}\|_F^2, \quad (68)$$

where $\text{diag}(\mathbf{w}^{(t)})$ is a diagonal matrix with the weights $w_i^{(t)}$ in the diagonal. The residuals $\text{diag}(\mathbf{w}^{(t)})\mathbf{B}$ are already linear and by column stacking the variables we can write them as $J_{\mathbf{B}}^{\text{reg}}\mathbf{b}$, where \mathbf{b} is a column stacked version of \mathbf{B} . If \mathbf{B} has k columns the matrix $J_{\mathbf{B}}^{\text{reg}}$ will consist of k copies of the matrix $\text{diag}(\mathbf{w}^{(t)})$. Additionally, each row of $J_{\mathbf{B}}^{\text{reg}}$ has only one non-zero element making the matrix extremely sparse. Similarly, we obtain the contribution due to the second bilinear factor \mathbf{C} , which can be written as $J_{\mathbf{C}}^{\text{reg}}\mathbf{c}$. Here we use $\mathbf{c} = \text{vec}(\mathbf{C}^T)$, as it alleviates the computations of the data terms,

hence J_C^{reg} consists of a k copies of $\text{diag}(w^{(t)})$ permuted to match this design choice. Given a current iterate $(\mathbf{b}^{(t)}, \mathbf{c}^{(t)})$ we write the regularization term as $\|J_B^{\text{reg}}\delta\mathbf{b} + \mathbf{r}_B\|^2 + \|J_C^{\text{reg}}\delta\mathbf{c} + \mathbf{r}_C\|^2$, where $\mathbf{r}_B = J_B^{\text{reg}}\mathbf{b}^{(t)}$, $\mathbf{r}_C = J_C^{\text{reg}}\mathbf{c}^{(t)}$, $\mathbf{b} = \mathbf{b}^{(t)} + \delta\mathbf{b}$ and $\mathbf{c} = \mathbf{c}^{(t)} + \delta\mathbf{c}$.

Linearizing the residuals $\mathcal{ABC}^T - \mathbf{b}$ around $(\mathbf{b}^{(t)}, \mathbf{c}^{(t)})$ gives an expression of the form

$$J_B^{\text{data}}\delta\mathbf{b} + J_C^{\text{data}}\delta\mathbf{c} + \mathbf{r}^{\text{data}}. \quad (69)$$

The particular shape of the Jacobians in this expression depends on the application; however, in all of our applications they are sparse. For example, in the missing data problem each residual corresponds to an element of the matrix \mathbf{X} which in turn only depends on k elements of \mathbf{B} and \mathbf{C} . Locally we may now write the objective function as

$$\|J_B\delta\mathbf{b} + J_C\delta\mathbf{c} + \mathbf{r}\|^2, \quad (70)$$

where

$$J_B = \begin{bmatrix} J_B^{\text{reg}} \\ 0 \\ J_B^{\text{data}} \end{bmatrix}, \quad J_C = \begin{bmatrix} 0 \\ J_C^{\text{reg}} \\ J_C^{\text{data}} \end{bmatrix}, \quad \mathbf{r} = \begin{bmatrix} \mathbf{r}_B \\ \mathbf{r}_C \\ \mathbf{r}^{\text{data}} \end{bmatrix}. \quad (71)$$

It was shown in [5] that each step of VarPro is equivalent to first minimizing (70) with the additional dampening term $\lambda\|\delta\mathbf{b}\|^2$ and then performing an exact optimization of (20) over the \mathbf{C} -variables (when fixing the \mathbf{B} -variables to their new values). Since we also have a reweighing we only do one iteration with VarPro before updating the weights $\mathbf{w}^{(t)}$.

The above procedure can return stationary points for which $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) > \mathcal{R}(\mathbf{BC}^T)$. Our last step is designed to escape such points by taking the current iterate and recompute the factorization of $\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ using SVD. If the SVD of $\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T = \sum_{i=1}^r \sigma_i \mathbf{U}_i \mathbf{V}_i^T$ we update $\tilde{\mathbf{B}}$ and $\tilde{\mathbf{C}}$ to $\tilde{\mathbf{B}}_i = \sqrt{\sigma_i} \mathbf{U}_i$ and $\tilde{\mathbf{C}}_i = \sqrt{\sigma_i} \mathbf{V}_i$ which we know reduces the energy and gives $\tilde{\mathcal{R}}(\tilde{\mathbf{B}}, \tilde{\mathbf{C}}) = \mathcal{R}(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T)$. Therefore we proceed by refactorizing the current iterate using SVD in each iteration. The detailed steps of the bilinear method are summarized in Algorithm 1.

C Additional Experiments on Real Data

pOSE: Psuedo Object Space Error

In this section we compare the energies over time for ADMM optimizing the same energy [6], *i.e.* with the regularizer \mathcal{R} , and $f = f_\mu$ as in (8) (of the main paper), and our proposed method. We let the bilinear method run until convergence, and let ADMM

Input: Robust penalty function f , linear operator \mathcal{A} and regularization parameter μ , damping parameter λ .
Initialize \mathbf{B} and \mathbf{C} with random entries
while *not converged* **do**
 Compute weights $\mathbf{w}^{(t)}$ from current iterate (\mathbf{B}, \mathbf{C})
 Compute the vectorizations $\mathbf{b} = \text{vec}(\mathbf{B})$, $\mathbf{c} = \text{vec}(\mathbf{C}^T)$
 Compute residuals \mathbf{r}_B , \mathbf{r}_C , and Jacobians J_B^{data} and J_C^{data} depending on \mathcal{A}
 Compute residual \mathbf{r}^{reg} , and Jacobians J_B^{reg} and J_C^{reg}
 Create full residual \mathbf{r} and Jacobians J_B and J_C
 Compute $\tilde{\mathbf{J}}^T \tilde{\mathbf{J}} + \lambda \mathbf{I} = J_B^T (\mathbf{I} - J_C J_C^+) J_B + \lambda \mathbf{I}$
 Compute $\mathbf{b}' = \mathbf{b} - (\tilde{\mathbf{J}}^T \tilde{\mathbf{J}} + \lambda \mathbf{I})^{-1} J_B^T \mathbf{r}$ and reshape into matrix \mathbf{B}'
 Compute \mathbf{C}' by minimizing (20) with fixed \mathbf{B}'
 if $\mathcal{R}(\mathbf{B}' \mathbf{C}'^T) + \|\mathcal{A}(\mathbf{B}' \mathbf{C}'^T) - \mathbf{b}\|^2 < \mathcal{R}(\mathbf{B} \mathbf{C}^T) + \|\mathcal{A}(\mathbf{B} \mathbf{C}^T) - \mathbf{b}\|^2$ **then**
 $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \text{svd}(\mathbf{B}' \mathbf{C}'^T)$
 Update $\mathbf{B} = \mathbf{U} \sqrt{\mathbf{\Sigma}}$ and $\mathbf{C} = \mathbf{V} \sqrt{\mathbf{\Sigma}}$
 Decrease λ
 else
 Increase λ
 end
end

Algorithm 1: Outline of the bilinear method.

execute the same time in seconds. As a comparison we use the nuclear norm relaxation and the discontinuous rank regularization. The results of the experiment are shown in Figure 6.

Again, note that the bilinear method optimizes the same energy as ADMM- \mathcal{R}_μ , and that, despite the initial fast lowering of the objective value, the ADMM approach fails to reach the global optimum, within the allotted 150 seconds. This holds true for all methods employing ADMM. In all experiments, the control parameter $\eta = 0.5$, and the μ parameter was chosen to be smaller than all non-zero singular values of the best known optimum (obtained using VarPro). For a fair comparison, the μ -value for the nuclear norm relaxation, was modified due to the shrinking bias, and was chosen to be the smallest value of μ for which a solution with accurate rank was obtained. Due to this modification, the energy it minimizes is not directly correlated to the others, but is shown for completeness. Furthermore, the iteration speed of ADMM is significantly faster than for VarPro, and therefore we show the elapsed time (in seconds) for all methods. The reported values are averaged over 50 instances with random initialization.

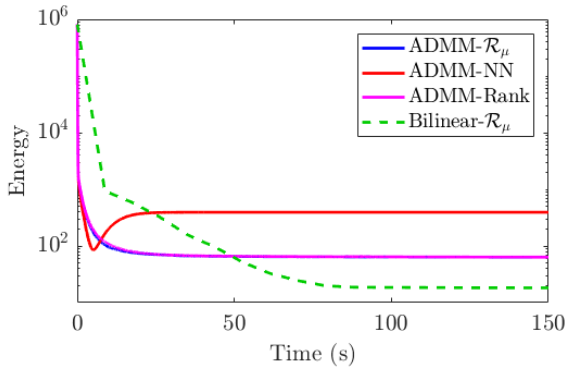


Figure 6: The average energy for the pOSE problem over 50 instances with random initializations, for test sequence *Door*. (Note that the energy for ADMM-Rank and ADMM- \mathcal{R}_μ are very similar).

Background Extraction

The missing data problem formulation can also be used in *e.g.* background extraction, where the goal is to separate the foreground from the background in a video sequence. For this experiment, security footage of an airport is used. The frame size is 144×176 pixels, and we use the first 200 frames, as in [3]. The camera does not move, hence the background is static.

By concatenating the vectorization of the frames into a matrix we expect it to be additively decomposable in terms of a low rank matrix (background) and a sparse matrix (foreground). We follow the setup used in [1], and crop the width to half of the height, and shift it 20 pixels to the right after 100 frames to simulate a virtual pan of the camera. This increases the complexity of the background, as it is no longer static. Lastly, we randomly drop 70 % of the entries. To allow for smaller singular values, we use Geman, as it is a robust penalty with shrinking bias. The results are shown in Figure 8.

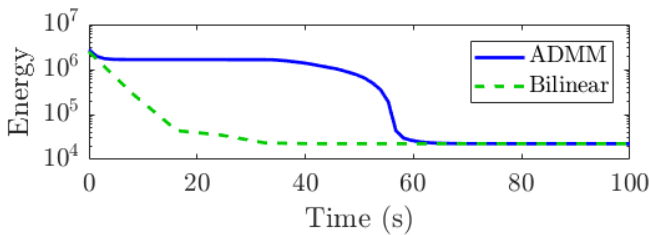


Figure 7: Energy minimization comparison for the background extraction experiment.



Figure 8: Background extraction using Geman. Samples from frame no. 40, 70, 100, 130, 170 and 200. *Top row*: Original images. *Middle row*: Training data with 70 % missing data. *Bottom row*: Reconstruction of background (bilinear method).

Initially ADMM struggles to find the correct balance between lowering the rank and fitting the data, which is seen in Figure 7, where the objective is almost unaffected the first forty seconds. At this point, the bilinear method has already converged.

Photometric Stereo

Photometric stereo can be used for estimating depth and surface orientation from images of the same object and view with varying lighting directions. Assuming M lighting directions and N pixels define $\mathbf{I} \in \mathbb{R}^{M \times N}$, where I_{ij} is the light intensity for lighting direction i and pixel j . Assuming Lambertian reflectance, uniform albedo and a distant light source, $\mathbf{I} = \mathbf{LN}$, where $\mathbf{L} \in \mathbb{R}^{M \times 3}$ contains the lighting directions and $\mathbf{N} \in \mathbb{R}^{3 \times N}$ the unknown surface normals. Thus, the resulting problem is to find a

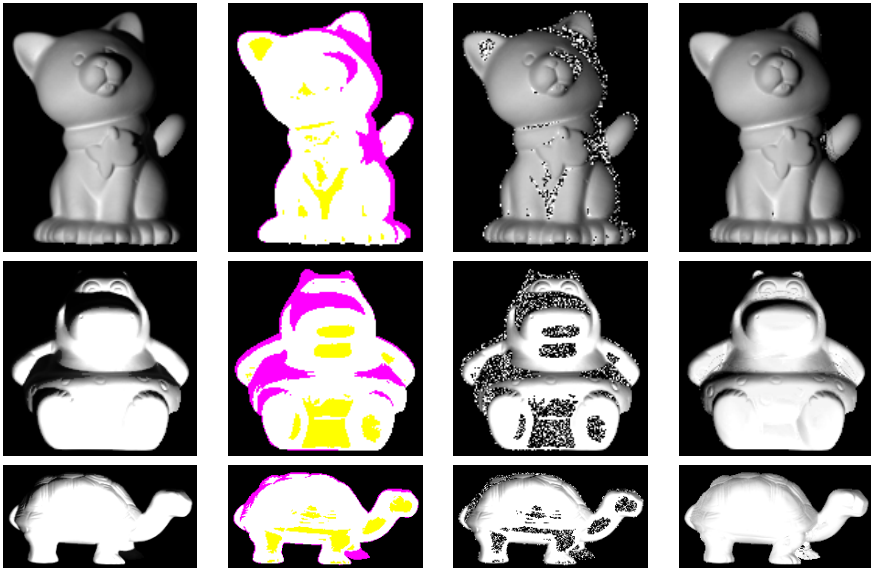


Figure 9: Images from the photometric stereo experiment. *From left to right:* (a) Ground truth image, (b) missing data mask with static background (black), dark pixels (purple), bright pixels (yellow), (c) reconstruction using ADMM, and (d) reconstruction using the Bilinear formulation.

rank 3 approximation of the intensity matrix I .

We use the Harvard Photometric Stereo testset [2], which contains images of various objects from varying lighting direction. The images are scaled to 160×125 pixels, and only the foreground pixels are used in the optimization. Similar to [1], we introduce missing data by thresholding dark pixels with pixel value less than 40 and bright pixels with pixel value more than 205. The measurement matrix is reconstructed using the bilinear method and the ADMM equivalent with the \mathcal{R}_μ regularization. The result is shown in Figure 9. We let the bilinear method run until convergence and let the ADMM equivalent run for the same time in seconds, at which point the objective value is still decreasing when ADMM is interrupted; however, the reduction is almost negligible. In all cases ADMM fails to converge to a low rank solution in the same time as the bilinear method, which yields a consistent result.

References

- [1] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [2] R. T. Frankot and R. Chellappa. “A Method for enforcing integrability in shape from shading algorithms”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10 (1988), pp. 439–451.
- [3] J. He, L. Balzano, and A. Szlam. “Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2012, pp. 1568–1575.
- [4] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.
- [5] J. H. Hong, C. Zach, A. Fitzgibbon, and R. Cipolla. “Projective Bundle Adjustment from Arbitrary Initialization Using the Variable Projection Method”. In: *European Conference on Computer Vision (ECCV)*. 2016, pp. 477–493.
- [6] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [7] A. S. Lewis. “The convex analysis of unitarily invariant matrix functions”. In: *Journal of Convex Analysis* 2.1 (1995), pp. 173–183.
- [8] C. Olsson, M. Carlsson, F. Andersson, and V. Larsson. “Non-Convex Rank/Sparsity Regularization and Local Minima”. In: *Proceedings of the International Conference on Computer Vision* (2017).

Differentiable Fixed-Rank Regularisation using Bilinear Parameterisation

MARCUS VALTONEN ÖRNHAG¹, CARL OLSSON^{1,2} AND ANDERS HEYDEN¹
¹*Centre for Mathematical Sciences, Lund University*
²*Department of Electrical Engineering, Chalmers University of Technology*

Abstract: Low rank structures are present in many applications of computer vision and machine learning. A popular approach consists of explicitly parameterising the set or matrices with sought rank, leading to a bilinear factorisation, reducing the problem to find the bilinear factors. While such an approach can be efficiently implemented using second-order methods, such as Levenberg–Marquardt (LM) or Variable Projection (VarPro), it suffers from the presence of local minima, which makes theoretical optimality guarantees hard to derive.

Another approach is to penalise non-zero singular values to enforce a low-rank structure. In certain cases, global optimality guarantees are known; however, such methods often lead to non-differentiable (and even discontinuous) objectives, for which it is necessary to use subgradient methods and splitting schemes. If the objective is complex, such as in structure from motion, the convergence rates for such methods can be very slow.

In this paper we show how optimality guarantees can be lifted to methods that employ bilinear parameterisation when the sought rank is known. Using this approach the best of two worlds are combined: optimality guarantees and superior convergence speeds. We compare the proposed method to state-of-the-art solvers for prior-free non-rigid structure from motion.

1 Introduction

The singular value decomposition (SVD) has long been the main tool for enforcing rank constraint. It is well-known that when all elements are measured the optimal solution is obtained by finding the SVD of a matrix, thresholding the first k singular values, and recombining the remaining entries to obtain the optimal rank k approximation (measured in the Frobenius norm). In computer vision and machine learning applications, however, missing data patterns emerge, often in structured ways, and such simple methods are not directly applicable.

In this paper we will consider low rank approximation problems with structured missing data problems, which *e.g.* emerge in structure from motion problems. More

specifically, we are interested in solving

$$\min_{\text{rank}(\mathbf{X}) \leq r_0} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2. \quad (1)$$

where $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ is a linear operator, $\mathbf{b} \in \mathbb{R}^p$ and $\|\cdot\|$ is the standard Euclidean norm. We may, equivalently, consider the unconstrained problem formulation

$$\min_{\mathbf{X}} \mathbb{I}(\text{rank}(\mathbf{X}) \leq r_0) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2. \quad (2)$$

where $\mathbb{I}(\text{rank}(\mathbf{X}) \leq r_0)$ is the indicator function attaining the value 0 if $\text{rank}(\mathbf{X}) \leq r_0$ and ∞ otherwise. Due to being discontinuous, the indicator function is not suitable for numerics. The standard approach has been to relax it with a convex alternative, such as the nuclear norm $\|\mathbf{X}\|_* = \sum_{i=1}^n \sigma_i(\mathbf{X})$, where $\sigma_i(\mathbf{X})$ is the i :th singular value of \mathbf{X} . This relaxation has the theoretical benefit of being the convex envelope of the rank function over the set $\{\mathbf{X} : \sigma_1(\mathbf{X}) \leq 1\}$, see *e.g.* [10], which further has led to generalisations with performance guarantees [5, 6, 22, 24]. Due to the fact that the singular values are penalised equally hard, regardless of size, the nuclear norm has a shrinking bias [20]. For computer vision problems this is often an undesirable effect, which has been shown on structure-from-motion problems [8, 19]. Instead, non-convex relaxations that penalise smaller singular values harder than larger ones have been shown to improve performance [14, 16, 17, 21]. One of the downsides of using these types of regularisation terms is that they often result in non-differentiable objectives, which makes it necessary to use splitting schemes or subgradient based methods. This in turn may affect the convergence rates, making it infeasible for certain types of problems.

Since the sought rank is known, we may instead consider optimising over a bilinear factorisation $\mathbf{X} = \mathbf{B}\mathbf{C}^T$, where $\mathbf{B} \in \mathbb{R}^{m \times r}$ and $\mathbf{C} \in \mathbb{R}^{n \times r}$, which explicitly parameterises the family of matrices of size $m \times n$ with rank at most r . While methods utilising a bilinear parameterisation can be extended to second order methods, with fast convergence rates in the neighbourhood of local minima, they instead suffer from the presence of local minima. Therefore, attempts to unify regularisation terms with bilinear factorisation has gained some attention in recent years [2, 4, 20, 25, 27], with the hope of combining the best from both worlds: theoretical performance guarantees and fast convergence.

2 Related Work

Cabral *et al.* [4] considered the variational formulation of the nuclear norm

$$\|\mathbf{X}\|_* = \min_{\mathbf{B}\mathbf{C}^T = \mathbf{X}} \frac{1}{2} (\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2), \quad (3)$$

see [24], and unified the use of a regularised objective and bilinear factorisation. They are able to prove global optimality in cases where the obtained solution has lower rank than the number of columns. Bach [2] extended this property to regularisers on the form $\|\mathbf{X}\|_{s,t} = \min_{\mathbf{BC}^T=\mathbf{X}} \frac{1}{2} \sum_{i=1}^k (\|\mathbf{B}_i\|_s^2 + \|\mathbf{C}_i\|_t^2)$, where $\|\cdot\|_p$ is the ℓ_p -norm. These methods, however, rely on convexity, thus implicitly suffers from shrinking bias. Shang *et al.* [25] went beyond convexity and studied the Schatten semi-norms $\|\mathbf{X}\|_q = \sqrt[q]{\sum_{i=1}^N \sigma_i(\mathbf{X})^q}$, for $q = 1/2$ and $q = 2/3$ (which was generalised in [27]). The shrinking bias was drastically reduced, but the proposed method does not benefit from the convergence rates of second-order methods, as the non-convexity is treated using a splitting scheme.

Recently, Valtonen Örnhag *et al.* [20] studied a family of regularisers on the form $\mathcal{R}(\mathbf{X}) = \sum_{i=1}^k f(\sigma_i(\mathbf{X}))$, where f is a robust penalty function, assumed to be concave and nondecreasing on $[0, \infty)$ with $f(0) = 0$. They showed that objectives incorporating such regularisers, which are non-differentiable by nature, can be reformulated into differentiable objectives using bilinear factorisation. Furthermore, they can be optimised using second-order methods such as Levenberg–Marquardt (LM) and Variable Projection (VarPro). Theoretical optimality guarantees for the choice $f(x) = f_\mu(x) := \mu - \max(\sqrt{\mu} - x, 0)$ which is equivalent to the the convex envelope of the “soft rank” objective $\mu \text{rank}(\mathbf{X}) + \|\mathbf{X} - \mathbf{M}\|_F^2$ were studied. Previously known results, including global optimality guarantees, were transferred to the bilinear setting.

In this paper we will focus on overparameterising bilinear formulations for the “hard rank” objective (2). In [15] a special case was studied

$$\min_{\mathbf{X}} \mathbb{I}(\text{rank}(\mathbf{X}) \leq r_0) + \|\mathbf{X} - \mathbf{M}\|_F^2, \tag{4}$$

and it was shown that the convex envelope of the objective function is given by

$$\mathcal{R}_{r_0}(\mathbf{X}) + \|\mathbf{X} - \mathbf{M}\|_F^2, \tag{5}$$

where

$$\mathcal{R}_{r_0}(\mathbf{X}) = \max_{\mathbf{Z}} \sum_{i=r_0+1}^n \sigma_i(\mathbf{Z})^2 - \|\mathbf{X} - \mathbf{Z}\|_F^2, \tag{6}$$

Since (5) is the convex envelope of (4) the global minimisers are attained simultaneously. Furthermore, (5) is continuous and convex, which makes it tractable for practical problems. Inspired by the approach used in Valtonen Örnhag *et al.* [20] we show that it is possible to combine theoretical optimality guarantees and second order methods for the hard rank objective as well. Our contributions are:

- A novel method for regularising fixed-rank problems,

- Optimality guarantees for a wide range of problems,
- Comparison to state-of-the-art methods on Non-Rigid Structure from Motion.

3 Differentiable Regularisers

In this section we will derive an alternative regulariser to (6) with certain desired properties suitable in a framework utilising bilinear parameterisation. One such property is differentiability.

3.1 Bilinear Parameterisation and Pseudo-Singular Values

Let $\mathbf{X} \in \mathbb{R}^{m \times n}$, with $\text{rank}(\mathbf{X}) = r_0$. Then there exists a decomposition $\mathbf{B} \in \mathbb{R}^{n \times k}$ and $\mathbf{C} \in \mathbb{R}^{m \times k}$, with $r_0 \leq k$, such that $\mathbf{X} = \mathbf{BC}^T$. Furthermore, define the *pseudo-singular values*

$$\gamma_i(\mathbf{B}, \mathbf{C}) := \frac{\|\mathbf{B}_{[i]}\|^2 + \|\mathbf{C}_{[i]}\|^2}{2}, \quad (7)$$

where the square brackets indicate the columns of \mathbf{B} and \mathbf{C} such that the pseudo-singular values are sorted in descending order $\gamma_1(\mathbf{B}, \mathbf{C}) \geq \dots \geq \gamma_k(\mathbf{B}, \mathbf{C}) \geq 0$. Note that if $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is a SVD of \mathbf{X} , then the re-factorisation $\mathbf{B} = \mathbf{U}\sqrt{\mathbf{\Sigma}}$ and $\mathbf{C} = \mathbf{V}\sqrt{\mathbf{\Sigma}}$, such that $\mathbf{X} = \mathbf{BC}^T$, has the properties that the singular values and the pseudo-singular values coincide $\gamma_i(\mathbf{B}, \mathbf{C}) = \sigma_i(\mathbf{X})$, for all $i = 1, \dots, k$, if we use the convention that $\sigma_i(\mathbf{X}) = 0$ for $i > r_0$.

3.2 A Bilinear Fixed-Rank Regulariser

Let $\sigma(\mathbf{X})$ denote the singular value vector of \mathbf{X} , and note that the first term of (6) is unitarily invariant, whereas the second term can be expressed as $-\|\mathbf{X} - \mathbf{Z}\|_F^2 = 2\langle \mathbf{X}, \mathbf{Z} \rangle_F - \|\mathbf{X}\|_F^2 - \|\mathbf{Z}\|_F^2$. Recall that, by von Neumann's trace theorem, $|\langle \mathbf{X}, \mathbf{Z} \rangle_F| \leq \langle \sigma(\mathbf{X}), \sigma(\mathbf{Z}) \rangle$, with equality when \mathbf{X} and \mathbf{Z} are simultaneously unitarily diagonalisable. This reduces the problem to maximising over the singular values alone,

$$\mathcal{R}_{r_0}(\mathbf{X}) = \max_{\sigma(\mathbf{Z})} \left(\sum_{i=r_0+1}^n \sigma_i^2(\mathbf{Z}) - \sum_{i=1}^n (\sigma_i(\mathbf{Z}) - \sigma_i(\mathbf{X}))^2 \right). \quad (8)$$

A bilinear formulation equivalent to (8) can now be created by replacing the singular values by the pseudo-singular values (7), which yields

$$\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) := \max_{\mathbf{z} \in \mathcal{Z}} \left(\sum_{i=r_0+1}^n z_i^2 - \sum_{i=1}^n (z_i - \gamma_i(\mathbf{B}, \mathbf{C}))^2 \right), \quad (9)$$

where $\mathcal{Z} = \{\mathbf{z} : z_1 \geq \dots \geq z_n \geq 0\}$. Clearly, $\mathcal{R}_{r_0}(\mathbf{X}) = 0$ if and only if $\text{rank}(\mathbf{X}) \leq r_0$; hence, the proposed regulariser will penalise solutions admitting more than r_0 non-zero columns in the bilinear factorisation. By recomputing the bilinear factors such that $\mathbf{B} = \mathbf{U}\sqrt{\Sigma}$ and $\mathbf{C} = \mathbf{V}\sqrt{\Sigma}$, where $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$ is a SVD, yields $\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) = \mathcal{R}_{r_0}(\mathbf{BC}^T)$, thus, implicitly, enforces the rank constraint.

3.3 Differentiability

We next show that the cost function, including the bilinear regulariser (9), is differentiable.

Theorem 1. *The function $\mathcal{F} : \mathbb{R}^{m \times k} \times \mathbb{R}^{n \times k} \rightarrow \mathbb{R}$, defined as*

$$\mathcal{F}(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}(\mathbf{BC}^T) - \mathbf{b}\|^2, \quad (10)$$

is differentiable w.r.t. \mathbf{B} and \mathbf{C} .

Proof sketch. For a complete proof, see the supplementary material. Decompose $\mathcal{F}(\mathbf{B}, \mathbf{C}) = G(\mathbf{B}, \mathbf{C}) + H(\mathbf{B}, \mathbf{C})$ where $G(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) + \sum_{i=1}^k \gamma_i^2(\mathbf{B}, \mathbf{C})$ and $H(\mathbf{B}, \mathbf{C}) = -\sum_{i=1}^k \gamma_i^2(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}(\mathbf{BC}^T) - \mathbf{b}\|^2$. Clearly, H is differentiable, and we will show that G is convex and differentiable. Let $\gamma : \mathbb{R}^{m \times k} \times \mathbb{R}^{n \times k} \rightarrow \mathbb{R}^k$ denote the function that takes the bilinear factors and returns the pseudo-singular values,

$$\gamma(\mathbf{B}, \mathbf{C}) = \frac{1}{2} \left(\|\mathbf{B}_{[1]}\|^2 + \|\mathbf{C}_{[1]}\|^2, \dots, \|\mathbf{B}_{[k]}\|^2 + \|\mathbf{C}_{[k]}\|^2 \right). \quad (11)$$

and

$$\varphi(\gamma) = \max_{z_1 \geq z_2 \geq \dots \geq z_k \geq 0} L(\gamma, \mathbf{z})^T, \quad (12)$$

where $L(\gamma, \mathbf{z}) = -\sum_{i=1}^{r_0} z_i^2 + 2\sum_{i=1}^k z_i \gamma_i$. Note that $G(\mathbf{B}, \mathbf{C}) = \varphi(\gamma(\mathbf{B}, \mathbf{C}))$ and by the chain rule $\partial_{\mathbf{B}} G(\mathbf{B}, \mathbf{C}) = \nabla_{\mathbf{B}} \gamma(\mathbf{B}, \mathbf{C}) \partial_{\gamma} \varphi(\gamma(\mathbf{B}, \mathbf{C}))$. Here $\nabla_{\mathbf{B}} \gamma(\mathbf{B}, \mathbf{C}) = \text{blkdiag}(\mathbf{B}_{[1]}, \mathbf{B}_{[2]}, \dots, \mathbf{B}_{[k]})$, $\partial_{\gamma} \varphi = \left(\partial_{\gamma_{[1]}} \varphi(\gamma), \partial_{\gamma_{[2]}} \varphi(\gamma), \dots, \partial_{\gamma_{[k]}} \varphi(\gamma) \right)^T$ and

therefore $\partial_{\mathbf{B}}G(\mathbf{B}, \mathbf{C})$ is an $mk \times 1$ vector containing derivatives with respect to the elements of \mathbf{B} . Using a similar argument as in [15], the subdifferential can be written as

$$\partial\varphi(\gamma) = 2 \operatorname{argmax}_{z_1 \geq \dots \geq z_k \geq 0} \{L(\gamma, \mathbf{z})\}. \quad (13)$$

Analogously to [15], the optimising vector \mathbf{z} is given by

$$z_i \in \begin{cases} \{\max(\gamma_i, s)\}, & i \leq r_0 \\ \{s\}, & i \geq r_0, \gamma_i \neq 0 \\ [0, s], & i > r_0, \gamma_i = 0 \end{cases} \quad (14)$$

for some $s \geq \gamma_{r_0}$. For G to be differentiable with respect to \mathbf{B} it is sufficient that $\partial_{\mathbf{B}}G(\mathbf{B}, \mathbf{C})$ contains a single element. To see that this is true one notes that only the elements of the subgradient $\partial\varphi(\gamma)$, for which $\gamma_i = 0$ and $i > r_0$, can have non-singleton sets; however, $\gamma_i = 0$ implies that both $\mathbf{B}_{[i]}$ and $\mathbf{C}_{[i]}$ are zero vectors. Therefore all elements $\partial_{\gamma_{[i]}}\varphi(\gamma)$ that can take multiple values vanish in the multiplication $\nabla_{\mathbf{B}}\gamma(\mathbf{B}, \mathbf{C})\partial_{\gamma}\varphi(\gamma(\mathbf{B}, \mathbf{C}))$. In conclusion, G is differentiable with respect to \mathbf{B} , thus also \mathcal{F} . An identical argument now shows that the same is true for derivatives with respect to \mathbf{C} . \square

4 Optimality Conditions

In this section, we show that previously known results concerning the regulariser (6) can be transferred to the bilinear setting. This is done by first establishing a crucial relation between $\mathcal{R}_{r_0}(\mathbf{X})$ and the proposed regulariser $\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C})$. This is a generalisation of [20], and we use the same strategy, namely, to overparameterise the bilinear factorisation such that \mathbf{B} and \mathbf{C} have $2k$ columns. This, in turn, allows us to parameterise line segments between points of at most rank k , in addition to applying convexity properties.

It should be noted that overparameterisation introduces additional stationary points. Considering the data term $\|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2$, it is clear that the gradients w.r.t. the bilinear factors vanish at $(\mathbf{B}, \mathbf{C}) = (0, 0)$. This is not the case for gradients w.r.t. $\mathbf{X} = \mathbf{B}\mathbf{C}^T$ which are non-zero, in general. Nevertheless, it is possible to relate the local minima of the problem formulation (5) and the directional derivatives between low rank factorisations of the proposed bilinear formulation, which is done in Theorem 2.

Theorem 2. *Assume that $(\bar{\mathbf{B}}, \bar{\mathbf{C}}) \in \mathbb{R}^{m \times 2k} \times \mathbb{R}^{n \times 2k}$ is a local minimizer of (10), where $\bar{\mathbf{B}} = \mathbf{U}\sqrt{\Sigma}$ and $\bar{\mathbf{C}} = \mathbf{V}\sqrt{\Sigma}$, and $\bar{\mathbf{X}} = \mathbf{U}\Sigma\mathbf{V}^T$, with $r_0 < k$ non-zero columns and let $\mathcal{N}(\mathbf{X}) = \mathcal{R}_{r_0}(\mathbf{X}) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2$. If $\mathcal{R}_{r_0}(\tilde{\mathbf{X}}) = \tilde{\mathcal{R}}_{r_0}(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ then the directional derivatives $\mathcal{N}'_{\Delta\mathbf{X}}(\tilde{\mathbf{X}})$, where $\Delta\mathbf{X} = \tilde{\mathbf{X}} - \bar{\mathbf{X}}$, $\operatorname{rank}(\tilde{\mathbf{X}}) \leq k$ are non-negative.*

The assumptions that the local minimizers must be on the form $\bar{\mathbf{B}} = \mathbf{U}\sqrt{\bar{\boldsymbol{\Sigma}}}$ and $\bar{\mathbf{C}} = \mathbf{V}\sqrt{\bar{\boldsymbol{\Sigma}}}$, respectively, is not restricting the applicability, since there can be local minimizers for which $\tilde{\mathcal{R}}_{r_0}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) \neq \mathcal{R}_{r_0}(\bar{\mathbf{B}}\bar{\mathbf{C}}^T)$. In order to avoid such situations, we recompute the SVD each iteration, according to $\bar{\mathbf{B}} = \mathbf{U}\sqrt{\bar{\boldsymbol{\Sigma}}}$ and $\bar{\mathbf{C}} = \mathbf{V}\sqrt{\bar{\boldsymbol{\Sigma}}}$, assuming $\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ is an SVD of $\bar{\mathbf{B}}\bar{\mathbf{C}}^T$. By doing so, we enforce $\tilde{\mathcal{R}}_{r_0}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathcal{R}_{r_0}(\bar{\mathbf{B}}\bar{\mathbf{C}}^T)$.

Using the result from Theorem 2, we can apply the non-negativity argument of the directional derivatives, to problems obeying the *restricted isometry property* (RIP) [24]

$$(1 - \delta_{2k})\|\mathbf{X}\|_F^2 \leq \|\mathcal{A}(\mathbf{X})\|^2 \leq (1 + \delta_{2k})\|\mathbf{X}\|_F^2, \quad (15)$$

for some $0 < \delta_{2k} < 1$. The theory of local minimizers in the bilinear factors of $\|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2$ under the RIP constraint has recently been developed in [3, 11, 23]. This is a well-studied class of problems, for which it can be shown that no spurious local minima are introduced using the bilinear factorisation [23] (under standard regulatory assumptions on \mathcal{A}). The fundamental idea is to bound the distance between the global minimum and the local minima, which can be made small in terms of the residual error; however, this approach does not, in general, guarantee uniqueness of local minima. By lifting the results of [18] to the bilinear setting, we can in fact prove uniqueness under the RIP constraint. Theorem 3 shows how this can be incorporated to ensure global minimality.

Theorem 3. *Assume that $(\bar{\mathbf{B}}, \bar{\mathbf{C}})$ is a local minimizer of (10) fulfilling the assumptions of Theorem 2. If the singular values of $\mathbf{Z} = (\mathbf{I} - \mathcal{A}^*\mathcal{A})\bar{\mathbf{B}}\bar{\mathbf{C}}^T + \mathcal{A}^*\mathbf{b}$, where \mathcal{A}^* is the adjoint operator of \mathcal{A} , fulfil $\sigma_{r_0+1}(\mathbf{Z}) < (1 - 2\delta_{2r_0})\sigma_{r_0}(\mathbf{Z})$, then*

$$\bar{\mathbf{B}}\bar{\mathbf{C}}^T \in \arg \min_{\text{rank}(\mathbf{X}) \leq r_0} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2. \quad (16)$$

In most practical situations, the separation of the singular values of \mathbf{Z} is easily fulfilled. Consider, *e.g.* the case of exact data, when $\mathbf{b} = \mathcal{A}(\mathbf{X}_0)$, where $\text{rank}(\mathbf{X}_0) = r_0$. Then $\mathbf{Z} = \mathbf{X}_0$, and the condition is transferred to the properties of the local (global) minima \mathbf{X}_0 , *i.e.* $\sigma_{r_0+1}(\mathbf{X}_0) < (1 - 2\delta_{2r_0})\sigma_{r_0}(\mathbf{X}_0)$. The proof is given in the supplementary material.

5 Implementation

The method we propose is based on VarPro [12], which allows us to work with the low-rank factors \mathbf{B} and \mathbf{C} directly. Assuming a separable non-linear least squares problem, in the components $\mathbf{b} = \text{vec}(\mathbf{B})$ and $\mathbf{c} = \text{vec}(\mathbf{C}^T)$, VarPro reduces the problem by solving

a nonlinear problem in \mathbf{b} only. This is done by marginalising \mathbf{c} , prior to optimising over \mathbf{b} . For a thorough overview of VarPro for computer vision applications, see [13].

Many computer vision and machine learning problems are large, but highly structured. Failure to retain any previous structure by adding a regulariser may result in a non-feasible optimisation scheme, due to the computational cost of estimating the (modified) Jacobians. In order to make minimal impact on existing structures, we therefore proceed to linearise the proposed regulariser. This is done by using the alternative formulation in [1], which yields

$$\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) = \frac{1}{r_0 - \ell} \left(\sum_{i>\ell} \gamma_i(\mathbf{B}, \mathbf{C}) \right)^2 - \sum_{i>\ell} \gamma_i^2(\mathbf{B}, \mathbf{C}), \quad (17)$$

where ℓ is the smallest non-negative integer fulfilling $\gamma_\ell \geq \frac{1}{r_0 - \ell} \sum_{i>\ell} \gamma_i(\mathbf{B}, \mathbf{C}) \geq \gamma_{\ell+1}$. After expanding the parenthesis, we arrive at

$$\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) = \frac{1}{r_0 - \ell} \sum_{i>\ell} \sum_{\substack{j>\ell \\ i \neq j}} \gamma_i \gamma_j - \left(1 - \frac{1}{r_0 - \ell} \right) \sum_{i>\ell} \gamma_i^2. \quad (18)$$

For each $i > \ell$ let $\alpha_i = \sum_{\substack{j>\ell \\ i \neq j}} \gamma_j$, then

$$\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) = \sum_{i>\ell} f_i(\gamma_i(\mathbf{B}, \mathbf{C})), \quad (19)$$

where $f_i(x) = \kappa \alpha_i x - (1 - \kappa)x^2$ with $\kappa = 1/(r_0 - \ell)$. By considering the first order Taylor expansion $f_i(x) \approx f_i(x_0) + f'_i(x_0)(x - x_0)$ termwise (discarding constant terms) about x_0 , we get the approximation

$$\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) \approx \sum_{i>\ell} w_i^{(t)} \left(\|\mathbf{B}_i^{(t)}\|^2 + \|\mathbf{C}_i^{(t)}\|^2 \right), \quad (20)$$

where $\mathbf{B}^{(t)}$ and $\mathbf{C}^{(t)}$ are the current iterates, and

$$w_i^{(t)} = \begin{cases} 0, & i \leq \ell \\ \frac{1}{2} f'_i \left(\frac{\|\mathbf{B}_i^{(t)}\|^2 + \|\mathbf{C}_i^{(t)}\|^2}{2} \right) & i > \ell. \end{cases} \quad (21)$$

When $\text{rank}(\mathbf{BC}^T) \leq r_0$, we use the right limit of the weights, *i.e.* $w_i^{(t)} = 0$ for $i \leq r_0$ and $\frac{1}{2} f'_i \left((\|\mathbf{B}_i^{(t)}\|^2 + \|\mathbf{C}_i^{(t)}\|^2)/2 \right)$ otherwise. We give a more detailed exposition of the algorithm in the supplementary material.

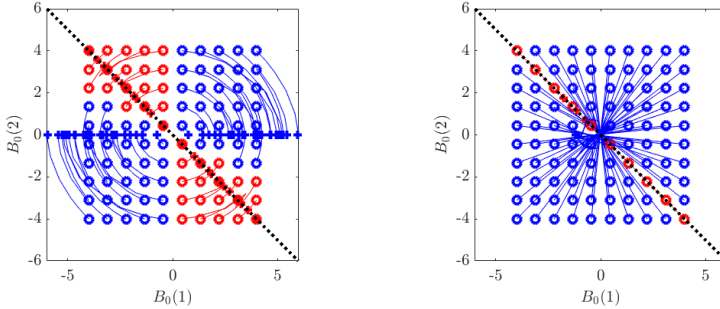


Figure 1: Convergence for different initial values of the first bilinear factor B_0 , for regular VarPro (left) and using overparameterisation (right). Circles indicate the initial positions, whereas crosses indicate the final positions. The lines connecting the markers indicate the intermediate updates. The path is coloured blue if the algorithm converged to the correct global minimum for the corresponding initial point, otherwise it is coloured red. The dotted line indicate the fixed points, discussed in Section 6.

6 Why Overparameterisation?

Consider the original problem of solving (2). One could, with good reason, ask whether it suffices to use VarPro with a bilinear parameterisation admitting a solution of rank at most r_0 , *i.e.* with $k = r_0$ columns. Why bother overparameterising with $k > r_0$ columns and regularise the excess columns? In this section we will show, by a toy example, that constraining the solvers to seek solutions in a low rank manifold may converge to incorrect solutions for a significant part of initial points, whereas, by using overparameterisation, convergence to false minima is significantly reduced.

Consider the problem of finding a 2×2 matrix of rank 1, minimising $\|\mathcal{A}(X) - \mathbf{b}\|^2$, where $\mathcal{A}(X) = A \text{vec } X$, with

$$A = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}. \quad (22)$$

The problem is constructed such that there is a unique solution to $\mathcal{A}(X) = \mathbf{b}$, given by

$$X^* = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \quad (23)$$

which is also a rank 1 matrix. Therefore, the global minimum is 0, with the unique minimizer X^* . Note, however, that for regular VarPro, any bilinear factorisation on the

form

$$\bar{\mathbf{B}} = \frac{\alpha}{\sqrt{2}} [1 \quad -1]^T \quad \text{and} \quad \bar{\mathbf{C}} = \frac{1}{\sqrt{2\alpha}} [-1 \quad 1]^T, \quad (24)$$

where $\alpha \neq 0$, gives a fixed point corresponding to $\bar{\mathbf{X}} = \bar{\mathbf{B}}\bar{\mathbf{C}}^T \neq \mathbf{X}^*$. Indeed, the Jacobians J_b and J_c are non-zero, however $J_b^T \boldsymbol{\varepsilon} = 0$ and $J_c^T \boldsymbol{\varepsilon} = 0$, where $\boldsymbol{\varepsilon}$ is the residual vector. In fact, VarPro converges locally to points on this line, which is shown in Figure 1, where we illustrate the convergence from different initial points. In the example we use $\mathbf{C}_0 = [1 \quad 1]^T$ and fix $\lambda = 1$, but other values result in a similar overall trend. The initial points marked red converge to a point on the line (24), which is a significant part of the initial values. Avoiding such false minima can be alleviated by overparameterisation. Consider using $k = 2$ columns. Starting at the same initial point, by adding a zero column to both \mathbf{B}_0 and \mathbf{C}_0 , it turns out that (24) is not necessarily a fixed point, as $J_b^T \boldsymbol{\varepsilon}$ and $J_c^T \boldsymbol{\varepsilon}$ are non-zero, in general. This is further supported by Figure 1, where it is readily seen that only initial points on the line (24) converges to a non-global minimizer.

7 Non-Rigid Structure from Motion

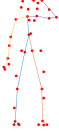

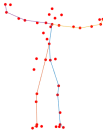
To show the benefits of using the proposed method, we compare it to state-of-the-art methods compatible with (1) on the CMU Motion Capture (MOCAP) dataset. We compare it to the method proposed by [15] which utilises ADMM and the proximal operator of \mathcal{R}_{r_0} . We include two relaxations for the “soft-rank” penalty: APGL [26] using nuclear norm, and IRNN [7] which is iteratively reweighted with the robust penalty function MCP [28]¹. Lastly, the standard VarPro [12] method is used in the comparison.

We shall treat this as a low-rank factorisation problem, by employing the approach proposed by Dai *et al.* [9]. Define the matrices

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{Y}_1 \\ \mathbf{Z}_1 \\ \vdots \\ \mathbf{X}_F \\ \mathbf{Y}_F \\ \mathbf{Z}_F \end{bmatrix} \quad \text{and} \quad \mathbf{X}^\sharp = \begin{bmatrix} \mathbf{X}_1 & \mathbf{Y}_1 & \mathbf{Z}_1 \\ \vdots & \vdots & \vdots \\ \mathbf{X}_F & \mathbf{Y}_F & \mathbf{Z}_F \end{bmatrix}, \quad (25)$$

¹This was chosen since $f(x) = f_\mu(x) := \mu - \max(\sqrt{\mu} - x, 0)$ (the convex envelope of the “soft rank” objective) is a special case of MCP.

Table 1: Rank vs datafit for the MOCAP experiment. For the bilinear method $k = 2r_0$ columns were used, and was set to a maximum of 80 iterations. Unregularised VarPro and ADMM- \mathcal{R}_{r_0} were allowed to run for the same time (in seconds), or until a local minimum was reached.

	r_0	APGL* [26]	IRNN* [7]	\mathcal{R}_{r_0} [15]	VarPro [12]	Our
<i>Drink</i> 	2	26.4174	17.0260	17.0586	17.0260	17.0260
	3	21.3455	11.0075	12.4201	10.9480	10.9478
	4	14.1693	5.9002	6.7927	5.6855	5.6855
	5	10.3880	4.5602	5.0217	4.4170	4.3143
	6	5.9997	4.3858	3.9884	3.5106	3.4804
	7	5.7065	3.4676	3.2135	2.9517	2.9048
	8	5.4290	2.9927	2.7530	2.6283	2.4146
	<i>Pickup</i> 	2	46.4320	25.3353	25.4007	25.3351
3		14.8115	8.7234	8.9140	9.3662	8.7234
4		13.7479	6.6112	6.9091	6.5398	6.4909
5		11.1644	4.8863	5.4458	4.9002	5.0270
6		9.5866	3.5880	4.6723	3.5187	3.4900
7		6.3250	3.3919	3.5680	2.7745	2.7342
8		5.8710	2.4373	2.8725	2.1132	2.2077
<i>Stretch</i> 		2	31.6038	21.8045	21.8224	21.8045
	3	18.4486	10.6094	10.6484	11.7657	10.6094
	4	14.8941	7.4601	7.4022	7.5132	7.2913
	5	11.4202	6.9302	6.1334	5.8904	5.8798
	6	9.4485	4.9070	4.9382	4.6281	4.6626
	7	8.2575	4.6458	4.1228	3.6614	3.6249
	8	6.8711	2.9488	3.1748	2.7256	2.7044

(*) APGL and IRNN use the “soft rank” constraint, for which a regularisation parameter must be set. For this experiment, we selected a large range of values and reported the *best values* (not the mean as for the other methods).

where \mathbf{X}_i , \mathbf{Y}_i and \mathbf{Z}_i contain the x -, y - and z -coordinates of the tracked points of the i :th image. Assuming K basis shapes, the matrix \mathbf{X}^\sharp may be decomposed into low-rank factors $\mathbf{X}^\sharp = \mathbf{C}\mathbf{B}^\sharp$, where $\mathbf{C} \in \mathbb{R}^{F \times K}$ are the shape coefficients and $\mathbf{B}^\sharp \in \mathbb{R}^{K \times 3n}$ contains the basis elements. The reason for working with the reshuffled matrices \mathbf{X}^\sharp and \mathbf{B}^\sharp , respectively, is to be able to enforce a stronger rank penalty.

Assuming orthographic cameras, the projection of the scene points are given by $\mathbf{M}_i = \mathbf{R}_i \mathbf{X}_i$, where $\mathbf{R}_i \in \mathbb{R}^{2 \times 3}$, with $\mathbf{R}_i \mathbf{R}_i^T = \mathbf{I}_2$. A suitable objective function [19] is thus given by

$$\min_{\text{rank}(\mathbf{X}^\sharp) \leq K} \|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F^2 + \|\mathbf{D}\mathbf{X}^\sharp\|_F^2, \quad (26)$$

where $\mathbf{R} \in \mathbb{R}^{2F \times 3F}$ is a block-diagonal matrix with the camera matrices \mathbf{R}_i on the main diagonal, and $\mathbf{M} \in \mathbb{R}^{2F \times n}$ contain the image points. It is known that only considering the datafit as an object is not ideal for NRSfM [9, 19], and, in some cases, promotes non-physical high-rank solutions, and, further penalising the derivative of the 3D projections have been suggested to increase performance [9]. Therefore, the term containing a (modified) difference operator $\mathbf{D} : \mathbb{R}^F \rightarrow \mathbb{R}^{\lfloor F/2 \rfloor}$, where $\lfloor \cdot \rfloor$ is the floor operator, is included to promote realistic reconstructions. The difference operator is modified to be block-diagonal, hence does not affect the structure of the Jacobians. The results can be seen in Table 1.

Note that we only report the best results for APGL and IRNN as they require the corresponding regularisation parameter to be correctly set. We, therefore, run several tests with varying regularisation strengths and pick the best result for each rank level. This is to minimise potential shrinking bias. In all but three cases the proposed method produced the lowest mean value.

8 Conclusions

In this paper we have presented a novel unification of bilinear parameterisation and rank regularisation utilising overparameterisation to achieve new theoretical optimality guarantees. These results were previously only known in the context of rank penalisation objectives, for which second-order methods are not feasible due to non-differentiability. Using our proposed algorithm we are able to lift essential parts of the theoretical framework developed for regularisation methods, while retaining a differentiable objective suitable for second-order methods. Among the theoretical contributions, we show new strong optimality results under the RIP constraint.

We proposed an algorithm based on VarPro, and show increased performance for difficult objectives for estimating the human pose in a Non-Rigid Structure from Motion framework, compared to state-of-the-art methods.

References

- [1] F. Andersson, M. Carlsson, and C. Olsson. “Convex envelopes for fixed rank approximation”. In: *Optimization Letters* (2017), pp. 1–13.
- [2] F. R. Bach. “Convex relaxations of structured matrix factorizations”. In: *CoRR* abs/1309.3117 (2013).

-
- [3] S. Bhojanapalli, B. Neyshabur, and N. Srebro. “Global Optimality of Local Search for Low Rank Matrix Recovery”. In: *Annual Conference in Neural Information Processing Systems (NIPS)*. 2016.
- [4] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [5] E. J. Candès, X. Li, Y. Ma, and J. Wright. “Robust Principal Component Analysis?” In: *J. ACM* 58.3 (2011), 11:1–11:37.
- [6] E. J. Candès and B. Recht. “Exact matrix completion via convex optimization”. In: *Foundations of Computational Mathematics* 9.6 (2009), pp. 717–772.
- [7] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Nonconvex Nonsmooth Low-Rank Minimization via Iteratively Reweighted Nuclear Norm”. In: *IEEE Transactions on Image Processing* 25 (Oct. 2015).
- [8] M. Carlsson, D. Gerosa, and C. Olsson. “An unbiased approach to compressed sensing”. In: *arXiv preprint arXiv:1806.05283* (2018).
- [9] Y. Dai, H. Li, and M. He. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization”. In: *International Journal of Computer Vision* 107.2 (2014), pp. 101–122.
- [10] M. Fazel, H. Hindi, and S. P. Boyd. “A rank minimization heuristic with application to minimum order system approximation”. In: *American Control Conference*. 2001.
- [11] R. Ge, J. D. Lee, and T. Ma. “Matrix Completion has No Spurious Local Minimum”. In: *Annual Conference on Neural Information Processing Systems (NIPS)*. 2016.
- [12] G. Golub and V. Pereyra. “The Differentiation of Pseudo-Inverses and Nonlinear Least Squares Problems Whose Variables Separate”. In: *SIAM Journal on Numerical Analysis* 10.2 (1973), pp. 413–432.
- [13] J. H. Hong and A. Fitzgibbon. “Secrets of Matrix Factorization: Approximations, Numerics, Manifold Optimization and Random Restarts”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.
- [14] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. “Fast and Accurate Matrix Completion via Truncated Nuclear Norm Regularization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2117–2130.
- [15] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.

- [16] K. Mohan and M. Fazel. “Iterative reweighted least squares for matrix rank minimization”. In: *Annual Allerton Conference on Communication, Control, and Computing*. 2010, pp. 653–661.
- [17] T. H. Oh, Y. W. Tai, J. C. Bazin, H. Kim, and I. S. Kweon. “Partial Sum Minimization of Singular Values in Robust PCA: Algorithm and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4 (2016), pp. 744–758.
- [18] C. Olsson, M. Carlsson, and E. Bylow. “A Non-convex Relaxation for Fixed-Rank Approximation”. In: *International Conference on Computer Vision Workshops (IC-CVW)*. Oct. 2017, pp. 1809–1817.
- [19] C. Olsson, M. Carlsson, F. Andersson, and V. Larsson. “Non-Convex Rank/Sparsity Regularization and Local Minima”. In: *Proceedings of the International Conference on Computer Vision* (2017).
- [20] M. V. Örnhog, C. Olsson, and A. Heyden. “Bilinear Parameterization For Differentiable Rank-Regularization”. In: *CoRR* abs/1811.11088 (2018).
- [21] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi. “Simultaneously Structured Models With Application to Sparse and Low-Rank Matrices”. In: *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2886–2908.
- [22] S. Oymak, K. Mohan, M. Fazel, and B. Hassibi. “A simplified approach to recovery conditions for low rank matrices”. In: *IEEE International Symposium on Information Theory Proceedings (ISIT)*. 2011, pp. 2318–2322.
- [23] D. Park, A. Kyrillidis, C. Caramanis, and S. Sanghavi. “Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach”. In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS*. 2017, pp. 65–74.
- [24] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *SIAM Rev.* 52.3 (Aug. 2010), pp. 471–501.
- [25] F. Shang, J. Cheng, Y. Liu, Z. Luo, and Z. Lin. “Bilinear Factor Matrix Norm Minimization for Robust PCA: Algorithms and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.9 (Sept. 2018), pp. 2066–2080.
- [26] K.-C. Toh and S. Yun. “An Accelerated Proximal Gradient Algorithm for Nuclear Norm Regularized Least Squares Problems”. In: *Pacific Journal of Optimization* 6 (Sept. 2010).

- [27] C. Xu, Z. Lin, and H. Zha. “A Unified Convex Surrogate for the Schatten- p Norm”. In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2017.
- [28] C.-H. Zhang et al. “Nearly unbiased variable selection under minimax concave penalty”. In: *The Annals of Statistics* 38.2 (2010), pp. 894–942.

Supplementary Material

A Proofs

Theorem 1

We start by giving a complete proof of Theorem 1.

Proof of Theorem 1. Decompose $\mathcal{F}(\mathbf{B}, \mathbf{C}) = G(\mathbf{B}, \mathbf{C}) + H(\mathbf{B}, \mathbf{C})$ where $G(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) + \sum_{i=1}^k \gamma_i^2(\mathbf{B}, \mathbf{C})$ and $H(\mathbf{B}, \mathbf{C}) = -\sum_{i=1}^k \gamma_i^2(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2$. Clearly, H is differentiable, and we will show that G is convex and differentiable. By introducing

$$L(\boldsymbol{\gamma}, \mathbf{z}) = -\sum_{i=1}^{r_0} z_i^2 + 2 \sum_{i=1}^k z_i \gamma_i, \quad (27)$$

it is clear that

$$G(\mathbf{B}, \mathbf{C}) = \varphi(\boldsymbol{\gamma}(\mathbf{B}, \mathbf{C})) = \max_{\mathbf{z} \in \mathcal{Z}} L(\boldsymbol{\gamma}(\mathbf{B}, \mathbf{C}), \mathbf{z}), \quad (28)$$

where $\varphi : \mathbb{R}^k \rightarrow \mathbb{R}$, is defined as $\varphi(\mathbf{x}) = \max_{\mathbf{z} \in \mathcal{Z}} L(\mathbf{x}, \mathbf{z})$, and, as before, $\mathcal{Z} = \{\mathbf{z} \mid z_1 \geq z_2 \geq \dots \geq z_k \geq 0\}$. Clearly, \mathcal{Z} is closed, and, due to the quadratic terms, the maximisers can be restricted to a convex set (*i.e.* there exists $M \in \mathbb{R}$ such that $z_1 \leq M$).

Since $L(\mathbf{x}, \mathbf{z})$ is convex in $\mathbf{x} \in \mathbb{R}^k$ for every $\mathbf{z} \in \mathcal{Z}$, Danskin's Theorem (see e.g. [2]) is applicable. In particular, this means that φ is convex and that the directional derivative exists. Furthermore, define the set of maximising points

$$\mathcal{Z}_0(\mathbf{x}) = \left\{ \mathbf{z}^* \mid L(\mathbf{x}, \mathbf{z}^*) = \max_{\mathbf{z} \in \mathcal{Z}} L(\mathbf{x}, \mathbf{z}) \right\}, \quad (29)$$

and note that $L(\cdot, \mathbf{z})$ is differentiable at \mathbf{x} for all $\mathbf{z} \in \mathcal{Z}$. It now follows analogously to [7] that the optimising pseudo-singular value vector \mathbf{z} is given by

$$z_i \in \begin{cases} \{\max(\gamma_i, s)\}, & i \leq r_0 \\ \{s\}, & i \geq r_0, \gamma_i \neq 0 \\ [0, s], & i > r_0, \gamma_i = 0 \end{cases} \quad (30)$$

for some $s \geq \gamma_{r_0}$. (This number can be given more specifically, see e.g. [1], but we do not need the exact value to prove differentiability).

We will show that the subgradients $\partial_{\mathbf{B}} G(\mathbf{B}, \mathbf{C})$ and $\partial_{\mathbf{C}} G(\mathbf{B}, \mathbf{C})$ are singleton sets, for an arbitrary choice of (\mathbf{B}, \mathbf{C}) , for which it follows that G is differentiable.

We will use the well-known fact that [9, Theorem 10.6, Corollary 10.9]

$$\partial(h \circ c)(\mathbf{x}) = \nabla c(\mathbf{x})^T \partial h(c(\mathbf{x})), \quad (31)$$

where $h : \mathbb{R}^m \rightarrow \mathbb{R}$ is a convex and L -Lipschitz continuous function, and $c : \mathbb{R}^d \rightarrow \mathbb{R}^m$ is a \mathcal{C}^1 -smooth mapping with a β -Lipschitz continuous Jacobian map.

Recall that $\gamma : \mathbb{R}^{m \times k} \times \mathbb{R}^{n \times k} \rightarrow \mathbb{R}^k$ is defined by

$$\gamma(\mathbf{B}, \mathbf{C}) = \frac{1}{2} \left(\|\mathbf{B}_{[1]}\|^2 + \|\mathbf{C}_{[1]}\|^2, \dots, \|\mathbf{B}_{[k]}\|^2 + \|\mathbf{C}_{[k]}\|^2 \right), \quad (32)$$

where the brackets indicate they are sorted in descending order. To simplify notation, assume that the vectors are already sorted, i.e. $[i] = i$, for all i . Consequently,

$$\nabla_{\mathbf{B}} \gamma(\mathbf{B}, \mathbf{C}) = \text{blkdiag}(\mathbf{B}_1, \dots, \mathbf{B}_k) \quad \text{and} \quad \nabla_{\mathbf{C}} \gamma(\mathbf{B}, \mathbf{C}) = \text{blkdiag}(\mathbf{C}_1, \dots, \mathbf{C}_k), \quad (33)$$

where $\text{blkdiag}(\mathbf{X}_1, \dots, \mathbf{X}_k)$ is a block-diagonal matrix with the blocks $\mathbf{X}_1, \dots, \mathbf{X}_k$. Only the elements of the subgradient $\partial\varphi(\gamma)$, for which $\gamma_i = 0$ and $i > r_0$, can have non-singleton sets; however, $\gamma_i = 0$ implies that both $\mathbf{B}_{[i]}$ and $\mathbf{C}_{[i]}$ are zero vectors of suitable length, which implies that the subdifferentials at such points will be the singleton set $\{0\}$. In conclusion, $\partial_{\mathbf{B}} G(\mathbf{B}, \mathbf{C})$ and $\partial_{\mathbf{C}} G(\mathbf{B}, \mathbf{C})$, are singleton sets, and G is differentiable, and therefore also \mathcal{F} . \square

Theorem 2

In order to prove Theorem 2 we need to characterise the subdifferential, which is shown in the following lemma. Our analysis will make use of the differentiable objective

$$\mathcal{D}(\mathbf{B}, \mathbf{C}) := \tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2, \quad (34)$$

the non-convex function

$$\mathcal{N}(\mathbf{X}) := \mathcal{R}_{r_0}(\mathbf{X}) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (35)$$

and the convex function

$$\mathcal{C}(\mathbf{X}) = \mathcal{R}_{r_0}(\mathbf{X}) + \|\mathbf{X} - \mathbf{Z}\|_F^2. \quad (36)$$

We will also use the functions

$$\tilde{\mathcal{G}}(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C}) + \|\mathbf{B}\mathbf{C}^T\|_F^2, \quad (37)$$

$$G(\mathbf{X}) = \mathcal{R}_{r_0}(\mathbf{X}) + \|\mathbf{X}\|_F^2, \quad (38)$$

$$H(\mathbf{X}) = \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2 - \|\mathbf{X}\|_F^2. \quad (39)$$

Note that $\mathcal{D}(\mathbf{B}, \mathbf{C}) = \tilde{\mathcal{G}}(\mathbf{B}, \mathbf{C}) + H(\mathbf{B}\mathbf{C}^T)$ and $\mathcal{N}(\mathbf{X}) = G(\mathbf{X}) + H(\mathbf{X})$. Furthermore, note that the definition of G is different compared to the previous section.

Lemma 1. *The subdifferential of $G(\mathbf{X})$ is given by*

$$\partial G(\mathbf{X}) = \{\mathbf{U}\partial g(\boldsymbol{\Sigma})\mathbf{V}^T + \mathbf{M} : \sigma_1(\mathbf{M}) \leq s^*(\mathbf{X}), \mathbf{U}^T \mathbf{M} = \mathbf{0} \text{ and } \mathbf{M}\mathbf{V}^T = \mathbf{0}\} \quad (40)$$

where $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ is the SVD and $\partial g(\boldsymbol{\Sigma})$ is the matrix of same size as $\boldsymbol{\Sigma}$ with diagonal elements $\partial g(\sigma_i)$.

Proof. Write $G(\mathbf{X}) = g \circ \sigma(\mathbf{X})$, where $\sigma(\mathbf{X})$ is the vector of singular values of \mathbf{X} and g is an absolutely symmetric function. Then, according to [6], the matrix $\mathbf{Y} \in \partial G(\mathbf{X})$ if and only if $\mathbf{Y} = \mathbf{U}' \text{diag}(\partial g \circ \sigma(\mathbf{X}))\mathbf{V}'^T$ when $\mathbf{X} = \mathbf{U}' \text{diag}(\sigma(\mathbf{X}))\mathbf{V}'^T$. (Here we use the full SVD with square orthogonal matrices \mathbf{U}' and \mathbf{V}' .) Now given a thin SVD $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ all possible full SVD's of \mathbf{X} can be written

$$\mathbf{X} = [\mathbf{U} \quad \mathbf{U}_\perp] \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}^T \\ \mathbf{V}_\perp^T \end{bmatrix}, \quad (41)$$

where \mathbf{U}_\perp and \mathbf{V}_\perp are singular vectors corresponding to singular values that are zero. Note that \mathbf{U}_\perp and \mathbf{V}_\perp are not uniquely defined since their corresponding singular values are all zero. Therefore we get

$$\begin{aligned} \mathbf{Y} &= [\mathbf{U}' \quad \mathbf{U}'_\perp] \begin{bmatrix} \partial g(\boldsymbol{\Sigma}) & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{V}'^T \\ \mathbf{V}'_\perp^T \end{bmatrix} \\ &= \mathbf{U}' \partial g(\boldsymbol{\Sigma})\mathbf{V}'^T + \mathbf{U}'_\perp \mathbf{D}\mathbf{V}'_\perp^T, \end{aligned} \quad (42)$$

where \mathbf{D} is a diagonal matrix with elements in $[-s^*(\mathbf{X}), s^*(\mathbf{X})]$. It is now clear that $\sigma_1(\mathbf{D}) = \sigma_1(\mathbf{U}'_\perp \mathbf{D}\mathbf{V}'_\perp^T) \leq s^*(\mathbf{X})$. Furthermore, since \mathbf{U}'_\perp and \mathbf{V}'_\perp can be any orthogonal bases of the spaces perpendicular to the column and row spaces of \mathbf{X} , it is clear that any matrix \mathbf{M} fulfilling $\mathbf{U}'^T \mathbf{M} = \mathbf{0}$, $\mathbf{M}\mathbf{V}' = \mathbf{0}$ and $\sigma_1(\mathbf{M}) \leq s^*(\mathbf{X})$ can be written $\mathbf{M} = \mathbf{U}'_\perp \mathbf{D}\mathbf{V}'_\perp^T$, hence

$$\partial G(\mathbf{X}) = \{\mathbf{U}\partial g(\boldsymbol{\Sigma})\mathbf{V}^T + \mathbf{M} : \sigma_1(\mathbf{M}) \leq s^*(\mathbf{X}), \mathbf{U}^T \mathbf{M} = \mathbf{0}, \mathbf{M}\mathbf{V} = \mathbf{0}\}. \quad (43)$$

□

The proof of Theorem 2 now follows identically to that of [8, Theorem 2], by replacing the subdifferential of the “soft rank” regulariser with the one obtained in Lemma 1.

Theorem 3

We use the same notation as in the previous section. For this we will need the following result:

Lemma 2. *If $\tilde{\mathbf{X}}$ is a solution to $\min_{\text{rank}(\mathbf{X}) \leq k} \mathcal{C}(\mathbf{X})$, with $\text{rank}(\tilde{\mathbf{X}}) = r_0 < k$, then $\tilde{\mathbf{X}}$ also solves $\min_{\mathbf{X}} \mathcal{C}(\mathbf{X})$. If the singular values of \mathbf{Z} fulfil $\sigma_{r_0+1}(\mathbf{Z}) \neq \sigma_{r_0}(\mathbf{Z})$, then $\tilde{\mathbf{X}}$ is the unique minimizer.*

Proof. Note that $\mathcal{C}(\mathbf{X})$ is the l.s.c. convex envelope of $\mathbb{I}(\text{rank}(\mathbf{X}) \leq r_0) + \|\mathbf{X} - \mathbf{Z}\|_F^2$, and, therefore, the global minimum is the same. The solution, which can be obtained in closed form through SVD, is unique if and only if $\sigma_{r_0+1}(\mathbf{Z}) \neq \sigma_{r_0}(\mathbf{Z})$. \square

We are now ready to prove Theorem 3.

Proof of Theorem 3. Since \mathcal{C} and \mathcal{N} has the same subdifferential (see [5]) at $\mathbf{X} = \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ it is clear that the directional derivatives $\mathcal{C}'_{\Delta\mathbf{X}}(\tilde{\mathbf{X}}) = \mathcal{N}'_{\Delta\mathbf{X}}(\tilde{\mathbf{X}}) \geq 0$, where $\Delta\mathbf{X} = \tilde{\mathbf{X}} - \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ and $\text{rank}(\tilde{\mathbf{X}}) \leq r_0$. By convexity of \mathcal{C} it is then also clear that

$$\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T \in \underset{\text{rank}(\mathbf{X}) \leq k}{\arg \min} \mathcal{C}(\mathbf{X}). \quad (44)$$

Since $\text{rank}(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T) = r_0 < k$, $\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ is also the unrestricted global minimizer of $\mathcal{C}(\mathbf{X})$ according to Lemma 2. By Lemma 1.1 of [7], it is then a stationary point of $\mathcal{N}(\mathbf{X})$.

What remains now is to prove that $\tilde{\mathbf{X}} = \tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ is a minimizer over all line segments $\tilde{\mathbf{X}} + t\Delta\mathbf{X}$. This can be done by estimating the growth of the directional derivatives along such lines. Note that $\tilde{\mathbf{X}}$ is a stationary point of $\mathcal{N}(\mathbf{X}) = G(\mathbf{X}) + H(\mathbf{X})$ if and only if $-\nabla H(\tilde{\mathbf{X}}) = 2\mathbf{Z} \in \partial G(\tilde{\mathbf{X}})$.

Since $\nabla H(\tilde{\mathbf{X}} + t\Delta\mathbf{X}) - \nabla H(\tilde{\mathbf{X}}) = t\nabla H(\Delta\mathbf{X}) = 2t(\mathcal{A}^* \mathcal{A}\Delta\mathbf{X} - \Delta\mathbf{X})$ we have

$$\langle \nabla H(\tilde{\mathbf{X}} + t\Delta\mathbf{X}) - \nabla H(\tilde{\mathbf{X}}), t\Delta\mathbf{X} \rangle = 2t^2(\|\mathcal{A}\Delta\mathbf{X}\|_F^2 - \|\Delta\mathbf{X}\|_F^2), \quad (45)$$

and due to RIP $\|\mathcal{A}\Delta\mathbf{X}\|_F^2 - \|\Delta\mathbf{X}\|_F^2 \geq -\delta_{2r_0}\|\Delta\mathbf{X}\|_F^2$. From Corollary 2.3 [7] we see that for any $2\mathbf{Z}' \in \partial G(\tilde{\mathbf{X}} + t\Delta\mathbf{X})$ we have

$$\langle \mathbf{Z}' - \mathbf{Z}, t\Delta\mathbf{X} \rangle > \frac{t^2(1-c)}{2} \|\Delta\mathbf{X}\|_F^2, \quad (46)$$

as long as $t \neq 0$. Since the directional derivatives $G'_{\Delta\mathbf{X}}(\mathbf{X}) = \max_{2\mathbf{Z}' \in \partial G(\mathbf{X})} \langle 2\mathbf{Z}', \Delta\mathbf{X} \rangle$, $H'_{\Delta\mathbf{X}}(\mathbf{X}) = \langle \nabla H(\mathbf{X}), \Delta\mathbf{X} \rangle$ and $2\mathbf{Z} + \nabla H'(\tilde{\mathbf{X}}) = 0$ we get

$$\mathcal{N}'_{\Delta\mathbf{X}}(\tilde{\mathbf{X}} + t\Delta\mathbf{X}) \geq \langle 2\mathbf{Z}' + \nabla H(\tilde{\mathbf{X}} + t\Delta\mathbf{X}), \Delta\mathbf{X} \rangle > 0. \quad (47)$$

\square

B Implementation Details

Next, we give a more detailed overview of our proposed method, which consists of three main parts. Initially, we consider the first order Taylor expansion (termsize) of the regularisation term by replacing $\tilde{\mathcal{R}}_{r_0}(\mathbf{B}, \mathbf{C})$ and instead consider the iteratively reweighted term (21), $\sum_{i=1}^k w_i^{(t)} \left(\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2 \right)$ as described in Section 5.

Then, we apply one step of VarPro with the Ruhe Wedin (RW-2) approximation, see [3] for details on the implementation. Since the Jacobians with respect to both the \mathbf{B} and \mathbf{C} parameters are necessary for updating the VarPro algorithm, they must be linearised. The contribution to the the regularisation term is thus given by

$$\|\text{diag}(\mathbf{w}^{(t)})\mathbf{B}\|_F^2 + \|\text{diag}(\mathbf{w}^{(t)})\mathbf{C}\|_F^2, \quad (48)$$

where $\text{diag}(\mathbf{w}^{(t)})$ is a diagonal matrix with the weights $w_i^{(t)}$ in the diagonal. Note that, the residuals $\text{diag}(\mathbf{w}^{(t)})\mathbf{B}$ are linear. We can therefore write them as $J_B^{\text{reg}}\mathbf{b}$, where \mathbf{b} is a column stacked version of \mathbf{B} . In the case where \mathbf{B} has k columns, the matrix J_B^{reg} will consist of k instances of the matrix $\text{diag}(\mathbf{w}^{(t)})$. Furthermore, each row of J_B^{reg} has only one non-zero element making the matrix extremely sparse. In a similar manner, we obtain the contribution due to the second bilinear factor \mathbf{C} , which we write as $J_C^{\text{reg}}\mathbf{c}$. We use the convention $\mathbf{c} = \text{vec}(\mathbf{C}^T)$, as it makes the computations of the data terms easier. Due to this choice J_C^{reg} consists of a k permuted copies of $\text{diag}(\mathbf{w}^{(t)})$. Given a current iterate $(\mathbf{b}^{(t)}, \mathbf{c}^{(t)})$ the regularisation term becomes $\|J_B^{\text{reg}}\delta\mathbf{b} + \mathbf{r}_B\|^2 + \|J_C^{\text{reg}}\delta\mathbf{c} + \mathbf{r}_C\|^2$, where $\mathbf{r}_B = J_B^{\text{reg}}\mathbf{b}^{(t)}$, $\mathbf{r}_C = J_C^{\text{reg}}\mathbf{c}^{(t)}$, $\mathbf{b} = \mathbf{b}^{(t)} + \delta\mathbf{b}$ and $\mathbf{c} = \mathbf{c}^{(t)} + \delta\mathbf{c}$.

Then, consider linearising the residuals $\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}$ around $(\mathbf{b}^{(t)}, \mathbf{c}^{(t)})$, which gives an expression of the form

$$J_B^{\text{data}}\delta\mathbf{b} + J_C^{\text{data}}\delta\mathbf{c} + \mathbf{r}^{\text{data}}. \quad (49)$$

The sparsity pattern of the Jacobians, as well as their dependency, depends on the application; however, for the NRSfM applications they are sparse (block-diagonal), making them easy to compute and invert. Locally, the objective function is approximated by

$$\|J_B\delta\mathbf{b} + J_C\delta\mathbf{c} + \mathbf{r}\|^2, \quad (50)$$

where

$$J_B = \begin{bmatrix} J_B^{\text{reg}} \\ 0 \\ J_B^{\text{data}} \end{bmatrix}, \quad J_C = \begin{bmatrix} 0 \\ J_C^{\text{reg}} \\ J_C^{\text{data}} \end{bmatrix}, \quad \mathbf{r} = \begin{bmatrix} \mathbf{r}_B \\ \mathbf{r}_C \\ \mathbf{r}^{\text{data}} \end{bmatrix}. \quad (51)$$

In [4] it was shown that each step of VarPro is equivalent to first minimising (50) with the additional dampening term $\lambda \|\delta \mathbf{b}\|^2$ and then performing an exact optimisation of (21) over the \mathbf{C} -variables (when fixing the \mathbf{B} -variables to their new values). Due to the reweighting, we only do one iteration with VarPro before updating the weights $\mathbf{w}^{(t)}$.

Since the above steps may return stationary points where $\tilde{\mathcal{R}}(\mathbf{B}, \mathbf{C}) > \mathcal{R}(\mathbf{B}\mathbf{C}^T)$, we perform a last step to escape such points. This is done by taking the current iterate and recompute the factorisation of $\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T$ using SVD. If the SVD of $\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T = \sum_{i=1}^r \sigma_i \mathbf{U}_i \mathbf{V}_i^T$ we proceed to update $\tilde{\mathbf{B}}$ and $\tilde{\mathbf{C}}$ to $\tilde{\mathbf{B}}_i = \sqrt{\sigma_i} \mathbf{U}_i$ and $\tilde{\mathbf{C}}_i = \sqrt{\sigma_i} \mathbf{V}_i$ which is guaranteed to reduce the energy, and yields $\tilde{\mathcal{R}}(\tilde{\mathbf{B}}, \tilde{\mathbf{C}}) = \mathcal{R}(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T)$. The detailed steps of the bilinear method are summarised in Algorithm 1.

Input: Linear operator \mathcal{A} and desired maximum output rank r_0 , damping parameter λ .

Initialise \mathbf{B} and \mathbf{C} with random entries

while not converged do

```

    Compute weights  $\mathbf{w}^{(t)}$  from current iterate  $(\mathbf{B}, \mathbf{C})$ 
    Compute the vectorisations  $\mathbf{b} = \text{vec}(\mathbf{B})$ ,  $\mathbf{c} = \text{vec}(\mathbf{C}^T)$ 
    Compute residuals  $\mathbf{r}_B$ ,  $\mathbf{r}_C$ , and Jacobians  $\mathbf{J}_B^{\text{data}}$  and  $\mathbf{J}_C^{\text{data}}$  depending on  $\mathcal{A}$ 
    Compute residual  $\mathbf{r}^{\text{reg}}$ , and Jacobians  $\mathbf{J}_B^{\text{reg}}$  and  $\mathbf{J}_C^{\text{reg}}$ 
    Create full residual  $\mathbf{r}$  and Jacobians  $\mathbf{J}_B$  and  $\mathbf{J}_C$ 
    Compute  $\tilde{\mathbf{J}}^T \tilde{\mathbf{J}} + \lambda \mathbf{I} = \mathbf{J}_B^T (\mathbf{I} - \mathbf{J}_C \mathbf{J}_C^+) \mathbf{J}_B + \lambda \mathbf{I}$ 
    Compute  $\mathbf{b}' = \mathbf{b} - (\tilde{\mathbf{J}}^T \tilde{\mathbf{J}} + \lambda \mathbf{I})^{-1} \mathbf{J}_B \mathbf{r}$  and reshape into matrix  $\mathbf{B}'$ 
    Compute  $\mathbf{C}'$  by minimising (21) with fixed  $\mathbf{B}'$ 
    if  $\mathcal{R}(\mathbf{B}'\mathbf{C}'^T) + \|\mathcal{A}(\mathbf{B}'\mathbf{C}'^T) - \mathbf{b}\|^2 < \mathcal{R}(\mathbf{B}\mathbf{C}^T) + \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2$  then
         $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \text{svd}(\mathbf{B}'\mathbf{C}'^T)$ 
        Update  $\mathbf{B} = \mathbf{U}\sqrt{\mathbf{\Sigma}}$  and  $\mathbf{C} = \mathbf{V}\sqrt{\mathbf{\Sigma}}$ 
        Decrease  $\lambda$ 
    else
        Increase  $\lambda$ 
    end

```

end

Algorithm 1: Outline of the bilinear method.

References

- [1] F. Andersson, M. Carlsson, and C. Olsson. “Convex envelopes for fixed rank approximation”. In: *Optimization Letters* 11.8 (Dec. 2017), pp. 1783–1795.
- [2] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.
- [3] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.
- [4] J. H. Hong, C. Zach, A. Fitzgibbon, and R. Cipolla. “Projective Bundle Adjustment from Arbitrary Initialization Using the Variable Projection Method”. In: *European Conference on Computer Vision (ECCV)*. 2016, pp. 477–493.
- [5] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [6] A. S. Lewis. “The convex analysis of unitarily invariant matrix functions”. In: *Journal of Convex Analysis* 2.1 (1995), pp. 173–183.
- [7] C. Olsson, M. Carlsson, and E. Bylow. “A Non-convex Relaxation for Fixed-Rank Approximation”. In: *International Conference on Computer Vision Workshops (ICCVW)*. Oct. 2017, pp. 1809–1817.
- [8] M. V. Örnhog, C. Olsson, and A. Heyden. “Bilinear Parameterization For Differentiable Rank-Regularization”. In: *CoRR* abs/1811.11088 (2018).
- [9] R. Rockafellar and R. J.-B. Wets. *Variational Analysis*. Heidelberg, Berlin, New York: Springer Verlag, 1998.

A Unified Optimization Framework for Low-Rank Inducing Penalties

MARCUS VALTONEN ÖRNHAG¹ AND CARL OLSSON^{1,2}

¹*Centre for Mathematical Sciences, Lund University*

²*Department of Electrical Engineering, Chalmers University of Technology*

Abstract: In this paper we study the convex envelopes of a new class of functions. Using this approach, we are able to unify two important classes of regularizers from unbiased non-convex formulations and weighted nuclear norm penalties. This opens up for possibilities of combining the best of both worlds, and to leverage each method's contribution to cases where simply enforcing one of the regularizers are insufficient.

We show that the proposed regularizers can be incorporated in standard splitting schemes such as Alternating Direction Methods of Multipliers (ADMM), and other subgradient methods. Furthermore, we provide an efficient way of computing the proximal operator.

Lastly, we show on real non-rigid structure-from-motion (NRSFM) datasets, the issues that arise from using weighted nuclear norm penalties, and how this can be remedied using our proposed method.¹

1 Introduction

Dimensionality reduction using Principal Component Analysis (PCA) is widely used for all types of data analysis, classification and clustering. In recent years, numerous subspace clustering methods have been proposed, to address the shortcomings of traditional PCA methods. The work on Robust PCA by Candès *et al.* [7] is one of the most influential papers on the subject, which sparked a large research interest from various fields including computer vision. Applications include, but are not limited to, rigid and non-rigid structure-from-motion [1, 4], photometric stereo [2] and optical flow [13].

It is well-known that the solution to

$$\min_{\text{rank}(\mathbf{X}) \leq r} \|\mathbf{X} - \mathbf{X}_0\|_F^2, \quad (1)$$

¹Code available: <https://github.com/marcusvaltonen/UnifiedFramework>. This work was supported by the Swedish Research Council (grants no. 2015-05639 and 2018-05375) and the Swedish Foundation for Strategic Research (Semantic Mapping and Visual Navigation for Smart Robots).

where $\|\cdot\|_F$ is the Frobenius norm, can be given in closed form using the singular value decomposition (SVD) of the measurement matrix \mathbf{X}_0 . The character of the problem changes drastically, when considering objectives such as

$$\min_{\text{rank}(\mathbf{X}) \leq r} \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (2)$$

where $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$ is a linear operator, $\mathbf{b} \in \mathbb{R}^p$, and $\|\cdot\|$ is the standard Euclidean norm. In fact, such problems are in general known to be NP hard [14]. In many cases, however, the rank is not known *a priori*, and a “soft rank” penalty can be used instead

$$\min_{\mathbf{X}} \mu \text{rank}(\mathbf{X}) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2. \quad (3)$$

Here, the regularization parameter μ controls the trade-off between enforcing the rank and minimizing the residual error, and can be tuned to problem specific applications.

In order to treat objectives of the form (2) and (3), a convex surrogate of the rank penalty is often used. One popular approach is to use the nuclear norm [7, 30]

$$\|\mathbf{X}\|_* = \sum_{i=1}^n \sigma_i(\mathbf{X}), \quad (4)$$

where $\sigma_i(\mathbf{X}), i = 1, \dots, n$, is the i :th singular value of \mathbf{X} . One of the drawbacks of using the nuclear norm penalty is that both large and small singular values are penalized equally hard. This is referred to as shrinking bias, and to counteract such behavior, methods penalizing small singular values (assumed to be noise) harder have been proposed [9, 16, 21, 23, 26, 27, 29, 32].

1.1 Related Work

Our work is a generalization of Larsson and Olsson [21]. They considered problems on the form

$$\min_{\mathbf{X}} g(\text{rank}(\mathbf{X})) + \|\mathbf{X} - \mathbf{X}_0\|_F^2, \quad (5)$$

where the regularizer g is non-decreasing and piecewise constant,

$$g(k) = \sum_{i=1}^k g_i. \quad (6)$$

Note, that for $g_i \equiv \mu$ we obtain (3). Furthermore, if we let $g_i = 0$ for $i \leq r_0$, and ∞ otherwise, (2) is obtained. The objectives (5) are difficult to optimize, as they, in general, are non-convex and discontinuous. Thus, it is natural to consider a relaxation

$$\min_{\mathbf{X}} \mathcal{R}_g(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2, \quad (7)$$

where

$$\mathcal{R}_g(\mathbf{X}) = \max_{\mathbf{Z}} \left(\sum_{i=1}^n \min(g_i, \sigma_i^2(\mathbf{Z})) - \|\mathbf{X} - \mathbf{Z}\|_F^2 \right). \quad (8)$$

It was shown in [21], that this is the convex envelope of (5), hence share the same global minimizers.

Another type of regularizer that has been successfully used in low-level imaging applications [15, 36, 37] is the weighted nuclear norm (WNNM),

$$\|\mathbf{X}\|_{w,*} = \sum_{i=1}^k w_i \sigma_i(\mathbf{X}), \quad (9)$$

where $\mathbf{w} = (w_1, \dots, w_k)$ is a weight vector. Note that the WNNM formulation does not fit the assumptions (6), hence cannot be considered in this framework.

For certain applications, it is of interest to include both regularizers, which we will show in Section 6. Typically, this is preferable when the rank constraint alone is not strong enough to yield accurate reconstructions, and further penalization is necessary to restrict the solutions. To this end, we suggest to merge these penalties. In [28] a similar approach was suggested, but is not general enough to include penalties like WNNM.

Our main contributions are:

- A novel method for combining bias reduction and non-convex low-rank inducing objectives,
- An efficient and fast algorithm to compute the proposed regularizer,
- Theoretical insight in the quality of reconstructed missing data using WNNM, and practical demonstrations on how shrinking bias is perceived in these applications,
- A new objective for Non-Rigid Structure from Motion (NRSfM), with improved performance, compared to state-of-the-art prior-free methods, capable of working in cases where the image sequences are unordered.

First, however, we will lay the ground for the theory of a common framework of low-rank inducing objectives.

2 Problem Formulation and Motivation

In this paper we propose a new class of regularization terms for low rank matrix recovery problems that controls both the rank and the magnitude of the singular values of the

recovered matrix. Our objective function has the form

$$f_h(\mathbf{X}) = h(\sigma(\mathbf{X})) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (10)$$

where $h(\sigma(\mathbf{X})) = \sum_{i=1}^k h_i(\sigma_i(\mathbf{X}))$ and

$$h_i(\sigma_i(\mathbf{X})) = \begin{cases} 2a_i\sigma_i(\mathbf{X}) + b_i & \sigma_i(\mathbf{X}) \neq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

We assume that the sequences $\{a_i\}_{i=1}^k$ and $\{b_i\}_{i=1}^k$ are both non-decreasing.

Our approach unifies the formulation of [19] with weighted nuclear norm. The terms $2a_i\sigma_i(\mathbf{X})$ correspond to the singular value penalties of a weighted nuclear norm [15]. These can be used to control the sizes of the non-zero singular values. In contrast, the constants b_i corresponds to a rank penalization that is independent of these sizes and, as we will see in the next section, enables bias free rank selection.

2.1 Controlled Bias and Rank Selection

To motivate the use of both sets of variables $\{a_i\}_{i=1}^k$ and $\{b_i\}_{i=1}^k$, and to understand their purpose, we first consider the simple recovery problem $\min_{\mathbf{X}} f_h(\mathbf{X})$, where

$$f_h(\mathbf{X}) := h(\sigma(\mathbf{X})) + \|\mathbf{X} - \mathbf{X}_0\|_F^2. \quad (12)$$

Here \mathbf{X}_0 is assumed to consist of a set of large singular values $\sigma_i(\mathbf{X}_0)$, $i = 1, \dots, r$, corresponding to the matrix we wish to recover, and a set of small ones $\sigma_i(\mathbf{X}_0)$, $i = r + 1, \dots, k$, corresponding to noise that we want to suppress.

Due to von Neumann's trace theorem [22] the solution can be computed in closed form by considering each singular values separately, and minimize

$$\begin{cases} 2a_i\sigma_i(\mathbf{X}) + b_i + (\sigma_i(\mathbf{X}) - \sigma_i(\mathbf{X}_0))^2 & \sigma_i(\mathbf{X}) \neq 0, \\ \sigma_i(\mathbf{X}_0)^2 & \sigma_i(\mathbf{X}) = 0, \end{cases} \quad (13)$$

over $\sigma_i(\mathbf{X}) \geq 0$. Differentiating for the case $\sigma_i(\mathbf{X}) \neq 0$ gives a stationary point at $\sigma_i(\mathbf{X}) = \sigma_i(\mathbf{X}_0) - a_i$ if $\sigma_i(\mathbf{X}_0) - a_i > 0$. Since this point has objective value $2a_i\sigma_i(\mathbf{X}_0) - a_k^2 + b_k$ it is clear that this point will be optimal if

$$2a_i\sigma_i(\mathbf{X}_0) - a_i^2 + b_i \leq \sigma_i(\mathbf{X}_0)^2, \quad (14)$$

or equivalently $\sigma_i(\mathbf{X}_0) - a_i \geq \sqrt{b_i}$. Summarizing, we thus get the optimal singular values

$$\sigma_i(\mathbf{X}) = \begin{cases} \sigma_i(\mathbf{X}_0) - a_i & \sigma_i(\mathbf{X}_0) - a_i \geq \sqrt{b_i}, \\ 0 & \text{otherwise.} \end{cases} \quad (15)$$

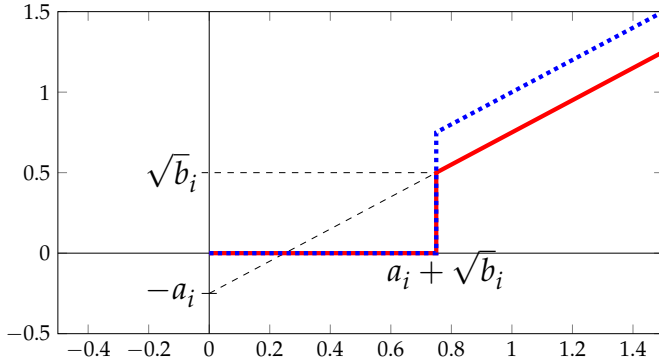


Figure 1: The optimal recovered singular value $\sigma_i(\mathbf{X})$ as a function (red curve) of the observed $\sigma_i(\mathbf{X}_0)$.

Note, that this is a valid sequence of singular values since under our assumptions $\sigma_i(\mathbf{X}_0) - a_i$ is decreasing and $\sqrt{b_i}$ increasing. The red curve of Figure 1 shows the recovered singular value as a function of the corresponding observed singular value. For comparison, we also plot the dotted blue curve which shows hard thresholding at $a_i + \sqrt{b_i}$, *i.e.* singular values smaller than $a_i + \sqrt{b_i}$ vanish while the rest are left unaltered.

Now, suppose that we want to recover the largest singular values unchanged. Using the weighted nuclear norm ($b_i = 0$) it is clear that this can only be done if we know that the sought matrix has rank r and let $a_i = 0$ for $i = 1, \dots, r$. For any other setting the regularization will subtract a_i from the corresponding non-zero singular value. In contrast, by letting $a_i = 0$ allows exact recovery of the large singular values by selecting b_i appropriately even when the rank is unknown. Hence, in the presence of a weak prior on the rank of the matrix, using only the b_i (the framework in [21]) allows exact recovery for a more general set of problems than use of the a_i (weighted nuclear norm formulations).

The above class of problems are well posed with a strong data term $\|\mathbf{X} - \mathbf{X}_0\|_F^2$. For problems with weaker data terms, priors on the magnitude of the singular values can still be very useful. In the context of NRSfM it has been observed [12, 27] that adding a bias can improve the distance to the ground truth reconstruction, even though it does not alter the rank. The reason is that, when the scene is not rigid, several reconstructions with the same rank may co-exist, thus resulting in similar projections. By introducing bias on the singular values, further regularization is enforced on the deformations, which may aid in the search for correct reconstructions. For example, with $a_1 = 0$ and $a_i > 0$, $i > 1$

we obtain a penalty that favors matrices that “are close to” rank 1. In the formulation of [12], where rank 1 corresponds to a rigid scene this can be thought of as an “as rigid as possible” prior, which is realistic in many cases [18, 24, 31, 33], but which has yet to be considered in the context of factorization methods.²

2.2 The Quadratic Envelope

As discussed above the two sets of parameters $\{a_i\}$ and $\{b_i\}$ have complementary regularization effects. The main purpose of unifying them is to create more flexible priors allowing us to do accurate rank selection with a controlled bias. In the following sections, we also show that they have relaxations that can be reliably optimized. Specifically, the resulting formulation $h(\sigma(\mathbf{X}))$, which is generally non-convex and discontinuous, can be relaxed by computing the so called quadratic envelope [10, 11]. The resulting relaxation $\mathcal{R}_h(\sigma(\mathbf{X}))$ is continuous and in addition $\mathcal{R}_h(\sigma(\mathbf{X})) + \|\mathbf{X} - \mathbf{X}_0\|_F^2$ is convex. For a more general data term there may be multiple local minimizers. However, it is known that

$$h(\sigma(\mathbf{X})) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (16)$$

and

$$\mathcal{R}_h(\sigma(\mathbf{X})) + \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (17)$$

have the same global minimizer when $\|\mathcal{A}\| < 1$ [10]. In addition, potential local minima of (17) are also local minima of (16); however, the converse does not hold. We also show that the proximal operator of $\mathcal{R}_h(\sigma(\mathbf{X}))$ can be efficiently computed which allows simple optimization using splitting methods such as ADMM [3].

3 A New Family of Functions

Consider functions on the form (12). This is a generalization of [21]; and the derivation for our objective follows a similar structure. We outline this in detail in the supplementary material, where we show that convex envelope f_h^{**} is given by

$$f_h^{**}(\mathbf{X}) = \mathcal{R}_h(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2, \quad (18)$$

²To regularize the problem Dai *et al.* incorporated a penalty of the derivatives of the 3D tracks, which also can be seen as a prior preferring rigid reconstructions. However, this option is not feasible for unsorted image collections.

where

$$\mathcal{R}_h(\mathbf{X}) := \max_{\mathbf{Z}} \left(\sum_{i=1}^n \min \left(b_i, [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right) + \|\mathbf{Z}\|_F^2 - \|\mathbf{X} - \mathbf{Z}\|_F^2 - \sum_{i=1}^n [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right). \quad (19)$$

As in [21], the optimization can be reduced to the singular values only,

$$\mathcal{R}_h(\mathbf{X}) = \max_{\sigma(\mathbf{Z})} \left(\sum_{i=1}^n \min \left(b_i, [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right) + \sigma_i^2(\mathbf{Z}) - (\sigma_i(\mathbf{X}) - \sigma_i(\mathbf{Z}))^2 - [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right). \quad (20)$$

This can be achieved by applying von Neumann's trace theorem (see supplementary material). The optimization problem is concave, hence can be solved with standard convex solvers such as MOSEK or CVX; however, in the next section we show that the problem can be turned into a series of one-dimensional problems, and the resulting algorithm for computing (19) is magnitudes faster than applying a general purpose solver.

4 Finding the Maximizing Sequence

Following the approach used in [21], consider the program

$$\begin{aligned} & \max_s \quad f(s) \\ & \text{s.t.} \quad \sigma_{i+1}(\mathbf{Z}) \leq s \leq \sigma_{i-1}(\mathbf{Z}). \end{aligned} \quad (21)$$

where $\sigma_i(\mathbf{Z})$ is the i :th singular value of the maximizing sequence in (20), and

$$f(s) = \min \{ b_i, [s - a_i]_+^2 \} - (s - \sigma_i(\mathbf{X}))^2 + s^2 - [s - a_i]_+^2. \quad (22)$$

The objective function f can be seen as the pointwise minimum of two concave functions, namely, $f_1(s) = b_i + 2\sigma_i(\mathbf{X})s - \sigma_i^2(\mathbf{X}) - [s - a_i]_+^2$ and $f_2(s) = 2\sigma_i(\mathbf{X})s - \sigma_i(\mathbf{X})^2$, *i.e.* $f(s) = \min\{f_1(s), f_2(s)\}$, hence f is concave.

The individual unconstrained optimizers are given by $s_i = a_i + \max\{\sqrt{b_i}, \sigma_i(\mathbf{X})\}$. In previous work [21], where $a_i \equiv 0$, an algorithm was devised to find the maximizing singular vector, by turning it to an optimization problem of a single variable. This method is not directly applicable, as the sequence $\{s_i\}_{i=1}^k$, in general, does not satisfy

Data: Weights \mathbf{a} , \mathbf{b} , and $\sigma(\mathbf{X})$

Result: Maximizing vector $\sigma(\mathbf{Z}^*) = \{\sigma_i\}_i$

Initialize with the unconstrained maximizers $\sigma_i = a_i + \max\{\sqrt{b_i}, \sigma_i(\mathbf{X})\}$;

while $\sigma(\mathbf{Z}^*)$ is not a valid singular value vector **do**

Find local extrema of $\sigma(\mathbf{Z}^*)$ and generate subintervals $\{t_k\}_{k \in \mathcal{I}}$;

for $k \in \mathcal{I}$ **do**

Find scalar $s^* = \operatorname{argmax}_s f(s)$ where f is defined in (22);

Update $\sigma_i = s^*$ for all $i \in t_k$.

end

end

Algorithm 1: Algorithm for finding the maximizing singular value vector.

the necessary conditions³. In fact, the number of local extrema in the sequence $\{s_i\}_{i=1}^k$ is only limited by its length. We show an example of a sequence in Figure 2, and the corresponding maximizing sequence. Nevertheless, it is possible to devise an algorithm that returns the maximizing singular value vector, as we will show shortly.

In order to do so, we can apply some of the thoughts behind the proof behind [21]. Consider the more general optimization problem of minimizing $g(\sigma) = \sum_{i=1}^k f_i(\sigma_i)$, subject to $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k \geq 0$, where f_i are concave. Then, given the unconstrained sequence of minimizers $\{s_i\}_{i=1}^k$, the elements of the constrained sequence $\{\sigma_i\}_{i=1}^k$ can be limited to three choices

$$\sigma_i = \begin{cases} s_i & \text{if } \sigma_{i+1} \leq s_i \leq \sigma_{i-1}, \\ \sigma_{i-1} & \text{if } \sigma_{i-1} < s_i, \\ \sigma_{i+1} & \text{if } s_i < \sigma_{i+1}. \end{cases} \quad (23)$$

Furthermore, the regions between local extreme points (of the unconstrained singular values) are constant.

Lemma 1. *Assume s_p and s_q are local extrema of $\{s_i\}_{i=1}^k$ and that the subsequence $\{s_i\}_{i=p}^q$ are non-decreasing. Then the corresponding subsequence of the constrained problem $\{\sigma_i\}_{i=p}^q$ is constant.*

Proof. Consider σ_i for some $p \leq i \leq q - 1$. If $\sigma_i > s_i$, then by (23) we have $\sigma_{i+1} = \sigma_i$. If instead $\sigma_i \leq s_i$, we have $\sigma_{i+1} \leq \sigma_i \leq s_i \leq s_{i+1}$ and by (23), $\sigma_{i+1} = \sigma_i$. \square

³In order to use the algorithm, the sequence $\{s_i\}_{i=1}^k$ must be non-increasing for $i < p$, non-decreasing for $p \leq i \leq q$, and non-increasing for $i > q$, for some p , and q .

We can now devise an algorithm that returns the maximizing sequence, see Algorithm 1. Essentially, the algorithm starts at the unconstrained solution, and then adds more constraints, by utilizing Lemma 1, until all of them are fulfilled.

Theorem 1. *Algorithm 1 returns the maximizing sequence.*

Proof. See the supplementary material. □

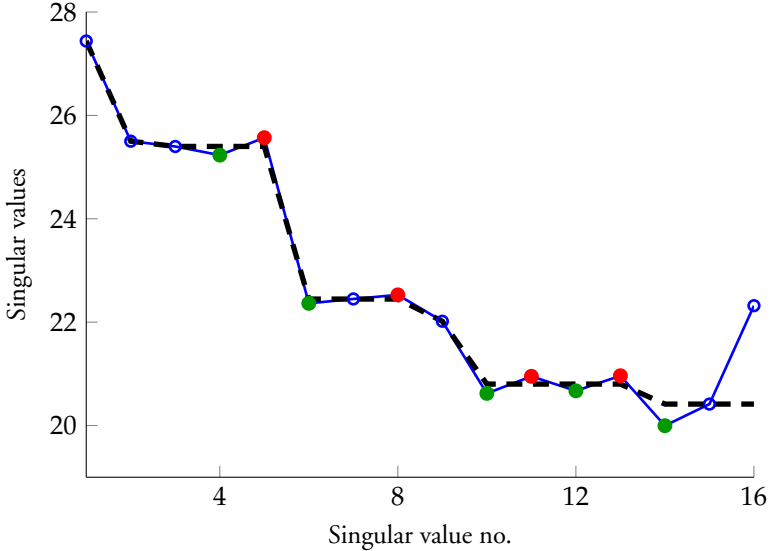


Figure 2: Example of a sequence of unconstrained maximizers (blue line), local extrema (green and red) and the maximizing sequences (dashed black) obtained by Algorithm 1.

5 ADMM and the Proximal Operator

We employ the splitting method ADMM [3], which is a standard tool for problems of this type. Thus, consider the augmented Lagrangian

$$L(\mathbf{X}, \mathbf{Y}, \mathbf{\Lambda}) = f_h^{**}(\mathbf{X}) + \rho \|\mathbf{X} - \mathbf{Y} + \mathbf{\Lambda}\|_F^2 + \mathcal{C}(\mathbf{Y}) - \rho \|\mathbf{\Lambda}\|_F^2, \quad (24)$$

where \mathbf{X} and \mathbf{Y} are minimized sequentially, and $\mathbf{\Lambda}$ is the dual variable. All variables are of the same dimensionality. The function \mathcal{C} is assumed to be convex and incorporates

Table 1: Distance to ground truth (normalized) mean valued over 20 problem instances for different percentages of missing data and data patterns. The standard deviation of the noise is kept constant at $\sigma = 0.1$. Best results are marked in bold.

Missing data (%)	PCP [6]	WNNM [15]	Unifying [5]	LpSq [25]	S12L12 [32]	S23L23 [32]	IRNN [9]	APGL [34]	$\ \cdot\ _*$ [3]	\mathcal{R}_μ [21]	Our
0	0.0400	0.0246	0.0406	0.0501	0.0544	0.0545	0.0551	0.0229	0.1959	0.0198	0.0199
20	0.3707	0.2990	0.3751	0.1236	0.1322	0.0972	0.0440	0.0233	0.2287	0.0257	0.0198
40	1.0000	0.6185	0.9355	0.1265	0.1222	0.1137	0.0497	0.0291	0.3183	0.2105	0.0248
60	1.0000	0.8278	1.0000	0.1354	0.1809	0.1349	0.0697	0.0826	0.5444	0.3716	0.0466
80	1.0000	0.9810	1.0000	0.7775	0.6573	0.5945	0.2305	0.4648	0.8581	0.9007	0.3117
0	0.0399	0.0220	0.0399	0.0491	0.0352	0.0344	0.0491	0.0205	0.1762	0.0176	0.0177
10	0.3155	0.2769	0.1897	0.1171	0.0881	0.0874	0.0926	0.1039	0.2607	0.0829	0.0802
20	0.4681	0.4250	0.3695	0.1893	0.1346	0.1340	0.1430	0.1686	0.3425	0.2146	0.1343
30	0.5940	0.5143	0.4147	0.1681	0.2822	0.3081	0.1316	0.1594	0.3435	0.4137	0.1277
40	0.7295	0.6362	0.9331	0.2854	0.4262	0.4089	0.1731	0.2800	0.5028	0.5072	0.1705
50	0.7977	0.7228	0.9162	0.4439	0.5646	0.5523	0.2847	0.4219	0.5831	0.6464	0.3128

additional priors. In each iteration, we solve

$$\mathbf{X}_{t+1} = \arg \min_{\mathbf{X}} f_h^{**}(\mathbf{X}) + \rho \|\mathbf{X} - \mathbf{Y}_t + \mathbf{\Lambda}_t\|_F^2, \quad (25)$$

$$\mathbf{Y}_{t+1} = \arg \min_{\mathbf{Y}} \rho \|\mathbf{X}_{t+1} - \mathbf{Y} + \mathbf{\Lambda}_t\|_F^2 + \mathcal{C}(\mathbf{Y}), \quad (26)$$

$$\mathbf{\Lambda}_{t+1} = \mathbf{X}_{t+1} - \mathbf{Y}_{t+1} + \mathbf{\Lambda}_t. \quad (27)$$

To evaluate the proximal operator f_h^{**} one must solve

$$\min_{\mathbf{X}} \mathcal{R}_h(\mathbf{X}) + \|\mathbf{X} - \mathbf{X}_0\|_F^2 + \rho \|\mathbf{X} - \mathbf{M}\|_F^2. \quad (28)$$

Note, that due to the definition of (19), this can be seen as a convex-concave min-max problem, by restricting the minimization of \mathbf{X} over a compact set. By first solving for \mathbf{X} one obtains,

$$\mathbf{X} = \mathbf{M} + \frac{\mathbf{X}_0 - \mathbf{Z}}{\rho} = \frac{(\rho + 1)\mathbf{Y} - \mathbf{Z}}{\rho}, \quad (29)$$

where $\mathbf{Y} = \frac{\mathbf{X}_0 + \rho \mathbf{M}}{1 + \rho}$. Similarly, as in [21], we get a program of the type (excluding constants)

$$\begin{aligned} \max_{\mathbf{Z}} \left(\sum_{i=1}^n \min \left(b_i, [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right) - \frac{\rho + 1}{\rho} \|\mathbf{Z} - \mathbf{Y}\|_F^2 \right. \\ \left. + \|\mathbf{Z}\|_F^2 - \sum_{i=1}^n [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right). \end{aligned} \quad (30)$$

Again, the optimization can be reduced to the singular values only. This bears strong resemblance to (21), and we show in the supplementary material that Algorithm 1 can be modified, with minimal effort, to solve this problem as well.

6 Experiments

We demonstrate the shortcomings of using WNNM for non-rigid reconstruction estimation and structure-from-motion, and show that our proposed method performs as good or better than the current state-of-the-art. In all applications, we apply the popular approach [8, 15, 17] to choose the weights inversely proportional to the singular values,

$$w_i = \frac{C}{\sigma_i(\mathbf{X}_0) + \varepsilon}, \quad (31)$$

where $\varepsilon > 0$ is a small number (to avoid division by zero), and \mathbf{X}_0 is an initial estimate of the matrix \mathbf{X} . The trade-off parameter C will be tuned to the specific application. In the experiments, we use $w_i = 2a_i$, and choose b_i depending on the specific application. This allows us to control the rank of the obtained solution without excessive penalization of the non-zero singular values.

6.1 Synthetic Missing Data

In this section we consider the missing data problem with unknown rank

$$\min_{\mathbf{X}} \mu \text{rank}(\mathbf{X}) + \|\mathbf{W} \odot (\mathbf{X} - \mathbf{M})\|_F^2, \quad (32)$$

where \mathbf{M} is a measurement matrix, \odot denotes the Hadamard (or element-wise) product, and \mathbf{W} is a missing data mask, with $w_{ij} = 1$ if the entry (i, j) is known, and zero otherwise.

Ground truth matrices \mathbf{M}_0 of size 32×512 with $\text{rank}(\mathbf{M}_0) = 4$ are generated, and to simulate noise, a matrix \mathbf{N} is added to obtain the measurement matrix $\mathbf{M} = \mathbf{M}_0 + \mathbf{N}$. The entries of the noise matrix are normally distributed with zero mean and standard deviation $\sigma = 0.1$.

When benchmarking image inpainting and deblurring, it is common to assume a uniformly distributed missing data pattern. This assumption, however, is not applicable in many other subfields of computer vision. In structure-from-motion the missing data pattern is typically very structured, due to tracking failures. For comparison we show the reconstruction results for several methods, on both uniformly random missing data patterns and tracking failures. The tracking failure patterns were generated as in [20]. The results are shown in Table 1. Here we use the $a_i = \frac{\sqrt{\mu}}{\sigma_i(\mathbf{M}) + \varepsilon}$, and $b_i = \frac{\mu}{\sigma_i(\mathbf{M}) + \varepsilon}$, with $\varepsilon = 10^{-6}$. All other parameters are set as proposed by the respective authors.

6.2 Non-Rigid Deformation with Missing Data

This experiment is constructed to highlight the downsides of using WNNM, and to illustrate how shrinking bias can manifest itself in a real-world application. Non-rigid deformations can be seen as a low-rank minimizing problem by assuming that the tracked image points are moving in a low-dimensional subspace. This allows us to model the points using a linear shape basis, where the complexity of the motion is limited by the number of basis elements. This in turn, leads to the task of accurately making trade-offs while enforcing a low (and unknown) rank, which leads to the problem formulation

$$\min_{\mathbf{X}} \mu \text{rank}(\mathbf{X}) + \|\mathbf{W} \odot (\mathbf{X} - \mathbf{M})\|, \quad (33)$$

where $\mathbf{X} = \mathbf{C}\mathbf{B}^T$, with \mathbf{B} being concatenated basis elements and \mathbf{C} the corresponding coefficient matrix. We use the experimental setup from [19], where a KLT tracker is used on a video sequence. The usage of the tracker naturally induces a structured missing data pattern, due to the inability to track the points through the entire sequence.

We consider the relaxation of (33)

$$\min_{\mathbf{X}} \mathcal{R}_h(\mathbf{X}) + \|\mathbf{W} \odot (\mathbf{X} - \mathbf{M})\|_F^2, \quad (34)$$

and choose $a_i = \frac{C}{\sigma_i(\mathbf{M}) + \epsilon}$ and $b_i = 0$ for $i \leq 3$ and $b_i = 1/(C + \epsilon)$ otherwise. This choice of \mathbf{b} encourages a rank 3 solution without penalizing the large singular values. By choosing the parameter C , one may vary the strength of the fixed-rank regularization versus the weighted nuclear norm penalty. The datafit vs the parameter C is shown in Table 2, and the reconstructed points for four frames of the *book* sequence are shown in Figure 3.

Notice that, despite the superior datafit for $C = 1$ (encouraging the WNNM penalty), it is clear by visual inspection that the missing points are suboptimally recovered. In Figure 3 the white center marker is the origin, and we note a tendency for the WNNM penalty to favor solutions where the missing points are closer to the origin. This is the consequence of a shrinking bias, and is only remedied by leaving the larger singular values intact, thus excluding WNNM as a viable option for such applications.

Table 2: Datafit for different values of C . Note that the datafit for $C = 1$ is better than for $C = 10^{-2}$. This comes at the cost of incorrectly reconstructing the missing points, as is shown in Figure 3. The datafit is measured as $\|\mathbf{W} \odot (\mathbf{X} - \mathbf{M})\|_F$.

C	10^{-2}	1	100
Datafit	0.8354	0.4485	6.5221

6.3 Motion Capture

The popular prior-free objective, proposed by Dai *et al.* [12], for NRSfM

$$\min_{\mathbf{X}} \mu \|\mathbf{X}^\sharp\|_* + \|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F^2, \quad (35)$$

where \mathbf{X}^\sharp a stacked version of \mathbf{X} (see [12] for details), suffers from shrinking bias, due to the nuclear norm penalty. Essentially, the nuclear norm penalty is a way of relaxing the soft rank penalty,

$$\min_{\mathbf{X}} \mu \text{rank}(\mathbf{X}^\sharp) + \|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F^2, \quad (36)$$

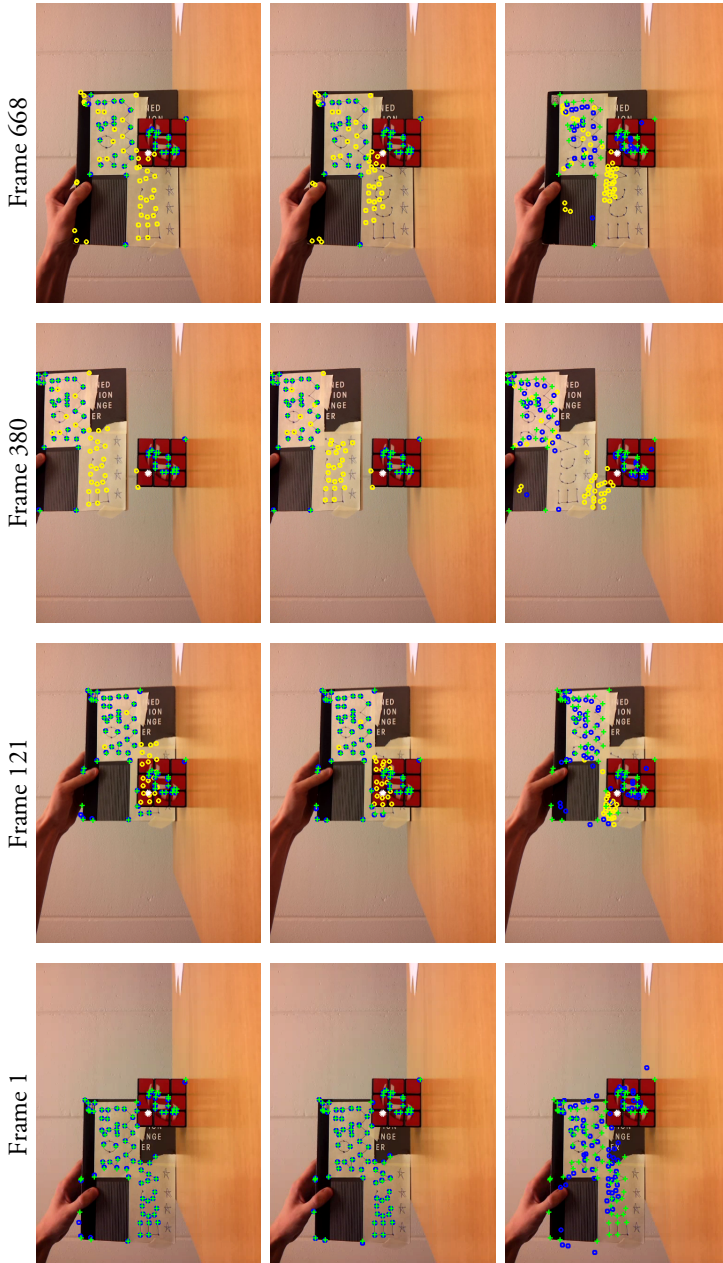


Figure 3: From top to bottom, $C = 10^{-2}$, $C = 1$ and $C = 100$. The white center dot is the origin in the chosen coordinate system. The green crosses show the observed data, and the blue dots the reconstruction of these points. The yellow dots correspond to the recovered (and missing) data. Notice the shrinking bias which is evident due to the recovered missing data being drawn towards the center of the image as the WNNM penalty increases.

however, it was shown in [27], that simply using the convex envelope of the rank function leads to non-physical reconstructions. To tackle this situation, it was proposed to penalize the 3D trajectories using a difference operator \mathbf{D} ,

$$\min_{\mathbf{X}} \mu \text{rank}(\mathbf{X}^\sharp) + \|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F^2 + \|\mathbf{D}\mathbf{X}^\sharp\|_F^2. \quad (37)$$

While such an objective leads to more physical solutions [27], it also restricts the method to ordered sequences of images. To allow for unordered sequences, we replace the difference operator with an increasing penalty for smaller singular values, modelled by an increasing sequence of weights $\{a_i\}$. More specifically, we consider the problem of minimizing

$$\min_{\mathbf{X}} \mathcal{R}_h(\mathbf{X}^\sharp) + \|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F^2, \quad (38)$$

where sequences $\{a_i\}$ and $\{b_i\}$ are non-decreasing. This bears resemblance to the weighted nuclear norm approach presented in [17], recently, which coincide for the special case $b_i \equiv 0$. Furthermore, this modified approach exhibits far superior reconstruction results compared to the original method proposed by Dai *et al.* [12]. In our comparison, we employ the same initialization heuristic for the weights w_i on the singular values as in [15, 17], namely

$$w_i = \frac{C}{\sigma_i(\mathbf{X}_0^\sharp) + \varepsilon}, \quad (39)$$

where $\varepsilon = 10^{-6}$ and $C > 0$. The matrix $\mathbf{X}_0^\sharp = \mathbf{R}^+ \mathbf{M}$, where \mathbf{R}^+ is the pseudo-inverse of \mathbf{R} , has successfully been used as an initialization scheme for NRSfM by others [12, 17, 35].

In practice, we choose $2a_i = w_i$, as in (39), with $C = 2\sqrt{\mu}$ and $b_i = w_i$, with $C = \mu$. This enforces mixed a soft-rank and hard rank thresholding.

We select four sequences from the CMU MOCAP dataset, and compare to the original method proposed by Dai *et al.* [12], the newly proposed weighted approach by Kumar [17], the method by Larsson and Olsson [21] and our proposed objective (38), all of which are prior-free, and do not assume that the image sequences are ordered. For the nuclear norm approach by Dai *et al.* we use the regularization parameter $\lambda = 2\sqrt{\mu}$, and for Kumar, we set $C = 2\sqrt{\mu}$ (as for \mathcal{R}_h) and run the different methods for a wide range of values for μ , using the same random initial solutions. We then measure the datafit, defined as $\|\mathbf{R}\mathbf{X} - \mathbf{M}\|_F$ and the distance to ground truth $\|\mathbf{X} - \mathbf{X}_{\text{gt}}\|_F$, and show how these depend on the output rank (here defined as the number of singular values larger than 10^{-6}). By doing so, we see the ability of the method to make accurate trade-offs between fitting the data and enforcing the rank. The results are shown in Figure 4.

Note that, the datafit for all methods decrease as the rank increases, which is to be expected; however, we immediately note that the “soft rank” penalty (3), in this case, is too weak. This manifests itself by mostly fitting to data, and the distance to ground truth does not correlate with the datafit for solutions with rank larger than three. For the revised method by Kumar [17], as well as ours, the correlation between the two quantities is much stronger. What is interesting to see is that our method consistently performs better than the WNNM approach for lower rank levels, suggesting that the shrinking bias is affecting the quality of these reconstruction. Note, however, that the minimum distance to ground truth, obtained using the WNNM approach is as good (or better) than the one obtained using \mathcal{R}_h . To obtain such a solution, however, requires careful tuning of the μ parameter and is unlikely to work on other datasets.

7 Conclusions

Despite success in many low-level imaging applications, there are limitations of the applicability of WNNM in other applications of low-rank regularization. In this paper, we have provided theoretical insight into the issues surrounding shrinking bias, and proposed a solution where the shrinking bias can be partly or completely eliminated, while keeping the rank low. This can be done using the proposed \mathcal{R}_h regularizer, which has the benefit of unifying weighted nuclear norm regularization with another class of low-rank inducing penalties. Furthermore, an efficient way of computing the regularizer has been proposed, as well as the related proximal operator, which makes it suitable for optimization using splitting scheme, such as ADMM.

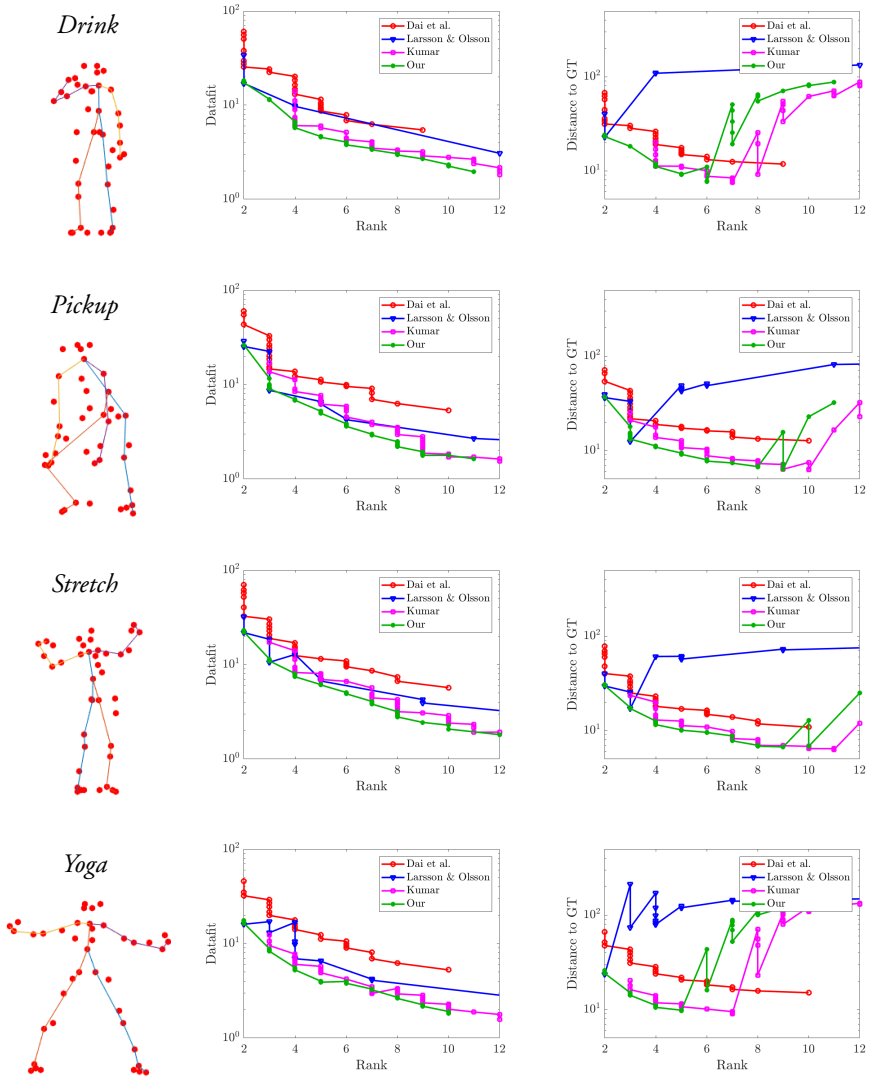


Figure 4: Results for the experiment on the CMU MOCAP dataset. *First column*: Example images with skeleton added for visualization. *Second column*: The datafit, measured as $\|\mathbf{RX} - \mathbf{M}\|_F$, as a function of the rank. *Last column*: Distance to ground truth, measured as $\|X - X_{\text{gt}}\|_F$, as a function of the rank.

References

- [1] R. Angst, C. Zach, and M. Pollefeys. “The generalized trace-norm and its application to structure-from-motion problems”. In: *International Conference on Computer Vision*. 2011.
- [2] R. Basri, D. Jacobs, and I. Kemelmacher. “Photometric Stereo with General, Unknown Lighting”. In: *International Journal of Computer Vision* 72.3 (May 2007), pp. 239–257. ISSN: 0920-5691.
- [3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Found. Trends Mach. Learn.* 3.1 (2011), pp. 1–122.
- [4] C. Bregler, A. Hertzmann, and H. Biermann. “Recovering non-rigid 3D shape from image streams”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2000.
- [5] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [6] E. J. Candès, X. Li, Y. Ma, and J. Wright. “Robust Principal Component Analysis?” In: *J. ACM* 58.3 (2011), 11:1–11:37.
- [7] E. J. Candès, X. Li, Y. Ma, and J. Wright. “Robust principal component analysis?” In: *Journal of the ACM (JACM)* 58.3 (2011), p. 11.
- [8] E. J. Candès, M. B. Wakin, and S. P. Boyd. “Enhancing sparsity by reweighted l^1 minimization”. In: *Journal of Fourier analysis and applications* 14.5-6 (2008), pp. 877–905.
- [9] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Nonconvex Nonsmooth Low-Rank Minimization via Iteratively Reweighted Nuclear Norm”. In: *IEEE Transactions on Image Processing* 25 (Oct. 2015).
- [10] M. Carlsson. “On convexification/optimization of functionals including an l_2 -misfit term”. In: *arXiv preprint arXiv:1609.09378* (2016).
- [11] M. Carlsson, D. Gerosa, and C. Olsson. “An unbiased approach to compressed sensing”. In: *arXiv preprint arXiv:1806.05283* (2018).
- [12] Y. Dai, H. Li, and M. He. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization”. In: *International Journal of Computer Vision* 107.2 (2014), pp. 101–122.

-
- [13] R. Garg, A. Roussos, and L. Agapito. “A Variational Approach to Video Registration with Subspace Constraints”. In: *International Journal of Computer Vision* 104.3 (2013), pp. 286–314.
- [14] N. Gillis and F. Glinier. “Low-Rank Matrix Approximation with Weights or Missing Data is NP-hard”. In: *SIAM Journal on Matrix Analysis and Applications* 32.4 (2011), pp. 1149–1165.
- [15] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang. “Weighted Nuclear Norm Minimization and Its Applications to Low Level Vision”. In: *International Journal of Computer Vision* 121 (July 2016), pp. 183–208.
- [16] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. “Fast and Accurate Matrix Completion via Truncated Nuclear Norm Regularization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2117–2130.
- [17] S. Kumar. “Non-Rigid Structure from Motion: Prior-Free Factorization Method Revisited”. In: *The IEEE Winter Conference on Applications of Computer Vision (WACV)*. Mar. 2020.
- [18] S. Kumar, Y. Dai, and H. Li. “Superpixel Soup: Monocular Dense 3D Reconstruction of a Complex Dynamic Scene”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Nov. 2019).
- [19] V. Larsson, E. Bylow, C. Olsson, and F. Kahl. “Rank Minimization with Structured Data Patterns”. In: *European Conference on Computer Vision*. 2014.
- [20] V. Larsson and C. Olsson. “Compact Matrix Factorization with Dependent Subspaces”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017, pp. 4361–4370.
- [21] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [22] L. Mirsky. “A Trace Inequality of John von Neumann.” In: *Monatshefte für Mathematik* 79 (1975), pp. 303–306.
- [23] K. Mohan and M. Fazel. “Iterative reweighted least squares for matrix rank minimization”. In: *Annual Allerton Conference on Communication, Control, and Computing*. 2010, pp. 653–661.
- [24] R. A. Newcombe, D. Fox, and S. M. Seitz. “DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 343–352.
- [25] F. Nie, H. Wang, X. Cai, H. Huang, and C. H. Q. Ding. “Robust Matrix Completion via Joint Schatten p -Norm and l_p -Norm Minimization”. In: *International Conference on Data Mining (ICDM)*. 2012, pp. 566–574.

- [26] T. H. Oh, Y. W. Tai, J. C. Bazin, H. Kim, and I. S. Kweon. “Partial Sum Minimization of Singular Values in Robust PCA: Algorithm and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4 (2016), pp. 744–758.
- [27] C. Olsson, M. Carlsson, F. Andersson, and V. Larsson. “Non-Convex Rank/Sparsity Regularization and Local Minima”. In: *Proceedings of the International Conference on Computer Vision* (2017).
- [28] C. Olsson, M. Carlsson, and D. Gerosa. “Bias Reduction in Compressed Sensing”. In: *arXiv preprint arxiv:1812.11329* (2018).
- [29] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi. “Simultaneously Structured Models With Application to Sparse and Low-Rank Matrices”. In: *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2886–2908.
- [30] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *SIAM Rev.* 52.3 (Aug. 2010), pp. 471–501.
- [31] C. Russell, R. Yu, and L. Agapito. “Video-Popup: Monocular 3D Reconstruction of Dynamic Scenes”. In: *European Conference on Computer Vision (ECCV)*. 2014.
- [32] F. Shang, J. Cheng, Y. Liu, Z. Luo, and Z. Lin. “Bilinear Factor Matrix Norm Minimization for Robust PCA: Algorithms and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.9 (Sept. 2018), pp. 2066–2080.
- [33] J. Taylor, A. Jepson, and K. Kutulakos. “Non-Rigid Structure from Locally-Rigid Motion”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2010, pp. 2761–2768.
- [34] K.-C. Toh and S. Yun. “An Accelerated Proximal Gradient Algorithm for Nuclear Norm Regularized Least Squares Problems”. In: *Pacific Journal of Optimization* 6 (Sept. 2010).
- [35] J. Valmadre, S. Sridharan, S. Denman, C. Fookes, and S. Lucey. “Closed-Form Solutions for Low-Rank Non-Rigid Reconstruction”. In: *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. Nov. 2015, pp. 1–6.
- [36] J. Xu, L. Zhang, D. Zhang, and X. Feng. “Multi-channel Weighted Nuclear Norm Minimization for Real Color Image Denoising”. In: *International Conference on Computer Vision (ICCV)* (2017).
- [37] N. Yair and T. Michaeli. “Multi-Scale Weighted Nuclear Norm Image Restoration”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

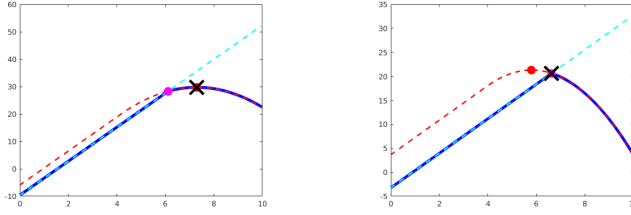


Figure 5: Two different cases for values of a_i , b_i and $\sigma_i(\mathbf{X})$ of (21).

Supplementary Material

A Von Neumann's Trace Theorem

We use von Neumann's trace theorem repeatedly in the main paper, hence we state it here for completeness, using the inner products $\langle \mathbf{X}, \mathbf{Y} \rangle = \text{tr}(\overline{\mathbf{X}}^T \mathbf{Y})$, and $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}$.

Theorem 2 (Von Neumann [3]). *Let $\mathbf{X}, \mathbf{Y} \in \mathbb{C}^{n \times n}$ and $\sigma(\mathbf{X})$ be the singular value vector of \mathbf{X} . Then*

$$\langle \mathbf{X}, \mathbf{Y} \rangle \leq \langle \sigma(\mathbf{X}), \sigma(\mathbf{Y}) \rangle,$$

with equality if and only if \mathbf{X} and \mathbf{Y} are simultaneously unitarily diagonalizable.

Consider maximization over \mathbf{Z} in (19) and note that

$$-\|\mathbf{X} - \mathbf{Z}\|_F^2 = -\|\mathbf{X}\|_F^2 - \|\mathbf{Z}\|_F^2 + 2\langle \mathbf{X}, \mathbf{Z} \rangle, \quad (40)$$

and by Theorem 2, $\langle \mathbf{X}, \mathbf{Z} \rangle \leq \langle \sigma(\mathbf{X}), \sigma(\mathbf{Z}) \rangle$, with equality if \mathbf{X} and \mathbf{Z} are simultaneously unitarily diagonalizable. Note that the Frobenius norm is unitarily invariant, with $\|\mathbf{X}\|_F^2 = \sum_i \sigma_i(\mathbf{X})^2$. Therefore

$$-\|\mathbf{X} - \mathbf{Z}\|_F^2 \leq -\sum_i (\sigma_i(\mathbf{X}) - \sigma_i(\mathbf{Z}))^2, \quad (41)$$

with equality if \mathbf{X} and \mathbf{Z} are simultaneously unitarily diagonalizable, i.e. $\mathbf{X} = \mathbf{U} \mathbf{D}_{\sigma(\mathbf{X})} \mathbf{V}^T$ and $\mathbf{Z} = \mathbf{U} \mathbf{D}_{\sigma(\mathbf{Z})} \mathbf{V}^T$, where \mathbf{D}_x is a diagonal matrix with x on the main diagonal.

The remaining terms of (19) only depend on the singular values of \mathbf{Z} and therefore the maximum occurs when we select \mathbf{Z} so that we have equality in (41). This establishes the equality between (19) and (20) of the main paper.

B The Fenchel Conjugate

In this section, we compute the Fenchel conjugate of (12), which is necessary in order to find the convex envelope. Let $\langle \mathbf{X}, \mathbf{Y} \rangle = \text{tr}(\mathbf{X}^T \mathbf{Y})$, and note that we can write

$$\langle \mathbf{Y}, \mathbf{X} \rangle - \|\mathbf{X} - \mathbf{X}_0\|_F = \|\mathbf{Z}\|_F^2 - \|\mathbf{X}_0\|_F^2 - \|\mathbf{Z} - \mathbf{X}\|_F^2, \quad (42)$$

where $\mathbf{Z} = \frac{1}{2}\mathbf{Y} + \mathbf{X}_0$. By definition, the Fenchel conjugate of (12) is given by

$$f_h^*(\mathbf{Y}) = \sup_{\mathbf{X}} \langle \mathbf{Y}, \mathbf{X} \rangle - f_h(\mathbf{X}) = \sup_{\mathbf{X}} \|\mathbf{Z}\|_F^2 - \|\mathbf{X}_0\|_F^2 - \|\mathbf{X} - \mathbf{Z}\|_F^2 - h(\sigma(\mathbf{X})), \quad (43)$$

where we use (42) in the last step. Note that the function h , as well as the Frobenius norm, is unitarily invariant. Furthermore, $\|\mathbf{X} - \mathbf{Z}\|_F^2 = \|\mathbf{X}\|_F^2 + \|\mathbf{Z}\|_F^2 - 2\langle \mathbf{X}, \mathbf{Z} \rangle$, and $\langle \mathbf{X}, \mathbf{Z} \rangle \leq \langle \sigma(\mathbf{X}), \sigma(\mathbf{Z}) \rangle$ by von Neumann's trace inequality, with equality if \mathbf{X} and \mathbf{Z} are simultaneously unitarily diagonalizable. This reduces the problem to optimizing over the singular values alone, which, after some manipulation, can be written as

$$f_h^*(\mathbf{Y}) = \max_{\sigma(\mathbf{X})} -\|\mathbf{X}_0\|_F^2 - \sum_{i=1}^k \left(\sigma_i^2(\mathbf{X}) - 2[\sigma_i(\mathbf{Z}) - a_i]\sigma_i(\mathbf{X}) + b_i \right), \quad (44)$$

where $\text{rank}(\mathbf{X}) = k$. Considering each singular value separately leads to a program on the form

$$\min_{x_i} x_i^2 - 2[\sigma_i(\mathbf{Z}) - a_i]x_i + b_i, \quad (45)$$

subject to $\sigma_{i+1}(\mathbf{X}) \leq x_i \leq \sigma_{i-1}(\mathbf{X})$. The sequence of unconstrained minimizers is given by $x_i = \sigma_i(\mathbf{Z}) - a_i$. If there exists $x_i < 0$, then this is not the solution to the constrained problem. Nevertheless, the sequence is non-increasing, hence there is an index p , such that $x_p \geq 0$ and $x_{p+1} < 0^4$.

Note that

$$\sum_{i=1}^k x_i^2 - 2s_i x_i = \|\mathbf{x}\|^2 - 2\langle \mathbf{x}, \mathbf{s} \rangle, \quad (46)$$

hence we can consider optimizing $\|\mathbf{x} - \mathbf{s}\|^2 = \|\mathbf{x}\|^2 - 2\langle \mathbf{x}, \mathbf{s} \rangle + \|\mathbf{s}\|^2$ subject to $x_1 \geq x_2 \geq \dots \geq x_k \geq 0$. Furthermore, $s_1 \geq s_2 \geq \dots \geq s_k$.

Assume that minimum is obtained at \mathbf{x}^* and fix x_p^* . Since $s_j < 0$ for all $j > p$, we must have $x_j^* = 0$ for $j > p$. It is now clear that, $x_j^* = s_j$ otherwise, hence $x_j^* = \max\{s_j, 0\} = [s_j]_+$. Inserting into (44) gives

$$f_h^*(\mathbf{Y}) = \max_k -\|\mathbf{X}_0\|_F^2 - \sum_{i=1}^k \left(b_i - [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right). \quad (47)$$

⁴We allow the case $p = 0$, in which case the zero vector is optimal.

Since $[s_i]_+ = [\sigma_i(\mathbf{Z}) - a_i]_+$ is non-increasing, and b_i is non-decreasing, the maximizing k is obtained when

$$[\sigma_k(\mathbf{Z}) - a_k]_+^2 \geq b_k \text{ and } b_{k+1} \geq [\sigma_{k+1}(\mathbf{Z}) - a_{k+1}]_+^2. \quad (48)$$

For the maximizing $k = k^*$, we can write

$$\begin{aligned} & - \sum_{i=1}^{k^*} \left(b_i - [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right) \\ & = \sum_{i=1}^n [\sigma_i(\mathbf{Z}) - a_i]_+^2 - \sum_{i=1}^n \min\{b_i, [\sigma_i(\mathbf{Z}) - a_i]_+^2\}. \end{aligned} \quad (49)$$

From this observation, we get

$$\begin{aligned} f_h^*(\mathbf{Y}) & = \sum_{i=1}^n \left[\sigma_i\left(\frac{1}{2}\mathbf{Y} + \mathbf{X}_0\right) - a_i \right]_+^2 - \|\mathbf{X}_0\|_F^2 \\ & \quad - \sum_{i=1}^n \min\left(b_i, \left[\sigma_i\left(\frac{1}{2}\mathbf{Y} + \mathbf{X}_0\right) - a_i \right]_+^2 \right). \end{aligned} \quad (50)$$

C The Convex Envelope

Applying the definition of the bi-conjugate $f_h^{**}(\mathbf{X}) = \sup_{\mathbf{Y}} \langle \mathbf{Y}, \mathbf{X} \rangle - f_h^*(\mathbf{Y})$ to (50), and introduce the change of variables $\mathbf{Z} = \frac{1}{2}\mathbf{Y} + \mathbf{X}_0$ we get

$$\begin{aligned} f_h^{**}(\mathbf{X}) & = \max_{\mathbf{Z}} 2\langle \mathbf{X}, \mathbf{Z} - \mathbf{X}_0 \rangle - \sum_{i=1}^n [\sigma_i(\mathbf{Z}) - a_i]_+^2 \\ & \quad + \|\mathbf{X}_0\|_F^2 + \sum_{i=1}^n \min\left(b_i, [\sigma_i(\mathbf{Z}) - a_i]_+^2 \right). \end{aligned} \quad (51)$$

By expanding squares and simplifying, $2\langle \mathbf{X}, \mathbf{Z} - \mathbf{X}_0 \rangle + \|\mathbf{X}_0\|_F^2 = \|\mathbf{X} - \mathbf{X}_0\|_F^2 - \|\mathbf{X} - \mathbf{Z}\|_F^2 + \|\mathbf{Z}\|_F^2$, which yields (19).

D Obtaining the Maximizing Sequences

In this section we give the proof for the convergence of Algorithm 1, and how to modify it to cope with the corresponding problem for the proximal operator.

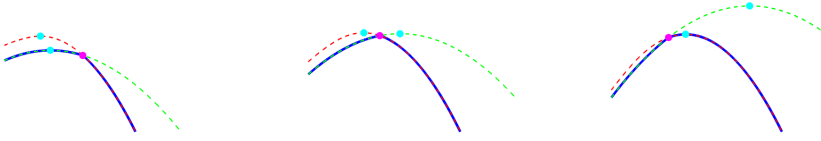


Figure 6: Illustration of the three different cases for the proximal operator.

E Proof of Theorem 1

Proof of Theorem 1. First, we will show that each step in the algorithm returns a solution to a constrained subproblem P_i , corresponding to a (partial) set of desired constraints \mathcal{Z}_i .

Let P_0 denote the unconstrained problem with solution $f \in \mathbb{R}_+^n$. Denote the first interval generated in Algorithm 1 by $\iota_1 = \{m_1, \dots, n_1\}$, and consider optimizing the first subproblem P_1

$$\max_{z_{m_1} \geq \dots \geq z_{n_1}} c(\mathbf{z}), \tag{52}$$

where $\mathcal{Z}_1 = \{\mathbf{z} \in \mathcal{Z}_0 \mid z_{m_1} \geq \dots \geq z_{n_1}\}$. By Lemma 1 the solution vector is constant over the subinterval $z_i = s$ for $i \in \iota_1$, which is returned by the algorithm. The next step generates a solution to subproblem of the form

$$\begin{aligned} \max_{z_{m_1} \geq \dots \geq z_{n_1}} c(\mathbf{z}), \\ \vdots \\ z_{m_k} \geq \dots \geq z_{n_k} \end{aligned} \tag{53}$$

corresponding to subproblem P_k . If the solution to subproblem P_k is in \mathcal{Z} , then it is a solution to the problem, otherwise one must add more constraints. We solve problems on the form

$$\max_{z \in \mathcal{Z}_0} c(\mathbf{z}) \geq \max_{z \in \mathcal{Z}_1} c(\mathbf{z}) \geq \dots \geq \max_{z \in \mathcal{Z}_\ell} c(\mathbf{z}) = \max_{z \in \mathcal{Z}} c(\mathbf{z}), \tag{54}$$

where the last step yields a solution fulfilling the desired constraints. Furthermore $\mathcal{Z}_0 \supset \mathcal{Z}_1 \supset \dots \supset \mathcal{Z}_\ell \supset \mathcal{Z}$, where $\mathcal{Z} = \{\mathbf{z} \mid z_1 \geq \dots \geq z_n \geq 0\}$. Finally, it is easy to see that the algorithm terminates, since there are only finitely many possible subintervals. \square

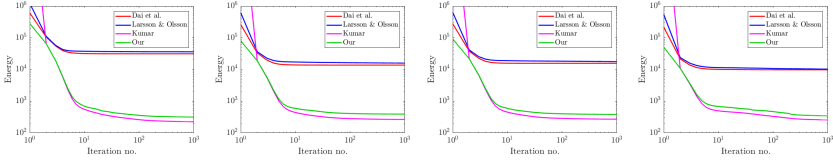


Figure 7: Convergence for the different methods compared in Section 6.3. NB: The energies are different, and have been averaged over 35 different values of μ (the same values as in Figure 4).

F Modifying Algorithm 1

Following the approach used in [2], consider the program

$$\begin{aligned}
 \max_s \quad & \min\{b_i, [s - a_i]_+^2\} - \frac{\rho + 1}{\rho} (s - \sigma_i(\mathbf{Y}))^2 \\
 & + s^2 - [s - a_i]_+^2, \\
 \text{s.t.} \quad & \sigma_{i+1}(\mathbf{Z}) \leq s \leq \sigma_{i-1}(\mathbf{Z}).
 \end{aligned} \tag{55}$$

Note that the objective function is the pointwise minimum of

$$\begin{aligned}
 f_1(s) &= b_i - \frac{\rho + 1}{\rho} (s - \sigma_i(\mathbf{Y}))^2 + s^2 - [s - a_i]_+^2, \\
 f_2(s) &= s^2 - \frac{\rho + 1}{\rho} (s - \sigma_i(\mathbf{Y}))^2,
 \end{aligned} \tag{56}$$

both of which are concave, since $\frac{\rho+1}{\rho} > 1$. For f_1 the maximum is obtained in $s = \frac{a_i \rho}{\rho+1} + \sigma_i(\mathbf{Y})$, if $s \geq a_i$ otherwise when $s = (\rho + 1)\sigma_i(\mathbf{Y})$. The minimum of f_2 is obtained when $s = (\rho + 1)\sigma_i(\mathbf{Y})$.

There are three possible cases, also shown in Figure 6.

1. The maximum occurs when $s > a_i + \sqrt{b_i}$, hence $f_1(s) < f_2(s)$, hence $s = \frac{a_i \rho}{\rho+1} + \sigma_i(\mathbf{Y})$.
2. The maximum occurs when $s < a_i + \sqrt{b_i}$, where $f_1(s) > f_2(s)$, hence $s = (\rho + 1)\sigma_i(\mathbf{Y})$.
3. When $s = a_i + \sqrt{b_i}$, which is valid elsewhere.

in summary

$$s_i = \begin{cases} \frac{a_i \rho}{\rho + 1} + \sigma_i(\mathbf{Y}), & \frac{a_i}{\rho + 1} + \sqrt{b_i} < \sigma_i(\mathbf{Y}), \\ a_i + \sqrt{b_i}, & \frac{a_i + \sqrt{b_i}}{1 + \rho} \leq \sigma_i(\mathbf{Y}) \leq \frac{a_i}{\rho + 1} + \sqrt{b_i}, \\ (1 + \rho)\sigma_i(\mathbf{Y}), & \sigma_i(\mathbf{Y}) < \frac{a_i + \sqrt{b_i}}{1 + \rho}, \end{cases} \quad (57)$$

By replacing the sequence of unconstrained minimizers $\{s_i\}$ defined by (57), with the corresponding sequence in Section 4 (of the main paper), and changing the objective function of Algorithm 1, to the one in (55), the maximizing singular value vector for the proximal operator is obtained.

G Convergence: Motion Capture

In this section we compare the convergence of the different regularizers used in Section 6.3, see Figure 7. Note that the energies are different, and one can only compare the number of steps needed until convergence. For this particular choice of a_i and b_i the \mathcal{R}_h regularizer behaves much like WNNM used in [1].

References

- [1] S. Kumar. “Non-Rigid Structure from Motion: Prior-Free Factorization Method Revisited”. In: *The IEEE Winter Conference on Applications of Computer Vision (WACV)*. Mar. 2020.
- [2] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [3] L. Mirsky. “A Trace Inequality of John von Neumann.” In: *Monatshefte für Mathematik* 79 (1975), pp. 303–306.

Accurate Optimization of Weighted Nuclear Norm for Non-Rigid Structure from Motion

JOSÉ PEDRO IGLESÍAS¹, CARL OLSSON^{1,2} AND MARCUS VALTONEN ÖRNHAG²
¹*Department of Electrical Engineering, Chalmers University of Technology*
²*Centre for Mathematical Sciences, Lund University*

Abstract: Fitting a matrix of a given rank to data in a least squares sense can be done very effectively using 2nd order methods such as Levenberg–Marquardt by explicitly optimizing over a bilinear parameterization of the matrix. In contrast, when applying more general singular value penalties, such as weighted nuclear norm priors, direct optimization over the elements of the matrix is typically used. Due to non-differentiability of the resulting objective function, first order sub-gradient or splitting methods are predominantly used. While these offer rapid iterations it is well known that they become inefficient near the minimum due to zig-zagging and in practice one is therefore often forced to settle for an approximate solution.

In this paper we show that more accurate results can in many cases be achieved with 2nd order methods. Our main result shows how to construct bilinear formulations, for a general class of regularizers including weighted nuclear norm penalties, that are provably equivalent to the original problems. With these formulations the regularizing function becomes twice differentiable and 2nd order methods can be applied. We show experimentally, on a number of structure from motion problems, that our approach outperforms state-of-the-art methods.¹

1 Introduction

Matrix recovery problems of the form

$$\min_{\mathbf{X}} f(\sigma(\mathbf{X})) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2, \quad (1)$$

where \mathcal{A} is a linear operator and $\sigma(\mathbf{X}) = (\sigma_1(\mathbf{X}), \sigma_2(\mathbf{X}), \dots)$ are the singular values of \mathbf{X} , are frequently occurring in computer vision. Applications range from high level 3D reconstruction problems to low level pixel manipulations [3, 6, 11, 16–18, 24, 32,

¹This work was supported by the Swedish Research Council (grants no. 2015-05639, 2016-04445 and 2018-05375), the Swedish Foundation for Strategic Research (Semantic Mapping and Visual Navigation for Smart Robots) and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

40, 42, 46]. In structure from motion (SfM) the most common approaches enforce a given low rank r without additionally penalizing non-zero singular values [6, 20, 40] (a special case of (1) by letting f assign zero if fewer than r singular values are non-zero and infinity otherwise).

Since the rank of a matrix \mathbf{X} is bounded by to the number of columns/rows in a bilinear parameterization $\mathbf{X} = \mathbf{B}\mathbf{C}^T$, the resulting optimization problem can be written $\min_{\mathbf{B}, \mathbf{C}} \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2$. This gives a smooth objective function and can therefore be optimized using 2nd order methods. In SfM problems, where the main interest is the extraction of camera matrices from \mathbf{B} and 3D points from \mathbf{C} , this is typically the preferred option [7]. In a series of recent papers Hong *et al.* showed that optimization with the VarPro algorithm is remarkably robust to local minima converging to accurate solutions [20, 22, 23]. In [21] they further showed how uncalibrated rigid SfM with a proper perspective projection can be solved within a factorization framework. On the downside, typically the iterations are costly since (even when the Schur complement trick is used) 2nd order methods require an inversion of a relatively large hessian matrix, which may hinder application when suitable sparsity patterns are not present.

For low level vision problems such as denoising and inpainting, eg. [18, 24, 32], the main interest is to recover the elements of \mathbf{X} and not the factorization. In this context more general regularization terms that also consider the size of the singular values are often used. Since the singular values are non-differentiable functions of the elements in \mathbf{X} first order methods are usually employed. The simplest option is perhaps a splitting methods such as ADMM [5] since the proximal operator $\arg \min_{\mathbf{X}} f(\sigma(\mathbf{X})) + \|\mathbf{X} - \mathbf{X}_0\|^2$, can often be computed in closed form [13, 18, 24, 27, 32]. Alternatively, subgradient methods can be used to handle the non-differentiability of the regularization term [11].

It is well known that while first order methods have rapid iterations and make large improvements the first couple of iterations they have a tendency to converge slowly when approaching the optimum. For example, [5] recommends to use ADMM when a solution in the vicinity of the optimal point is acceptable, but suggests to switch to a higher order method when high accuracy is desired. For low level vision problems where success is not dependent on achieving an exact factorization of a particular size, first order methods may therefore be suitable. In contrast, in the context of SfM, having roughly estimated elements in \mathbf{X} causes the obtained factorization \mathbf{B} , \mathbf{C} to be of a much larger size than necessary yielding poor reconstructions with too many deformation modes.

In this paper we aim to extend the class of methods that can be optimized using bilinear parameterization allowing accurate estimation of a low rank factorization from a general class of regularization terms. While our theory is applicable for many objectives we focus on weighted nuclear norm penalties since these have been successfully used in SfM applications. We show that these can be optimized with 2nd order methods

which significantly increases the accuracy of the reconstruction. We further show that with these improvements the model of Hong *et al.* [21] can be extended to handle non-rigid reconstruction with a proper perspective model, as opposed to the orthographic projection model adopted by other factorization based approaches, e.g. [13, 17, 27, 46].

1.1 Related Work and Contributions

Minimization directly over \mathbf{X} has been made popular since the problem is convex when f is convex and absolutely symmetric, that is, $f(|\mathbf{x}|) = f(\mathbf{x})$ and $f(\mathbf{\Pi}\mathbf{x}) = f(\mathbf{x})$, where $\mathbf{\Pi}$ is any permutation [29]. Convex penalties are however of limited interest since they generally prefer solutions with many small non-zero singular values to those with few large ones. A notable exception is the nuclear norm [9, 10, 14, 35, 36] which penalizes the sum of the singular values. Under the RIP assumption [36] exact or approximate low rank matrix recovery can then be guaranteed [10, 36]. On the other hand, since the nuclear norm penalizes large singular values, it suffers from a shrinking bias [8, 11, 28].

An alternative approach that unifies bilinear parameterization with regularization approaches is based on the observation [36] that the nuclear norm $\|\mathbf{X}\|_*$ of a matrix \mathbf{X} can be expressed as $\|\mathbf{X}\|_* = \min_{\mathbf{B}\mathbf{C}^T = \mathbf{X}} \frac{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2}{2}$. Thus when $f(\sigma(\mathbf{X})) = \mu \sum_i \sigma_i(\mathbf{X})$, where μ is a scalar controlling the strength of the regularization, optimization of (1) can be formulated as

$$\min_{\mathbf{B}, \mathbf{C}} \mu \frac{\|\mathbf{B}\|_F^2 + \|\mathbf{C}\|_F^2}{2} + \|\mathcal{A}\mathbf{B}\mathbf{C}^T - \mathbf{b}\|^2. \quad (2)$$

Optimizing directly over the factors has the advantages that the number of variables is much smaller and the objective function is two times differentiable so second order methods can be employed. While (2) is non-convex because of the bilinear terms, the convexity of the nuclear norm can still be used to show that any local minimizer \mathbf{B} , \mathbf{C} with $\text{rank}(\mathbf{B}\mathbf{C}^T) < k$, where k is the number of columns in \mathbf{B} and \mathbf{C} , is globally optimal [2, 19]. The formulation (2) was for vision problems in [8]. In practice it was observed that the shrinking bias of the nuclear norm makes it too weak to enforce a low rank when the data is noisy. Therefore, a ‘‘continuation’’ approach where the size of the factorization is gradually reduced was proposed. While this yields solutions with lower rank, the optimality guarantees no longer apply. Bach *et al.* [2] showed that

$$\|\mathbf{X}\|_{s,t} := \min_{\mathbf{X} = \mathbf{B}\mathbf{C}^T} \sum_{i=1}^k \frac{\|\mathbf{B}_i\|_s^2 + \|\mathbf{C}_i\|_t^2}{2}, \quad (3)$$

where \mathbf{B}_i , \mathbf{C}_i are the i th columns of \mathbf{B} and \mathbf{C} respectively, is convex for any choice of vector norms $\|\cdot\|_s$ and $\|\cdot\|_t$. In [19] it was shown that a more general class of 2-homogeneous factor penalties result in a convex regularization similar to (3). The

property that a local minimizer \mathbf{B}, \mathbf{C} with $\text{rank}(\mathbf{BC}^T) < k$, is global is also extended to this case. Still, because of convexity, it is clear that these formulations will suffer from a similar shrinking bias as the nuclear norm.

One way of reducing shrinking bias is to use penalties that are constant for large singular values. Shang *et al.* [38] showed that penalization with the Schatten semi-norms $\|\mathbf{X}\|_q = \sqrt[q]{\sum_{i=1}^N \sigma_i(\mathbf{X})^q}$, for $q = 1/2$ and $2/3$, can be achieved using a convex penalty on the factors \mathbf{B} and \mathbf{C} . A generalization to general values of q is given in [44]. An algorithm that address a general class of penalties for symmetric matrices is presented in [26]. In [34] it was shown that if f is given by $f(\sigma(\mathbf{X})) = \sum_i g(\sigma_i(\mathbf{X}))$, where g is differentiable, concave and non-decreasing then (1) can be optimized using 2nd order methods such as Levenberg–Marquardt or VarPro. This is achieved by re-parameterizing the matrix \mathbf{X} using a bilinear factorization $\mathbf{X} = \mathbf{BC}^T$ and optimizing

$$\min_{\mathbf{B}, \mathbf{C}} f(\gamma(\mathbf{B}, \mathbf{C})) + \|\mathcal{A}(\mathbf{BC}^T) - \mathbf{b}\|^2. \quad (4)$$

Here $\gamma(\mathbf{B}, \mathbf{C}) = (\gamma_1(\mathbf{B}, \mathbf{C}), \gamma_2(\mathbf{B}, \mathbf{C}), \dots)$ and $\gamma_i(\mathbf{B}, \mathbf{C}) = \frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2}$. In contrast to the singular value $\sigma_i(\mathbf{X})$ the function $\gamma_i(\mathbf{B}, \mathbf{C})$ is smooth which allows optimization with second order methods. It is shown in [34] that if \mathbf{X}^* is optimal in (1) then the factorization $\mathbf{B} = \mathbf{L}\sqrt{\Sigma}$, $\mathbf{C} = \mathbf{R}\sqrt{\Sigma}$, where $\mathbf{X}^* = \mathbf{L}\Sigma\mathbf{R}^T$ is the SVD of \mathbf{X}^* , is optimal in (4) (here we assume that \mathbf{L} is $m \times r$, Σ is $r \times r$ and \mathbf{R} is $n \times r$, with $\text{rank}(\mathbf{X}) = r$). Note also that this choice gives $\gamma_i(\mathbf{B}, \mathbf{C}) = \sigma_i(\mathbf{X}^*)$.

A less restrictive way of reducing bias is to re-weight the nuclear norm and use $f(\sigma(\mathbf{X})) = \sum_i a_i \sigma_i(\mathbf{X})$ [18, 24, 27]. Assigning low weights to the first (largest) singular values allows accurate matrix recovery. In addition the weights can be used to regularize the size of the non-zero singular values which has been shown to be an additional useful prior in NRSfM [27]. Note however that the singular values are always ordered in non-increasing order. Therefore, while the function is linear in the singular values it is in fact non-convex and non-differentiable in the elements of \mathbf{X} whenever the singular values are not distinct (typically the case in low rank recovery).

In this paper we show that this type of penalties allow optimization with $\gamma(\mathbf{B}, \mathbf{C})$ instead of $\sigma(\mathbf{X})$. In particular we study the optimization problem

$$\min_{\mathbf{B}, \mathbf{C}} f(\gamma(\mathbf{B}, \mathbf{C})) \quad (5)$$

$$\text{s.t.} \quad \mathbf{BC}^T = \mathbf{X}, \quad (6)$$

and its constraint set for a fixed \mathbf{X} . We characterize the extreme-points of the feasible set using permutation matrices and give conditions on f that ensure that the optimal solution is of the form $\gamma(\mathbf{B}, \mathbf{C}) = \mathbf{\Pi}\sigma(\mathbf{X})$, where $\mathbf{\Pi}$ is a permutation. For the weighted nuclear norm $f(\sigma(\mathbf{X})) = \mathbf{a}^T \sigma(\mathbf{X})$ we show that if the elements of \mathbf{a} are non-decreasing

the optimal solution has $\gamma(\mathbf{B}^*, \mathbf{C}^*) = \sigma(\mathbf{X})$. A simple consequence of this result is that

$$\min_{\mathbf{B}, \mathbf{C}} \mathbf{a}^T \gamma(\mathbf{B}, \mathbf{C}) + \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2 \quad (7)$$

is equivalent to $\min_{\mathbf{X}} \mathbf{a}^T \sigma(\mathbf{X}) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2$. While the latter is non-differentiable the former is smooth and can be minimized efficiently with second order methods.

Our experimental evaluation confirms that this approach outperforms current first order methods in terms of accuracy as can be expected. On the other hand first order methods make large improvements the first couple of iterations and therefore we combine the two approaches. We start out with a simple ADMM implementation and switch to our second order approach when only minor progress is being made. Note however that since the original formulation is non-convex local minima can exist. In addition bilinear parameterization introduces additional stationary points that are not present in the original \mathbf{X} parameterization. One such example is $(\mathbf{B}, \mathbf{C}) = (0, 0)$, where all gradients vanish. Still our experiments show that the combination of these methods often converge to a good solution from random initialization.

2 Bilinear Parameterization Penalties

In this section we will derive a dependence between the singular values $\sigma_i(\mathbf{X})$ and the $\gamma_i(\mathbf{B}, \mathbf{C})$, when $\mathbf{B}\mathbf{C}^T = \mathbf{X}$. For ease of notation we will suppress the dependence on \mathbf{X} and (\mathbf{B}, \mathbf{C}) since this will be clear from the context. Let \mathbf{X} have the SVD $\mathbf{X} = \mathbf{R}\mathbf{\Sigma}\mathbf{L}^T$, $\mathbf{B} = \mathbf{R}\sqrt{\mathbf{\Sigma}}$ and $\mathbf{C} = \mathbf{L}\sqrt{\mathbf{\Sigma}}$. We will study other potential factorizations $\mathbf{X} = \hat{\mathbf{B}}\hat{\mathbf{C}}^T$ using $\hat{\mathbf{B}} = \mathbf{B}\mathbf{V}$, $\hat{\mathbf{C}} = \mathbf{C}\mathbf{H}$ and $\mathbf{V}\mathbf{H}^T = \mathbf{I}_{r \times r}$. In this section we will further assume that \mathbf{V} is a square $r \times r$ matrix and therefore \mathbf{H}^T is its inverse. (We will generalize the results to the rectangular case in Section 3).

We begin by noting that $\gamma_j = \frac{\|\hat{\mathbf{B}}_j\|^2 + \|\hat{\mathbf{C}}_j\|^2}{2} = \frac{\|\mathbf{B}\mathbf{V}_j\|^2 + \|\mathbf{C}\mathbf{H}_j\|^2}{2}$, where \mathbf{V}_j and \mathbf{H}_j are columns j of \mathbf{V} and \mathbf{H} respectively. We have $\|\mathbf{B}\mathbf{V}_j\|^2 = \mathbf{V}_j^T \mathbf{B}^T \mathbf{B} \mathbf{V}_j = \mathbf{V}_j^T \mathbf{\Sigma} \mathbf{V}_j = \|\sqrt{\mathbf{\Sigma}}\mathbf{V}_j\|^2$, and similarly $\|\mathbf{C}\mathbf{H}_j\|^2 = \|\sqrt{\mathbf{\Sigma}}\mathbf{H}_j\|^2$ and therefore $\gamma_j = \frac{\|\sqrt{\mathbf{\Sigma}}\mathbf{V}_j\|^2 + \|\sqrt{\mathbf{\Sigma}}\mathbf{H}_j\|^2}{2}$.

This gives $\gamma_j = \left(\frac{\sigma_1(v_{1j}^2+h_{1j}^2)+\sigma_2(v_{2j}^2+h_{2j}^2)+\dots+\sigma_r(v_{rj}^2+h_{rj}^2)}{2} \right)$, or in matrix form

$$\begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_r \end{pmatrix} = \frac{1}{2} \underbrace{\begin{pmatrix} v_{11}^2 & v_{21}^2 & \dots & v_{r1}^2 \\ v_{12}^2 & v_{22}^2 & \dots & v_{r2}^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_{1r}^2 & v_{2r}^2 & \dots & v_{rr}^2 \end{pmatrix}}_{=V^T \odot V^T} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_r \end{pmatrix} + \frac{1}{2} \underbrace{\begin{pmatrix} h_{11}^2 & h_{21}^2 & \dots & h_{r1}^2 \\ h_{12}^2 & h_{22}^2 & \dots & h_{r2}^2 \\ \vdots & \vdots & \ddots & \vdots \\ h_{1r}^2 & h_{2r}^2 & \dots & h_{rr}^2 \end{pmatrix}}_{=H^T \odot H^T} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_r \end{pmatrix}. \quad (8)$$

Minimizing (5) over different factorizations is therefore equivalent to solving

$$\min_{\gamma, M \in \mathcal{S}} f(\gamma), \quad (9)$$

$$\text{s.t.} \quad \gamma = M\sigma. \quad (10)$$

where $\mathcal{S} = \{ \frac{1}{2}(V^T \odot V^T + H^T \odot H^T); VH^T = I \}$. It is clear that $V = H = \Pi^T$, where Π is any permutation, is feasible in the above problem since permutations are orthogonal. In addition they contain only zeros and ones and therefore it is easy to see that this choice gives $\gamma = \frac{1}{2}(\Pi \odot \Pi + \Pi \odot \Pi)\sigma = \Pi\sigma$. In the next section we will show that these are extreme points of the feasible set, in the sense that they can not be written as convex combinations of other points in the set. Extreme points are important for optimization since the global minimum is guaranteed to be attained (if it exists) in such a point if the objective function has concavity properties. This is for example true if f is quasi-concave, that is, the super-level sets $S_\alpha = \{x \in \mathbb{R}_{\geq 0}^r; f(x) \geq \alpha\}$ are convex. To see this let $x = \lambda x_1 + (1 - \lambda)x_2$, and consider the super-level set S_α where $\alpha = \min(f(x_1), f(x_2))$. Since both $x_1 \in S_\alpha$ and $x_2 \in S_\alpha$ it is clear by convexity that so is x and therefore $f(x) \geq \min(f(x_1), f(x_2))$.

2.1 Extreme Points and Optimality

We now consider the optimization problem (9)-(10) and a convex relaxation of the constraint set. For this purpose we let \mathcal{D} be the set of doubly stochastic matrices $\mathcal{D} = \{M \in \mathbb{R}^{r \times r}; m_{ij} \geq 0, \sum_i m_{ij} = 1, \sum_j m_{ij} = 1\}$. Note that if V is orthogonal, and therefore $H = V$, then the row sum $\sum_{j=1}^r \frac{v_{ij}^2+h_{ij}^2}{2}$, and the column sum $\sum_{i=1}^r \frac{v_{ij}^2+h_{ij}^2}{2}$ are both one. Hence such a matrix is in \mathcal{D} . To handle non-orthogonal matrices we define the set of superstochastic matrices \mathcal{S}_W as all matrices $M = D + N$, where $D \in \mathcal{D}$ and N is a matrix with non-negative elements. It can be shown that (see Theorem 6 in [4]) that $\mathcal{S} \subset \mathcal{S}_W$. In addition it is easy to see that \mathcal{S}_W is convex since it

consists of affine constraints. Therefore the problem

$$\min_{\gamma, \mathbf{M} \in \mathcal{S}_W} f(\gamma), \tag{11}$$

$$\text{s.t.} \quad \gamma = \mathbf{M}\sigma. \tag{12}$$

is a relaxation of (9)-(10). Next we show that the two problems have the same minimum if a minimizer to (11)-(12) exists when f is quasi-concave (on $\mathbb{R}_{\geq 0}^r$). As mentioned previously, the minimum (over \mathcal{S}_W) is then attained in an extreme point of \mathcal{S}_W . We therefore need the following characterization.

Lemma 1. *The extreme points of \mathcal{S}_W are $r \times r$ permutation matrices.*

Proof. First we note that any extreme point of \mathcal{S}_W has to be in \mathcal{D} since if $\mathbf{M} = \mathbf{D} + \mathbf{N}$ with $\mathbf{N} \neq 0$ then $\mathbf{M} = \frac{1}{2}\mathbf{D} + \frac{1}{2}(\mathbf{D} + 2\mathbf{N})$, which is a convex combination of two points in \mathcal{S}_W . By Birkhoff's Theorem [15] any matrix in \mathcal{D} can be written as a convex combination of permutation matrices. \square

Since permutation matrices are orthogonal with 0/1 elements it is clear they can be written $\mathbf{\Pi} = \frac{1}{2}(\mathbf{\Pi} \odot \mathbf{\Pi} + \mathbf{\Pi} \odot \mathbf{\Pi})$, with $\mathbf{\Pi}\mathbf{\Pi}^T = \mathbf{I}$. Therefore the extreme points of \mathcal{S}_W are also in \mathcal{S} . Hence if the minimum of (11)-(12) is attained, there is an optimal extreme point of \mathcal{S}_W which also solves (9)-(10), and therefore the solution is given by a permutation $\mathbf{V} = \mathbf{H} = \mathbf{\Pi}$.

We conclude this section by giving sufficient conditions for the minimum of (11)-(12) to exist, namely that f is lower semi-continuous and non-decreasing in all of its variables, that is, if $\tilde{\gamma}_i \geq \gamma_i$ for all i then $f(\tilde{\gamma}) \geq f(\gamma)$. Since the singular values are all positive it is clear that the elements of $(\mathbf{D} + \mathbf{N})\sigma$ are larger than those of $\mathbf{D}\sigma$. Hence when f is non-decreasing it is enough to consider minimization over \mathcal{D} . We then have a lower semi-continuous objective function on a compact set for which the minimum is known to be attained.

We can now summarize the results of this section in the following theorem:

Theorem 1. *Let f be quasi-concave (and lower semi-continuous) on $\mathbb{R}_{\geq 0}^r$ fulfilling $f(\tilde{\gamma}) \geq f(\gamma)$ when $\tilde{\gamma}_i \geq \gamma_i$ for all i . Then there is an optimal γ^* of (9)-(10) that is of the form $\gamma^* = \mathbf{\Pi}\sigma$ where $\mathbf{\Pi}$ is a permutation.*

3 Non-Square Matrices

In the previous section we made the assumption that \mathbf{V} and \mathbf{H} were square matrices, which corresponds to searching over $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ consisting of r columns when $\text{rank}(\mathbf{X}) = r$. In addition since \mathbf{V} and \mathbf{H} are invertible this means that $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ have linearly independent columns. In this section we generalize the result from Section 2.1

to rectangular matrices \mathbf{V} and \mathbf{H} . Therefore we suppose that \mathbf{V} and \mathbf{H} are non-square of size $r \times p$, $p > r$, with $\mathbf{V}\mathbf{H}^T = \mathbf{I}_{r \times r}$, and consider the slightly modified problem

$$\min_{\gamma, \mathbf{V}, \mathbf{H}} f(\gamma), \quad (13)$$

$$\gamma = \frac{1}{2}(\mathbf{V}^T \odot \mathbf{V}^T + \mathbf{H}^T \odot \mathbf{H}^T)\boldsymbol{\sigma} \quad (14)$$

$$\mathbf{V}\mathbf{H}^T = \mathbf{I}_{r \times r} \quad (15)$$

Note that $\mathbf{V}\mathbf{H}^T$ do not commute and we therefore only assume that \mathbf{V} is a left inverse of \mathbf{H}^T . In what follows we show that by adding zeros to the vector $\boldsymbol{\sigma}$ we can extend \mathbf{V} , \mathbf{H} into square matrices without changing the objective function.

Note that we may assume that \mathbf{V} has full row rank since otherwise $\mathbf{X} \neq \mathbf{B}\mathbf{V}\mathbf{H}^T\mathbf{C}^T$. Let \mathbf{V}^\dagger be the Moore–Penrose pseudo inverse and $\mathbf{O}_{\mathbf{V}_\perp}$ a $(p-r) \times p$ matrix containing a basis for the space orthogonal to the row space of \mathbf{V} (and the column space of \mathbf{V}^\dagger). Since $\mathbf{V}\mathbf{H}^T = \mathbf{I}_{r \times r}$ the matrix \mathbf{H}^T is of the form $\mathbf{H}^T = \mathbf{V}^\dagger + \mathbf{O}_{\mathbf{V}_\perp}^T \mathbf{K}_1$, where \mathbf{K}_1 is a $(p-r) \times r$ coefficient matrix. We now want to find matrices $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{H}}$ such that $\begin{bmatrix} \mathbf{V} \\ \tilde{\mathbf{V}} \end{bmatrix} \begin{bmatrix} \mathbf{V}^\dagger + \mathbf{O}_{\mathbf{V}_\perp}^T \mathbf{K}_1 & \tilde{\mathbf{H}}^T \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{r \times r} & 0 \\ 0 & \mathbf{I}_{(p-r) \times (p-r)} \end{bmatrix}$. To do this we first select $\tilde{\mathbf{H}}^T = \mathbf{O}_{\mathbf{V}_\perp}^T$ since $\mathbf{V}\mathbf{O}_{\mathbf{V}_\perp}^T = 0$. Then we let $\tilde{\mathbf{V}} = \mathbf{O}_{\mathbf{V}_\perp} + \mathbf{K}_2\mathbf{V}$, where \mathbf{K}_2 is a size $(p-r) \times r$ coefficient matrix, since this gives $\tilde{\mathbf{V}}\tilde{\mathbf{H}}^T = \mathbf{I}_{(p-r) \times (p-r)}$. To determine \mathbf{K}_2 we consider $\tilde{\mathbf{V}}(\mathbf{V}^\dagger + \mathbf{O}_{\mathbf{V}_\perp}^T \mathbf{K}_1) = \mathbf{K}_2\mathbf{I}_{r \times r} + \mathbf{I}_{(p-r) \times (p-r)}\mathbf{K}_1 = \mathbf{K}_2 + \mathbf{K}_1$. Selecting $\mathbf{K}_2 = -\mathbf{K}_1$ thus gives square matrices such that $\begin{bmatrix} \mathbf{V} \\ \tilde{\mathbf{V}} \end{bmatrix} \begin{bmatrix} \mathbf{H}^T & \tilde{\mathbf{H}}^T \end{bmatrix} = \mathbf{I}$. Further letting $\tilde{\boldsymbol{\Sigma}} = \begin{bmatrix} \boldsymbol{\Sigma} & 0 \\ 0 & 0 \end{bmatrix}$ shows that $\|\mathbf{B}\mathbf{V}_i\| = \|\sqrt{\tilde{\boldsymbol{\Sigma}}} \begin{bmatrix} \mathbf{V}_i \\ \tilde{\mathbf{V}}_i \end{bmatrix}\|$ and $\|\mathbf{C}\mathbf{H}_i\| = \|\sqrt{\tilde{\boldsymbol{\Sigma}}} \begin{bmatrix} \mathbf{H}_i \\ \tilde{\mathbf{H}}_i \end{bmatrix}\|$ and the results of the previous section give that the minimizer of $f(\gamma_1, \gamma_2, \dots, \gamma_p)$ is a permutation of the elements in the vector $(\sigma_1, \sigma_2, \dots, \sigma_r, 0, \dots, 0)$. We therefore have the following result:

Corollary 1. *Let f be quasi-concave (and lower semi-continuous) on $\mathbb{R}_{\geq 0}^p$ fulfilling $f(\tilde{\gamma}) \geq f(\gamma)$ when $\tilde{\gamma}_i \geq \gamma_i$ for all i . Then an optimizer $\boldsymbol{\gamma}^*$ of (13)-(15) is of the form $\boldsymbol{\gamma}^* = \boldsymbol{\Pi}_{p \times r} \boldsymbol{\sigma}$ where $\boldsymbol{\Pi}_{p \times r}$ contains the first r columns of a $p \times p$ permutation matrix.*

4 Linear Objectives – Weighted Nuclear Norms

We now consider weighted nuclear norm regularization $f(\boldsymbol{\gamma}) = \mathbf{a}^T \boldsymbol{\gamma}$. To ensure that the problem is well posed we assume that the elements of \mathbf{a} are non-negative. It is then

clear that $f(\tilde{\gamma}) \geq f(\gamma)$ when $\tilde{\gamma}_i \geq \gamma_i$. Since linearity implies concavity the results of Sections 2.1 and 3 now show that the minimum of $f(\mathbf{M}\sigma)$, over $\mathbf{M} \in \mathcal{S}$ is attained in $\mathbf{M} = \mathbf{\Pi}$ for some permutation matrix. To ensure that the bilinear formulation is equivalent to the original one we need to show that the optimum occurs when $\mathbf{\Pi} = \mathbf{I}$. Suppose that the elements in \mathbf{a} are sorted in ascending order $a_1 \leq a_2 \leq \dots \leq a_p$. It is easy to see that for $\mathbf{\Pi}$ to give the smallest objective value it should sort the elements of γ so that $\gamma_1 \geq \gamma_2 \geq \dots \geq \gamma_p$, which means that $\mathbf{\Pi} = \mathbf{I}$ and $\gamma = \sigma$. We therefore conclude that minimizing (4) with a linear objective corresponds to regularization with a weighted nuclear norm with non-decreasing weights.

5 Experiments

In this section we start by describing implementation details of our method and then apply it to the problems of low matrix recovery and non-rigid structure recovery. Solving the weighted nuclear norm regularized problem (7) now amounts to minimizing

$$\sum_{i=1}^p a_i \frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2} + \|\mathcal{A}(\mathbf{BC}^T) - \mathbf{b}\|^2. \quad (16)$$

Note that the terms in the (16) can be combined into a single norm term by vertically concatenating the vectors \mathbf{B}_i and \mathbf{C}_i , weighted by $\sqrt{a_i/2}$, with $\mathcal{A}(\mathbf{BC}^T) - \mathbf{b}$. We define the resulting vector as $\mathbf{r}_a := \mathcal{A}_a(\mathbf{BC}^T) - \mathbf{b}_a$, giving the objective $\|\mathbf{r}_a(\mathbf{BC}^T)\|^2$, where the subscript reflects the dependence on the weights \mathbf{a} . Since the objective is smooth, standard methods such as Levenberg–Marquardt can be applied and Algorithm 1 shows an overview of the method used. Additional information about the algorithm is provided in the supplementary material.

The remainder of this section is organized as follows. The particular form of the data fitting term in (16) when applied to structure from motion is described in Section 5.1. In Section 5.2 we compare the convergence of first and second-order methods, and motivated by the ADMM fast iterations but low accuracy, as opposed to the bilinear parameterization’s high accuracy but slower iterations, we combine the two methods by initializing the bilinear parameterization with the ADMM’s solution [5, 27] for a non-rigid structure structure recovery problem. Our work focus on the increased accuracy of our method compared to first-order methods, so the comparison of our results with works such as [1, 31, 43, 45] (without the desired regularization term) are not covered.

Result: Optimal \mathbf{B} , \mathbf{C} to (16)
 $\mathbf{B} = \mathbf{U}\Sigma^{\frac{1}{2}}$, $\mathbf{C} = \mathbf{V}\Sigma^{\frac{1}{2}}$, where $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$ is the ADMM solution to (16) ;
 Choose initial $\alpha > 1$ and λ , and define $\mathbf{z} = [\text{vec}(\mathbf{B}); \text{vec}(\mathbf{C}^T)]$;
 Compute $error = \|r_a(\mathbf{B}\mathbf{C}^T)\|^2$;
while not converged do
 Compute $\mathbf{r} = r_a(\mathbf{B}\mathbf{C}^T)$, and the Jacobian \mathbf{J} of (16) in terms of \mathbf{z} ;
 Update $\tilde{\mathbf{z}} = \mathbf{z} - (\mathbf{J}^T\mathbf{J} + \lambda\mathbf{I})^{-1}\mathbf{J}^T\mathbf{r}$, where $\tilde{\mathbf{z}} = [\text{vec}(\tilde{\mathbf{B}}); \text{vec}(\tilde{\mathbf{C}}^T)]$;
 if $error > \|r_a(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T)\|^2$ **then**
 Update $\mathbf{z} \leftarrow \tilde{\mathbf{z}}$, $\lambda \leftarrow \alpha^{-1}\lambda$, and $error \leftarrow \|\mathcal{A}_a(\tilde{\mathbf{B}}\tilde{\mathbf{C}}^T) - \mathbf{b}_a\|^2$;
 else
 $\lambda \leftarrow \alpha\lambda$;
 end
end

Algorithm 1: Bilinear parameterization of weighted nuclear norm

5.1 Pseudo Object Space Error (pOSE) and Non-Rigid Structure from Motion

To compare the performance of 1st and 2nd order methods, we choose as objective function the Pseudo Object Space Error (pOSE) [21], which consists of a combination of the object space error $\ell_{\text{OSE}} := \sum_{(i,j) \in \Omega} \|\mathbf{P}_{i,1:2}\tilde{\mathbf{x}}_j - (\mathbf{p}_{i,3}^T\tilde{\mathbf{x}}_j)\mathbf{m}_{i,j}\|_2^2$ and the affine projection error $\ell_{\text{Affine}} := \sum_{(i,j) \in \Omega} \|\mathbf{P}_{i,1:2}\tilde{\mathbf{x}}_j - \mathbf{m}_{i,j}\|_2^2$, where $\mathbf{P}_{i,1:2}$ and $\mathbf{p}_{i,3}$ are, respectively, the first two and the third rows of the camera matrix \mathbf{P}_i , with $i = 1, \dots, F$; $\tilde{\mathbf{x}}_j$ is a 3D point in homogeneous coordinates, with $j = 1, \dots, P$; $\mathbf{m}_{i,j}$ is the 2D observation of the j :th point on the i :th camera; and Ω represents the set of observable data. The pOSE is then given by $\ell_{\text{pOSE}} := (1 - \eta)\ell_{\text{OSE}} + \eta\ell_{\text{Affine}}$ where $\eta \in [0, 1]$ balances the weight between the two errors. One of the main properties of pOSE is its wide basin of convergence while keeping a bilinear problem structure. The ℓ_{pOSE} , originally designed for rigid SfM, can be extended for the non-rigid case by replacing $\mathbf{P}_i\tilde{\mathbf{x}}_j$ by a linear combination of K shape basis, *i.e.*, $\mathbf{\Pi}_i\hat{\mathbf{S}}_j$, where $\mathbf{\Pi}_i \in \mathbb{R}^{3 \times (3K+1)}$ and $\hat{\mathbf{S}}_j \in \mathbb{R}^{3K+1}$ are structured as $\mathbf{\Pi}_i = [c_{i,1}\mathbf{R}_i \ \cdots \ c_{i,K}\mathbf{R}_i \ \mathbf{t}_i]$ and $\hat{\mathbf{S}}_j = [\mathbf{S}_{1,j}^T \ \cdots \ \mathbf{S}_{K,j}^T \ 1]^T$. We denote by $\mathbf{\Pi}$ and $\hat{\mathbf{S}}$ the vertical and horizontal concatenations of $\mathbf{\Pi}_i$ and $\hat{\mathbf{S}}_j$, respectively. Note that by construction $\text{rank}(\mathbf{\Pi}\hat{\mathbf{S}}) \leq 3K + 1$, and for $K = 1$ we have $\mathbf{\Pi}_i\hat{\mathbf{S}}_j = \mathbf{P}_i\tilde{\mathbf{x}}_j$, which corresponds to the rigid case.

5.2 Low-Rank Matrix Recovery with pOSE Errors

In this section we compare the convergence and accuracy of 1st and 2nd order methods, starting from the same initial guess, for low-rank matrix recovery with pOSE. In this problem, we define $\mathbf{X} = \mathbf{II}\hat{\mathbf{S}}$ and aim at minimizing

$$\min_{\mathbf{X}} \mathbf{a}^T \boldsymbol{\sigma}(\mathbf{X}) + \ell_{\text{pOSE}}(\mathbf{X}). \quad (17)$$

We apply our method and solve the problem (16) by using the bilinear factorization $\mathbf{X} = \mathbf{BC}^T$, with $\mathbf{B} \in \mathbb{R}^{3F \times r}$, and $\mathbf{C} \in \mathbb{R}^{P \times r}$, with $r \geq 3K + 1$. We test the performance of our method in 4 datasets: Door [33], Back [37], Heart [39], Paper [41]. The first one consists of image measurements of a rigid structure with missing data, while the remaining three datasets track points in deformable structures.

For the Door dataset, we apply two different selections of weights on the singular values of \mathbf{X} , corresponding to the nuclear norm, *i.e.*, $a_i = \mu_{NN}$, and truncated nuclear norm, *i.e.*, $a_i = 0, i = 1, \dots, 4$ and $a_i = \mu_{TNN}, i > 4$. We select $\mu_{NN} = 1.5 \times 10^{-3}$, and $\mu_{TNN} = 1$. For the Back, Heart and Paper datasets, we apply the nuclear norm and a weighted nuclear norm, in which the first four singular values of \mathbf{X} are not penalized and the remaining ones are increasingly penalized, *i.e.*, $a_i = 0, i = 1, \dots, 4$ and $a_i = (i - 4)\mu_{WNN}, i > 4$. We select $\mu_{NN} = 7.5 \times 10^{-4}$, $\mu_{WNN} = 2.25 \times 10^{-3}$. The values of the weights a_i are chosen such that there is a $3K + 1$ rank solution to (17), with $K = 1$ and $K = 2$ for the rigid and non-rigid datasets, respectively.

We compare the bilinear parameterization with three first-order methods commonly used for low-rank matrix recovery: Alternating Direction Method of Multipliers (ADMM) [5], Iteratively Reweighted Nuclear Norm (IRNN) [12], and Accelerated Gradient Descend (AGD) [30]. We also test the methods for two different cases of the ℓ_{pOSE} error, with $\eta = 0.05$ and $\eta = 0.95$, which correspond to the near-perspective and near-affine camera models, respectively. To improve numerical stability of the algorithms, as pre-processing step we normalize the image measurements matrix \mathbf{M} by its norm. The methods are initialized with the closed-form solution of the regularization-free problem, *i.e.*, $\mathbf{X} = \mathcal{A}^\dagger(\mathbf{b})$. The comparison of the four algorithms in terms of total log-loss over time is shown in Figure 1. The log-loss is used for better visualization purposes. The plots for the IRNN for the nuclear norm are omitted since it demonstrated slow convergence compared to the remaining three methods. A qualitative evaluation of the results on one of the images of the Door dataset for the truncated nuclear norm and near perspective camera model is shown in Figure 2. The qualitative results for the remaining datasets are provided in the supplementary material.

In general, we can observe that first-order methods demonstrate faster initial convergence, mostly due to faster iterations. However when near minima, the convergence rate

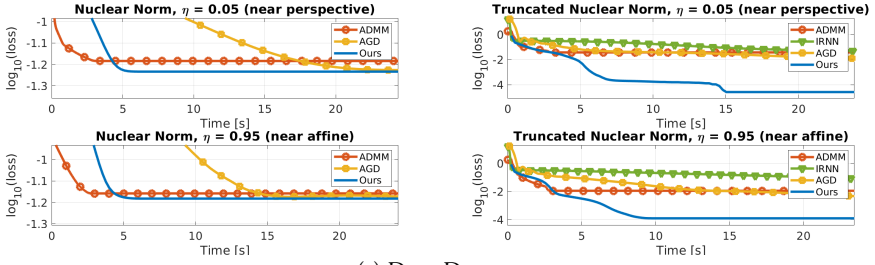
drops significantly and the methods tend to stall. Contrarily, bilinear parameterization compensates its slower iterations by demonstrating higher accuracy and reaching solutions with lower energy. This is specially visible for the near perspective camera model, which reinforces the advantages of using a second-order method on image data under perspective projection. To compensate for the slower convergence, we propose the initialization of the bilinear parameterization with the solution obtained with ADMM. In this way, the bilinear parameterization starts near the minimum and performs local refinement to further improve accuracy.

5.3 Non-Rigid Structure Recovery

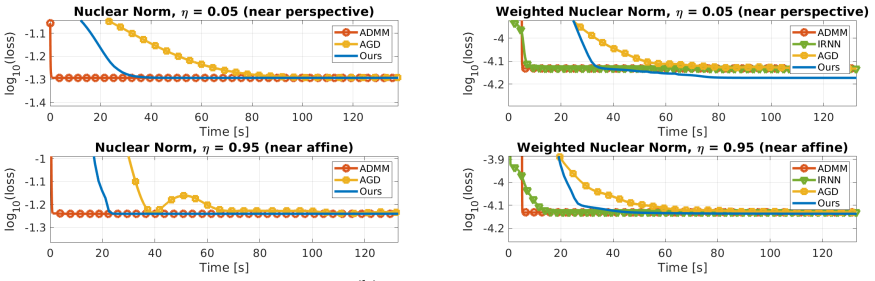
Consider now that the camera rotations in $\mathbf{\Pi}$ are known (or previously estimated). In this case we have $\mathbf{\Pi}\hat{\mathbf{S}} = \mathbf{R}\mathbf{X} + \mathbf{t}\mathbb{1}^T$, with $\mathbf{R} = \text{blkdiag}(\mathbf{R}_1, \dots, \mathbf{R}_F)$ and $\mathbf{t} = [t_1^T, \dots, t_F^T]^T$, where \mathbf{X} , the non-rigid structure, and \mathbf{t} are the unknowns. It is directly observed that $\text{rank}(\mathbf{\Pi}\hat{\mathbf{S}}) \leq \text{rank}(\mathbf{R}\mathbf{X}) + \text{rank}(\mathbf{t}\mathbb{1}^T)$, with the latter being equal to 1 by construction and independent on K . As consequence, it follows that $\text{rank}(\mathbf{R}\mathbf{X}) = \text{rank}(\mathbf{X}) \leq 3K$, and the rank regularization can be applied on \mathbf{X} . A similar problem was studied in [13] but for orthogonal camera models, where the authors propose the rank regularization to be applied on a reshaped version of \mathbf{X} , given by $\mathbf{X}^\# = g^{-1}(\mathbf{X})$, a $F \times 3P$, where the function g performs the permutation on the elements of $\mathbf{X}^\#$ to obtain \mathbf{X} . With this reshaping we have that $\text{rank}(\mathbf{X}^\#) \leq K$, meaning that we can factorize it as $\mathbf{X}^\# = \mathbf{B}\mathbf{C}^T$ with $\mathbf{B} \in \mathbb{R}^{F \times K}$ and $\mathbf{C} \in \mathbb{R}^{3P \times K}$. The optimization problem then becomes

$$\min_{\mathbf{B}, \mathbf{C}, \mathbf{t}} \sum_{i=1}^K a_i \frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2} + \ell_{\text{pOSE}}(\mathbf{R}g(\mathbf{B}\mathbf{C}^T) + \mathbf{t}\mathbb{1}^T). \quad (18)$$

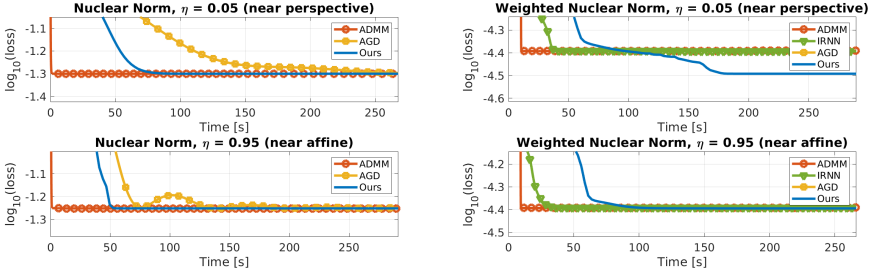
Solving this optimization problem requires small adjustments to be done to the proposed Algorithm 1, which can be consulted in the supplementary material. We apply our methods to the 5 datasets (Articulated, Balloon, Paper, Stretch, Tearing) from the NRSfM Challenge [25]. Each of these datasets include tracks of image points for orthogonal and perspective camera models for six different camera paths (Circle, Flyby, Line, Semi-circle, Tricky, Zigzag), as well as the ground-truth 3D structure for one of the frames. We use the 2D observation for the orthogonal camera model to compute the rotation matrix \mathbf{R} , as done in [13], and the ground-truth 3D structure to estimate the intrinsic camera matrix, which is assumed to be fixed during each sequence. The intrinsic camera matrix is used to obtain the calibrated 2D observation of the perspective camera model data. For the nuclear norm (NN), we set $a_i = 1 \times 10^{-3}$, $i = 1, \dots, K$. For the weighted



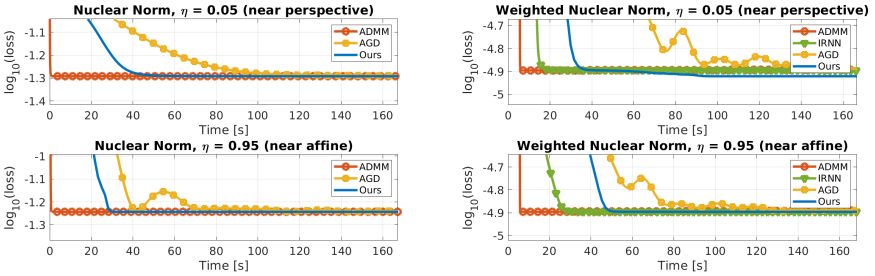
(a) Door Dataset.



(b) Heart Dataset.



(c) Back Dataset.



(d) Paper Dataset.

Figure 1: Convergence of the four methods for low-rank matrix recovery on the Door, Heart, Back and Paper datasets.

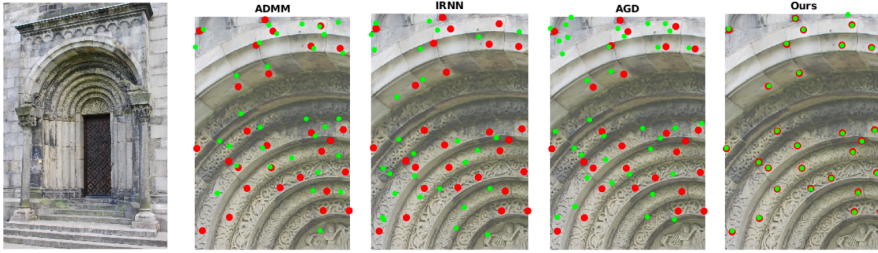


Figure 2: Evaluation of the four methods for low-rank matrix recovery on one of the images of the Door dataset. The red circles show the target image measurements and the green circles the estimate image points.

nuclear norm (WNN), the weights a are selected similarly to [27] $a_i = \frac{\zeta}{\sigma_i(g^{-1}(\mathbf{X}_0)) + \gamma}$, $i = 1, \dots, K$ where $\zeta = 5 \times 10^{-3}$, γ is a small number for numerical stability, and \mathbf{X}_0 is the closed-form solution of the objective $\min_{\mathbf{X}} \ell_{\text{POSE}}(\mathbf{R}\mathbf{X})$.

For these datasets we choose $K = 2$ and set the $\eta = 0.05$. As baseline we use the best performing first-order method according to the experiments Section 5.2, ADMM, and apply the method described in Algorithm 1 for local refinement starting from the ADMM’s solution. We also try our method for the orthogonal camera model (by setting $\eta = 1$), and compare it with BMM [13] and R-BMM [27], which correspond to ADMM implementations for nuclear norm and weighted nuclear norm, respectively. These methods perform a best rank K approximation to the obtained ADMM solution if $\text{rank}(\mathbf{X}^\#) > K$ after convergence. We let the ADMM-based methods run until convergence or stalling is achieved for fair comparison. The average log-losses, before and after refinement, obtained on each dataset are shown in Table 1. The average reconstruction errors, in millimeters, on each dataset relatively to the provided ground-truth structure are shown in Table 2. In Figure 3 we also show some qualitative results of the obtained 3D reconstruction of each of the objects in the 5 datasets. More qualitative results are provided in the supplementary material.

The results show that our method is able to achieve lower energies for all datasets comparatively with the ADMM baselines. Similarly to Section 5.2, the difference is more substantial for the perspective model. Furthermore, even though we are not explicitly minimizing the reconstruction error expressed in Table 2, we are able to consistently obtain the lowest reconstruction error for all datasets, sometimes with great improvements compared to the ADMM (see Balloon and Stretch in Figure 3). The same does not apply for the orthogonal data, where achieving lower energies did not lead to lower reconstruction errors.

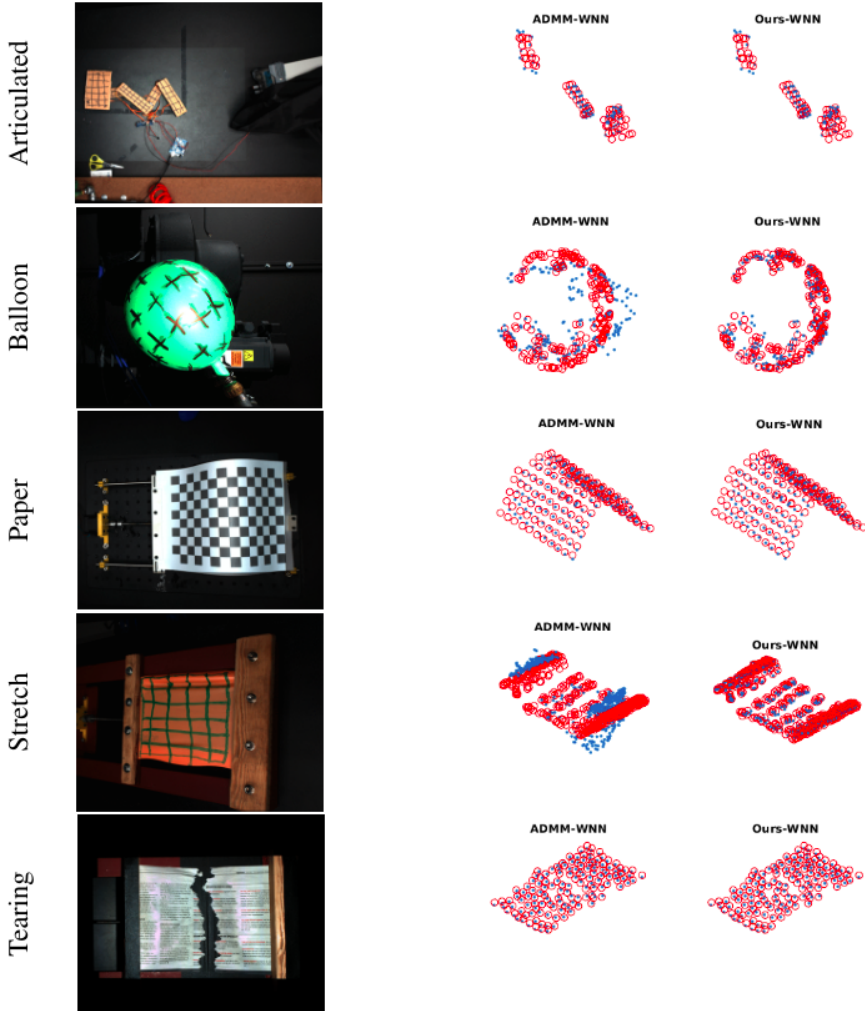


Figure 3: (Left) Example of the non-rigid objects in the 5 datasets of the NRSfM Challenge. (Right) Estimation (blue) and ground-truth (red) of the non-rigid 3D structure for the two methods with weighted nuclear norm regularization.

Table 1: Average log-loss on each of the perspective datasets over the 6 camera paths.

	Method \ Dataset	Articulated	Balloon	Paper	Stretch	Tearing
Orthogonal	BMM [13]	-1.645	-2.267	-1.712	-2.282	-1.453
	Ours-NN	-1.800	-2.352	-2.188	-2.509	-1.634
	R-BMM [27]	-1.648	-1.979	-1.855	-1.997	-1.522
	Ours-WNN	-1.648	-1.979	-1.855	-1.997	-1.522
Perspective	ADMM-NN	-2.221	-2.529	-2.338	-2.395	-1.471
	Ours-NN	-2.415	-2.657	-2.560	-2.622	-2.053
	ADMM-WNN	-2.455	-2.617	-2.195	-2.651	-1.688
	Ours-WNN	-2.486	-2.931	-2.777	-2.857	-2.103

Table 2: Average reconstruction errors, in millimeters, on each dataset over the 6 camera paths relatively to the provided ground-truth structure.

	Method \ Dataset	Articulated	Balloon	Paper	Stretch	Tearing
Orthogonal	BMM [13]	18.49	10.39	8.94	10.02	14.23
	Ours-NN	18.31	8.53	10.94	10.67	17.03
	R-BMM [27]	16.00	7.84	10.69	7.53	16.34
	Ours-WNN	15.03	8.05	10.45	9.01	16.20
Perspective	ADMM-NN	16.70	8.05	7.96	6.04	9.40
	Ours-NN	16.13	6.48	6.80	6.00	9.31
	ADMM-WNN	18.33	8.95	10.14	8.06	9.28
	Ours-WNN	16.53	6.27	5.68	5.93	8.42

6 Conclusions

In this paper we show that it is possible to optimize a general class of singular value penalties using a bilinear parameterization of the matrix. We show that with this parameterization weighted nuclear norm penalties turn in to smooth objectives that can be accurately solved with 2nd order methods. Our proposed approach starts by using ADMM which rapidly decreases the objective the first couple of iterations and switches to Levenberg–Marquardt when ADMM iterations make little progress. This results in a much more accurate solution and we showed that we were able to extend the recently proposed pOSE [21] to handle non-rigid reconstruction problems.

While 2nd order methods offer increased accuracy, our approach is expensive since iterations require the inversion of a large matrix. Exploring feasible alternatives such as preconditioning and conjugate gradient approaches is an interesting future direction.

Something that we have not discussed is adding constraints on the factors, which is possible since these are present in the optimization. This is very relevant for structure from motion problems and will likely be an fruitful direction to explore.

References

- [1] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. “Nonrigid structure from motion in trajectory space”. In: *Proceedings of the 21st International Conference on Neural Information Processing Systems*. 2008, pp. 41–48.
- [2] F. R. Bach. “Convex relaxations of structured matrix factorizations”. In: *CoRR* abs/1309.3117 (2013).
- [3] R. Basri, D. Jacobs, and I. Kemelmacher. “Photometric Stereo with General, Unknown Lighting”. In: *International Journal of Computer Vision* 72.3 (May 2007), pp. 239–257. ISSN: 0920-5691.
- [4] R. Bhatia and T. Jain. “On symplectic eigenvalues of positive definite matrices”. In: *Journal of Mathematical Physics* 56.11 (2015), p. 112201.
- [5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Found. Trends Mach. Learn.* 3.1 (2011), pp. 1–122.
- [6] C. Bregler, A. Hertzmann, and H. Biermann. “Recovering non-rigid 3D shape from image streams”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2000.

- [7] A. M. Buchanan and A. W. Fitzgibbon. “Damped newton algorithms for matrix factorization with missing data”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2005.
- [8] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [9] E. J. Candès, X. Li, Y. Ma, and J. Wright. “Robust Principal Component Analysis?” In: *J. ACM* 58.3 (2011), 11:1–11:37.
- [10] E. J. Candès and B. Recht. “Exact matrix completion via convex optimization”. In: *Foundations of Computational Mathematics* 9.6 (2009), pp. 717–772.
- [11] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Generalized Nonconvex Nonsmooth Low-Rank Minimization”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014).
- [12] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Nonconvex Nonsmooth Low-Rank Minimization via Iteratively Reweighted Nuclear Norm”. In: *IEEE Transactions on Image Processing* 25 (Oct. 2015).
- [13] Y. Dai, H. Li, and M. He. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization”. In: *International Journal of Computer Vision* 107.2 (2014), pp. 101–122.
- [14] M. Fazel, H. Hindi, and S. P. Boyd. “A rank minimization heuristic with application to minimum order system approximation”. In: *American Control Conference*. 2001.
- [15] G. Birkhoff. “Tres observaciones sobre el algebra lineal”. In: *Universidad Nacional de Tucuman, Revista. Serie A*. 1946, pp. 137–151.
- [16] R. Garg, A. Roussos, and L. Agapito. “A Variational Approach to Video Registration with Subspace Constraints”. In: *International Journal of Computer Vision* 104.3 (2013), pp. 286–314.
- [17] R. Garg, A. Roussos, and L. de Agapito. “Dense Variational Reconstruction of Non-rigid Surfaces from Monocular Video”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.
- [18] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang. “Weighted Nuclear Norm Minimization and Its Applications to Low Level Vision”. In: *International Journal of Computer Vision* 121 (July 2016), pp. 183–208.

-
- [19] B. D. Haeffele and R. Vidal. “Structured Low-Rank Matrix Factorization: Global Optimality, Algorithms, and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.6 (2020), pp. 1468–1482.
- [20] J. H. Hong and A. Fitzgibbon. “Secrets of Matrix Factorization: Approximations, Numerics, Manifold Optimization and Random Restarts”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.
- [21] J. H. Hong and C. Zach. “pOSE: Pseudo Object Space Error for Initialization-Free Bundle Adjustment”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018, pp. 1876–1885.
- [22] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.
- [23] J. H. Hong, C. Zach, A. Fitzgibbon, and R. Cipolla. “Projective Bundle Adjustment from Arbitrary Initialization Using the Variable Projection Method”. In: *European Conference on Computer Vision (ECCV)*. 2016, pp. 477–493.
- [24] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. “Fast and Accurate Matrix Completion via Truncated Nuclear Norm Regularization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2117–2130.
- [25] S. H. N. Jensen, A. Del Bue, M. E. B. Doest, and H. Aanæs. “A Benchmark and Evaluation of Non-Rigid Structure from Motion”. In: *arXiv preprint arXiv:1801.08388* (2018).
- [26] M. Krechetov, J. Marecek, Y. Maximov, and M. Takac. “Entropy-Penalized Semidefinite Programming”. In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. International Joint Conferences on Artificial Intelligence Organization, July 2019, pp. 1123–1129.
- [27] S. Kumar. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization : Revisited”. In: *CoRR* abs/1902.10274 (2019). URL: <http://arxiv.org/abs/1902.10274>.
- [28] V. Larsson and C. Olsson. “Convex Low Rank Approximation”. In: *International Journal of Computer Vision* 120.2 (2016), pp. 194–214.
- [29] A. S. Lewis. “The convex analysis of unitarily invariant matrix functions”. In: *Journal of Convex Analysis* 2.1 (1995), pp. 173–183.
- [30] H. Li and Z. Lin. “Provable accelerated gradient method for nonconvex low rank optimization”. In: *Machine Learning* 109 (June 2019).

- [31] M. Paladini, A. Del Bue, M. Stolic, M. Dodig, J. Xavier, and L. Agapito. “Factorization for non-rigid and articulated structure using metric projections”. In: 2009, pp. 2898–2905.
- [32] T. H. Oh, Y. W. Tai, J. C. Bazin, H. Kim, and I. S. Kweon. “Partial Sum Minimization of Singular Values in Robust PCA: Algorithm and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4 (2016), pp. 744–758.
- [33] C. Olsson and O. Enqvist. “Stable Structure from Motion for Unordered Image Collections”. In: *Proceedings of the Scandinavian Conference on Image Analysis (SCIA)*. 2011, pp. 524–535.
- [34] M. V. Örnberg, C. Olsson, and A. Heyden. “Bilinear Parameterization For Differentiable Rank-Regularization”. In: *CoRR* abs/1811.11088 (2018).
- [35] S. Oymak, K. Mohan, M. Fazel, and B. Hassibi. “A simplified approach to recovery conditions for low rank matrices”. In: *IEEE International Symposium on Information Theory Proceedings (ISIT)*. 2011, pp. 2318–2322.
- [36] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *SIAM Rev.* 52.3 (Aug. 2010), pp. 471–501.
- [37] C. Russell, J. Fayad, and L. Agapito. “Energy based multiple model fitting for non-rigid structure from motion”. In: 2011, pp. 3009–3016.
- [38] F. Shang, J. Cheng, Y. Liu, Z. Luo, and Z. Lin. “Bilinear Factor Matrix Norm Minimization for Robust PCA: Algorithms and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.9 (Sept. 2018), pp. 2066–2080.
- [39] D. Stoyanov, G. Mylonas, F. Deligianni, A. Darzi, and G. Yang. “Soft-Tissue Motion Tracking and Structure Estimation for Robotic Assisted MIS Procedures”. In: vol. 8. Feb. 2005, pp. 139–146.
- [40] C. Tomasi and T. Kanade. “Shape and Motion from Image Streams Under Orthography: A Factorization Method”. In: *International Journal of Computer Vision* 9.2 (1992), pp. 137–154.
- [41] A. Varol, M. Salzmann, E. Tola, and P. Fua. “Template-free monocular reconstruction of deformable surfaces”. In: *International Conference on Computer Vision (ICCV)*. 2009, pp. 1811–1818.
- [42] N. Wang, T. Yao, J. Wang, and D.-Y. Yeung. “A Probabilistic Approach to Robust Matrix Factorization”. In: *European Conference on Computer Vision (ECCV)*. 2012.

- [43] J. Xiao, J.-x. Chai, and T. Kanade. “A Closed-Form Solution to Non-rigid Shape and Motion Recovery”. In: *European Conference on Computer Vision (ECCV)*. May 2004, pp. 573–587.
- [44] C. Xu, Z. Lin, and H. Zha. “A Unified Convex Surrogate for the Schatten- p Norm”. In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2017.
- [45] J. Yan and M. Pollefeys. “A Factorization-Based Approach for Articulated Non-rigid Shape, Motion and Kinematic Chain Recovery From Video”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.5 (2008), pp. 865–877.
- [46] J. Yan and M. Pollefeys. “A Factorization-Based Approach for Articulated Non-rigid Shape, Motion and Kinematic Chain Recovery From Video”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 30.5 (2008), pp. 865–877.

Bilinear Parameterization for Non-Separable Singular Value Penalties

MARCUS VALTONEN ÖRNHAG¹, JOSÉ PEDRO IGLESIAS² AND CARL OLSSON^{1,2}

¹*Centre for Mathematical Sciences, Lund University*

²*Department of Electrical Engineering, Chalmers University of Technology*

Abstract: Low rank inducing penalties have been proven to successfully uncover fundamental structures considered in computer vision and machine learning; however, such methods generally lead to non-convex optimization problems. Since the resulting objective is non-convex one often resorts to using standard splitting schemes such as Alternating Direction Methods of Multipliers (ADMM), or other subgradient methods, which exhibit slow convergence in the neighbourhood of a local minimum. We propose a method using second order methods, in particular the variable projection method (VarPro), by replacing the non-convex penalties with a surrogate capable of converting the original objectives to differentiable equivalents. In this way we benefit from faster convergence.

The bilinear framework is compatible with a large family of regularizers, and we demonstrate the benefits of our approach on real datasets for rigid and non-rigid structure from motion. The qualitative difference in reconstructions show that many popular non-convex objectives enjoy an advantage in transitioning to the proposed framework.¹

1 Introduction

Low rank approximation and factorization methods are classical approaches for solving various computer vision problems, such as structure from motion [5, 11, 18, 24, 25, 38], photometric stereo [2, 7, 29] image segmentation [15], image restoration [9, 21, 27, 29, 45], background/foreground segmentation [7, 43], etc.

There are two main approaches when it comes to solving these problems, and which one is used largely depends on properties of the particular problem being addressed. The classical problem of low rank recovery with missing data

$$\min_{\text{rank}(\mathbf{X}) \leq k} \|\mathbf{W} \odot (\mathbf{X} - \mathbf{M})\|_F^2, \quad (1)$$

¹This work was supported by the Swedish Research Council (grants no. 2015-05639, 2016-04445 and 2018-05375), the strategic research project ELLIIT, the Swedish Foundation for Strategic Research (Semantic Mapping and Visual Navigation for Smart Robots) and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

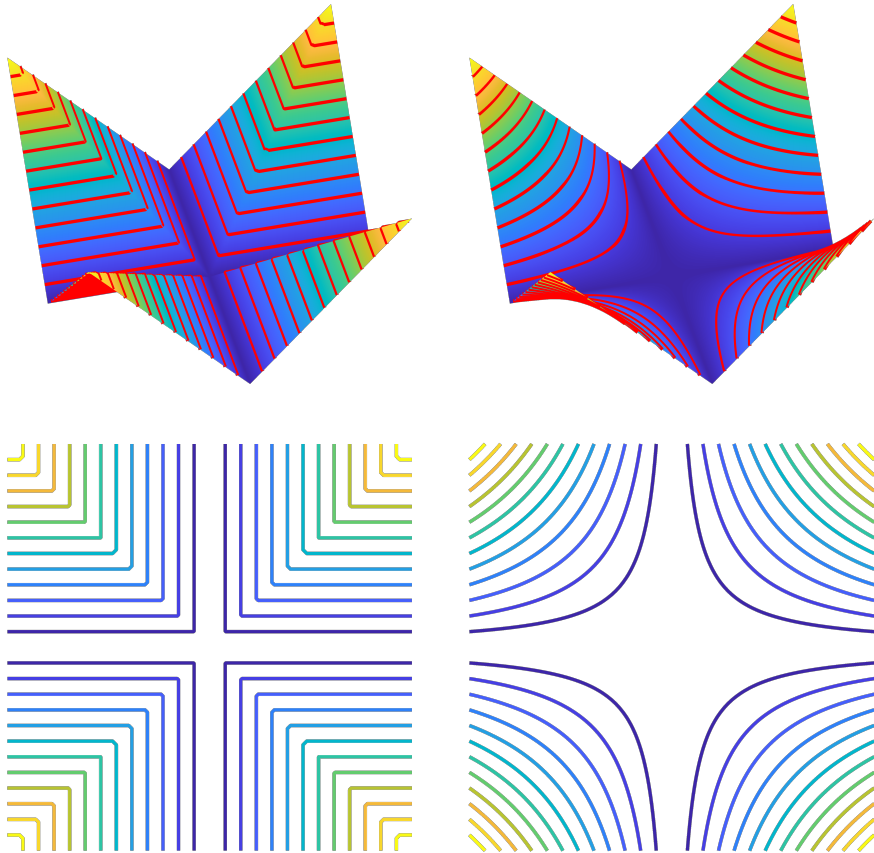


Figure 1: *Top row:* Weighted nuclear norm penalty (left) and the corresponding relaxation r_h , for $a_1 = 0$ and $a_2 = 1$, considered in this paper. *Bottom row:* Level sets, corresponding to the red lines in the top images.

is a core step in many structure from motion formulations [6]. Here M is a measurement matrix which is only partially known and W is a binary matrix removing residuals corresponding to unknown elements. The traditional approach, which is typically used when the rank of the sought matrix is known, enforces a particular rank by restricting the number of columns of the factors B and C and searches over the bilinear parameterization of the unknown matrix $X = BC^T$. Since the resulting objective is a least squares problem in both B and C , alternating updates of B and C can be used. While being extremely simple, this approach has been shown to be prone to “flatlining: requiring excessive

numbers of iterations before convergence” [6]. Instead [6] proposed a damped newton approach and empirically verified that this outperforms the alternation approach. In a number of recent papers Hong *et al.* [17–20] showed that the so called VarPro method is remarkably resilient to local minima. For example [19] reports convergence to the best solution from random initialization in 94% of the cases on the dinosaur sequence which is an admittedly difficult dataset with 77% missing data.

An alternative approach is to optimize directly over the elements of \mathbf{X} while applying penalties to the singular values. This is typically applied to problems of the more general class

$$\min_{\mathbf{X}} p(\sigma(\mathbf{X})) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2. \quad (2)$$

Here p is some penalty function encouraging a desired distribution of singular values $\sigma_i(\mathbf{X})$ of the matrix \mathbf{X} , see *e.g.* [21, 27, 29]. This way of directly optimizing over the elements of \mathbf{X} has been made popular by the work on nuclear norms [8, 34] and their generalizations [12–14, 26, 28], which has shown that with an appropriate choice of regularizer (2) can be made convex. While convex regularizers can be sufficient for applications such as image restoration, where a relatively high rank is acceptable, they are typically rather weak and do not give solutions with low enough rank for structure from motion problems. Consequently, they have to be combined with thresholding schemes to generate satisfactory solutions [7, 11].

To achieve better results, non-convex penalty functions [9, 16, 25, 29] are also frequently used in (2). Since these formulations are typically not differentiable, optimization relies on splitting methods such as ADMM [4]. These are essentially first order methods and, as such, convergence near the minimum can be slow. Indeed, [4] recommends to use these when an approximate solution is sufficient, but suggests to switch to second order methods when accuracy is needed.

In this paper, we derive such second order methods for a general class of objectives of the form (2). Our class covers commonly used regularizers, such as weighted nuclear norms, soft rank penalties and hard rank constraints. Note that these functions can be both non-convex and discontinuous. We show how to reformulate these into bilinear objectives, that can be accurately approximated with quadratic functions, allowing rapid convergence with second order methods such as VarPro or Levenberg–Marquardt.

1.1 Framework and Contributions

In this paper we consider a general framework of non-separable objectives the form

$$f_h(\mathbf{X}) = h(\sigma(\mathbf{X})) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2, \quad (3)$$

where

$$h(\sigma(\mathbf{X})) = \sum_{i=1}^{\text{rank}(\mathbf{X})} a_i \sigma_i(\mathbf{X}) + b_i. \quad (4)$$

Here the sequences $(a_i)_{i=1}^k$ and $(b_i)_{i=1}^k$ are both assumed to be non-decreasing. For different choices of a and b the general regularizer h reduces to commonly used singular value penalties. For example, with $b_i = \mu$ and $a_i \equiv 0$ we get the soft rank penalty $\mu \text{rank}(\mathbf{X})$, with $a_i \equiv 0$ if $i \leq k$ and ∞ otherwise, we get the hard constraint $\text{rank}(\mathbf{X}) \leq k$. With $b_i \equiv 0$, we get the weighted nuclear norm $\sum_i a_i \sigma_i(\mathbf{X})$, but the framework is large and several other regularizers are possible. We aim to optimize objectives including all such regularizers using second order methods, such as VarPro. This requires finding a good approximation—which is two times differentiable—of the objective function. For this purpose, we propose to use a relaxation $r_h(\sigma)$ of $h(\sigma)$ developed in [40] and consider

$$r_h(\sigma(\mathbf{X})) + \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2. \quad (5)$$

This results in a continuous and almost everywhere differentiable objective (see Section 2 for details, Figures 1 and 2 for examples). When introducing the terms $\gamma_i(\mathbf{B}, \mathbf{C}) = \frac{(\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2)}{2}$, where \mathbf{B}_i and \mathbf{C}_i are columns i of \mathbf{B} and \mathbf{C} , respectively, we obtain the bilinear formulation

$$\min_{\mathbf{B}, \mathbf{C}} r_h(\gamma(\mathbf{B}, \mathbf{C})) + \|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2. \quad (6)$$

The main contributions of this paper are:

- i) We show that (5) is equivalent to (6) by proving that

$$r_h(\sigma(\mathbf{X})) = \min_{\mathbf{X}=\mathbf{B}\mathbf{C}^T} r_h(\gamma(\mathbf{B}, \mathbf{C})), \quad (7)$$

see Theorem 1. Furthermore, if $\|\mathcal{A}\| < 1$ then the relaxation (5) is guaranteed to have the same global optimizers as the original (3), see [10].

- ii) We show that (6) can be accurately approximated by quadratic functions opening up the possibility of applying second order methods to the problem.
- iii) We propose a modified VarPro algorithm and show that it provides superior performance for difficult objectives common in computer vision applications.

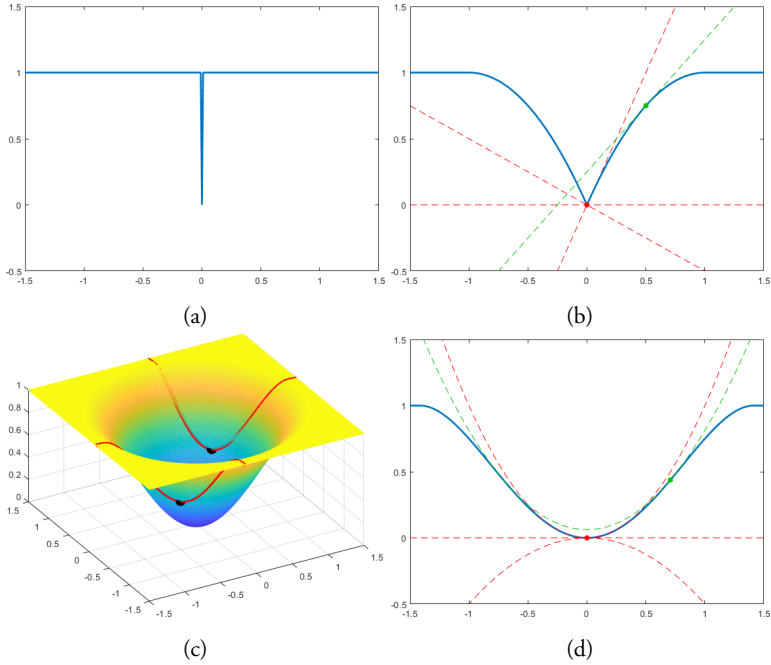


Figure 2: (a): One dimensional example of $h(x)$, with $a_1 = 0$ and $b_1 = 1$. (b): The relaxation $r_h(x)$. Replacing h with r_h makes the objective continuous and differentiable everywhere except in $x = 0$. In this point r_h has several subgradients (red lines) while on any other point has its unique tangent (e.g. green line). (c): The two dimensional function $r_h(\frac{b^2+c^2}{2})$. Parametrizing with squared variables smooths the function around around $(b, c) = (0, 0)$ giving an objective that can be locally well approximated with quadratic functions. (d): The function from (c), sliced along $c = 0$, with the quadratic approximations resulting from the subgradients shown in (b).

1.2 Related Work

There are some works that have previously proposed bilinear formulations for problems of the form (2). It was observed in [34] that

$$\|X\|_* = \min_{BC^T=X} \frac{\|B\|_F^2 + \|C\|_F^2}{2}. \quad (8)$$

Thus, when $p(\sigma(X)) = \|X\|_*$ optimization of (2) can be formulated as

$$\min_{B,C} \mu \frac{\|B\|_F^2 + \|C\|_F^2}{2} + \|ABC^T - \mathbf{b}\|^2. \quad (9)$$

Interestingly, even though (9) is non-convex, it can be shown that any local minimizer \mathbf{B}, \mathbf{C} with $\text{rank}(\mathbf{BC}^T) < k$, where k is the number of columns in \mathbf{B} and \mathbf{C} , is globally optimal [1, 15]. In addition, the objective function is smooth and second order methods can be employed.

A similar formulation was tested for structure from motion and photometric stereo in [7]. In practice, it was, however, observed that the nuclear norm is too weak to give low rank solutions when the data is noisy. Therefore, a ‘‘continuation’’ approach, where the size of the factorization is gradually reduced, is proposed. In [15] the above results were extended to a general class of 2-homogeneous factor penalties $\theta(\mathbf{B}, \mathbf{C})$. Interestingly, this formulation allows to add constraints such as non-negativity to the factors \mathbf{B} and \mathbf{C} . Their results show that $\min_{\mathbf{BC}^T=\mathbf{X}} \theta(\mathbf{B}, \mathbf{C})$ is equivalent to a convex regularization function $\Omega(\mathbf{X})$. The property that a local minimizer \mathbf{B}, \mathbf{C} with $\text{rank}(\mathbf{BC}^T) < k$ is global, is also extended to this class of functions.

Similar approaches to non-convex formulations have also been proposed. Shang *et al.*[36] showed that penalization with the Schatten semi-norms $\|\mathbf{X}\|_q = \sqrt[q]{\sum_{i=1}^N \sigma_i(\mathbf{X})^q}$, for $q = 1/2$ and $2/3$, can be achieved using a convex penalty on the factors \mathbf{B} and \mathbf{C} . A generalization to other values of q is given in [44]. A separable regularizer of the form $p(\sigma(\mathbf{X})) = \sum_{i=1}^n \tilde{p}(\sigma_i(\mathbf{X}))$ was considered in [41]. It was shown that when $\tilde{p}(\sigma)$ is concave and non-decreasing on $\sigma \geq 0$ then

$$p(\sigma(\mathbf{X})) = \min_{\mathbf{BC}^T=\mathbf{X}} \sum_{i=1}^n \tilde{p}(\gamma_i(\mathbf{B}, \mathbf{C})). \quad (10)$$

Separable penalties, such as the once mentioned above, are limited in the sense that they only consider the singular values separately. The penalty can therefore only be based on the magnitude of the singular values. As a consequence, they cannot count the number of non-zero singular values, thus making it impossible to penalize all matrices with rank larger than a predefined threshold. A non-separable formulation has very recently been addressed in [22]. Here (10) was generalized to weighted nuclear norms, that is $p(\sigma(\mathbf{X})) = \sum_{i=1}^n a_i \sigma_i(\mathbf{X})$. Note that, in contrast to the original nuclear norm, these are not convex since they are able to penalize smaller singular values harder (when $(a_i)_{i=1}^n$ is increasing).

One of the main benefits of our framework is that there are provably fewer local minima [10], compared to using the unrelaxed penalty used in [22]. Valtonen Örn-hag *et al.* [33] considered a special case of our framework, where the rank is known a priori. They did, however, not show equivalence between the proposed bilinear regularizer and the corresponding original formulation.

2 Overview of the Approach

In this section we present our general framework and explain and motivate our algorithmic approach.

2.1 A Continuous Relaxation

Objective (3) is typically discontinuous, and, therefore, local approximation (linear or quadratic) is not directly feasible. To circumvent this issue, we consider the quadratic envelop [10] $r_h(\mathbf{x})$ of $h(\mathbf{x})$

$$r_h(\mathbf{x}) = \left(h(\mathbf{x}) + \|\mathbf{x}\|^2 \right)^{**} - \|\mathbf{x}\|^2, \quad (11)$$

that is, we add a quadratic term to h , compute the convex envelope of the result, and subtract the quadratic term. Throughout the paper we will use the function $g(\mathbf{x}) := r_h(\mathbf{x}) - \|\mathbf{x}\|^2 = \left(h(\mathbf{x}) + \|\mathbf{x}\|^2 \right)^{**}$. In general, r_h and g do not have closed form expressions, but are obtained from

$$g(\mathbf{x}) = \max_{\mathbf{z} \in \mathbb{R}^n} \left(2\langle \mathbf{x}, \mathbf{z} \rangle - \sum_{i=1}^n \left[\left[|z_{[i]}| - a_i \right]_+^2 - b_i \right]_+ \right), \quad (12)$$

where $[\cdot]_+ := \max(\cdot, 0)$ and $z_{[i]}$ denotes the element with the i :th largest magnitude in \mathbf{z} . The optimization over \mathbf{z} is convex and can be solved efficiently, as outlined in [40].

Note that in (3) the function h always takes sorted non-negative vectors $\sigma(\mathbf{X})$. For a general vector \mathbf{x} we therefore think of h as being permutation and sign invariant, that is

$$h(\mathbf{x}) = \sum_{i=1}^{\text{card}(\mathbf{x})} a_i |x_{[i]}| + b_i, \quad (13)$$

where $\text{card}(\mathbf{x})$ is the number of non-zero elements in \mathbf{x} . Figure 1 shows an example of r_h and h with $b_i = 0$, *i.e.* $h(\sigma(\mathbf{X}))$ is a weighted nuclear norm of \mathbf{X} . Both functions are permutation invariant; however, r_h is additionally continuous and differentiable almost everywhere, except where one of the variables is equal to zero. Figure 2a shows a one dimensional example where $a_1 = 0$ and $b_1 = 1$. In this case the function h is discontinuous, which is generally the case when any of the b_i variables are non-zero. Figure 2b shows the relaxation r_h for the choice of h in 2a.

Replacing h with r_h gives us the relaxation (5) which in terms of regularity is significantly better behaved than (3). There are, of course, many ways of approximating the h function. The reason for choosing this particular relaxation is that it can be shown [10]

that if $\|\mathcal{A}\| < 1$ it has the same global minimizer as (3). In addition it has been shown that if RIP [34] holds, then, under moderate noise, (5) only has one stationary point for some choices of h [30, 31].

2.2 Bilinear Parameterization

The regularity of our formulation improves even further when we introduce the bilinear terms $\gamma_i(\mathbf{B}, \mathbf{C}) = \frac{(\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2)}{2}$ and consider (6). Here we have replaced \mathbf{X} with a factorization $\mathbf{B}\mathbf{C}^T$ and instead of penalizing the singular values of \mathbf{X} we penalize the vector $\gamma(\mathbf{B}, \mathbf{C})$ containing the elements $\gamma_i(\mathbf{B}, \mathbf{C})$. Note that the regularizer $r_h(\gamma(\mathbf{B}, \mathbf{C}))$ depends on which particular factorization $\mathbf{B}\mathbf{C}^T = \mathbf{X}$ of \mathbf{X} that we chose. The main theoretical result of this paper states that for any fixed matrix \mathbf{X} we have

$$\min_{\mathbf{B}\mathbf{C}^T = \mathbf{X}} r_h(\gamma(\mathbf{B}, \mathbf{C})) = r_h(\sigma(\mathbf{X})), \quad (14)$$

hence, minimization over factorizations $\mathbf{B}\mathbf{C}^T$ of \mathbf{X} will result in the same penalty as the singular value vector $\sigma(\mathbf{X})$. Furthermore, since the second term $\|\mathcal{A}(\mathbf{B}\mathbf{C}^T) - \mathbf{b}\|^2 = \|\mathcal{A}\mathbf{X} - \mathbf{b}\|^2$, regardless of which factorization we choose, it is clear that minimization of (6) is equivalent to (5).

Figure 2c shows $r_h\left(\frac{b^2+c^2}{2}\right)$ as a function of (b, c) . Note that introducing the squared variables makes the resulting function smooth at $(b, c) = (0, 0)$. To illustrate our main result, we also plot red curves on the surface corresponding to all points where $bc = 0.25$. While the regularization term can take many values over $bc = 0.25$, the two minimizers $(b, c) = \pm(0.5, 0.5)$, shown as black dots, both give the value $r_h\left(\frac{0.5^2+0.5^2}{2}\right) = r_h(0.25)$.

2.3 Quadratic Approximation and Optimization

The basis for our algorithm is the ability to accurately approximate $r_h(\gamma(\mathbf{B}, \mathbf{C}))$ with a quadratic relaxation. The principle can be illustrated by considering a general smooth function $r(\gamma)$ with the first order Taylor approximation around $\boldsymbol{\eta}$

$$r(\gamma) \approx r(\boldsymbol{\eta}) + \langle \nabla r(\boldsymbol{\eta}), \gamma - \boldsymbol{\eta} \rangle. \quad (15)$$

When inserting $\gamma = \gamma(\mathbf{B}, \mathbf{C})$ we obtain (ignoring the constants $r(\boldsymbol{\eta})$ and $-\langle \nabla r(\boldsymbol{\eta}), \boldsymbol{\eta} \rangle$) the quadratic approximation

$$\langle \nabla r(\boldsymbol{\eta}), \gamma(\mathbf{B}, \mathbf{C}) \rangle = \sum_{i=1}^n r'_i(\boldsymbol{\eta}) \frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2}, \quad (16)$$

Table 1: Log-loss for the Door, Back, Heart, and Paper datasets, for $\eta = 0.05$ and $K = 8$, for R_h for Nuclear Norm (NN) regularization, Weighted Nuclear Norm (WNN) regularization, and with linearly-increasing weights (LI) and singular value based (SV) weights. For our proposed method, we also show in red the relative loss improvement compared to the ADMM solution, *i.e.* $100 \times \frac{\text{loss}_{\text{ADMM}} - \text{loss}_{\text{Ours}}}{\text{loss}_{\text{ADMM}}}$.

	Method	NN	WNN	LI	SV
Cathedral (Rank 4)	ADMM	-0.4528	-2.4548	—	—
	Ours	-0.4528	-3.8110	—	—
			(95.6%)		
Door (Rank 4)	ADMM	-1.1946	-1.6247	—	—
	Ours	-1.1946	-4.6275	—	—
			(99.9%)		
École (Rank 4)	ADMM	-0.8747	-1.6869	—	—
	Ours	-0.8747	-4.4264	—	—
			(99.8%)		
Back (Rank 7)	ADMM	-1.3495	-4.8123	-5.2560	-5.2133
	Ours	-1.3536	-4.8129	-5.2863	-5.2471
		(0.93%)	(0.14%)	(6.74%)	(7.49%)
Heart (Rank 7)	ADMM	-1.3166	-4.1767	-5.0829	-5.0182
	Ours	-1.3338	-4.1783	-5.1153	-5.0426
		(3.88%)	(0.36%)	(7.19%)	(5.46%)
Paper (Rank 7)	ADMM	-1.3298	-5.3847	-5.8484	-5.8509
	Ours	-1.3839	-5.3910	-5.8964	-5.9023
		(11.71%)	(1.44%)	(10.46%)	(11.17%)

where r'_i is the partial derivative with respect to the i :th entry of r . Since our particular regularizer r_h is not differentiable everywhere (when parametrized with \mathbf{x}) we make use of the so called subdifferential $\partial r_h(\mathbf{x})$ of r_h . For a convex function g we have $2\mathbf{z} \in \partial g(\mathbf{x})$ if and only if

$$g(\mathbf{y}) \geq g(\mathbf{x}) + \langle 2\mathbf{z}, \mathbf{y} - \mathbf{x} \rangle, \quad (17)$$

for all \mathbf{y} . For the g defined in (12) it can be shown that (17) holds if and only if \mathbf{z} is a minimizer in (12). At a point where the function is differentiable the gradient is the only element in the subdifferential, and the right hand side of (17) is the Taylor approximation. However, in general it can contain several vectors and these can be seen as lower bounding linear approximations, as in (17).

If, as in our case, $r_h(\mathbf{x}) = g(\mathbf{x}) - \|\mathbf{x}\|^2$ we have that $2(\mathbf{z} - \mathbf{x}) \in \partial r_h(\mathbf{x})$ when $2\mathbf{z} \in \partial g(\mathbf{x})$. For a non-convex function the above inequality becomes approximate around \mathbf{x} (up to higher order terms). Figure 2b shows examples of subgradients at two points. At $\mathbf{x} = 0$ (red point) where the function is not differentiable there are several such linear bounds, while at $\mathbf{x} = 0.5$ there is only one tangent. For our class of functions, an element of \mathbf{z}_i is only non-unique when $x_i = 0$.

Suppose now that we want to approximate $r_h(\gamma(\mathbf{B}, \mathbf{C}))$ around a point $\boldsymbol{\eta} = \gamma(\bar{\mathbf{B}}, \bar{\mathbf{C}})$. Up to higher order terms, we then have

$$r_h(\boldsymbol{\gamma}) \geq r_h(\boldsymbol{\eta}) - 2\langle \mathbf{z} - \boldsymbol{\eta}, \boldsymbol{\eta} \rangle + 2 \sum_{i=1}^n (z_i - \eta_i) \frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2}. \quad (18)$$

Since $2\mathbf{z} \in \partial g(\boldsymbol{\eta})$ it can be shown that z_i can only take one value when $\eta_i \neq 0$, see also Figure 2 (we also give a proof of this in the supplementary material). Therefore the terms $r_h(\boldsymbol{\eta})$ and $\langle \mathbf{z} - \boldsymbol{\eta}, \boldsymbol{\eta} \rangle$ are constants. In addition, it can be shown that for any i where $\eta_i = 0$, z_i can be chosen freely as long as its magnitude is smaller than a certain number M_i . Since $\frac{(\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2)}{2}$ is non-negative it is clear that selecting z_i as large as possible gives a vector \mathbf{z} that maximizes the right hand side of (18) for all possible (\mathbf{B}, \mathbf{C}) . At iteration t we therefore use

$$r_h^{(t)}(\gamma(\mathbf{B}, \mathbf{C})) \approx \sum_{i=1}^n w_i^{(t)} \frac{\|\mathbf{B}_i\|^2 + \|\mathbf{C}_i\|^2}{2} \quad (19)$$

where $w_i^{(t)} = 2(z_i - \eta_i)$ and $\mathbf{z} \in \partial g(\boldsymbol{\eta})$ with $z_i = M_i$ when $\eta_i = 0$. Figure 2d shows $r_h(\frac{b^2}{2})$ and the quadratic approximations obtained from the subgradients plotted in 2b. Note that, in contrast to 2b, where we would have to use several subgradients to approximate the functions behaviour around $x = 0$, in 2d it is enough with the one that corresponds to the largest permissible value of z .

2.4 Overview of Algorithm

In this section we give a rough overview of the algorithm we propose (a detailed description is given in the supplementary material). Our algorithm is based on the Variable Projection (VarPro) approach which has been shown to be highly efficient for computer vision problems [17–20]. Our approximation

$$r_h^{(t)}(\gamma(\mathbf{B}, \mathbf{C})) + \|\mathcal{A}(\mathbf{B}\mathbf{C})^T - \mathbf{b}\|^2, \quad (20)$$

is bi-quadratic meaning that given \mathbf{B} we can solve for \mathbf{C} in closed form (and vice versa). The optimal $\mathbf{C}^*(\mathbf{B})$ as a function of the unknown \mathbf{B} can now be inserted back into (20) to give an objective in \mathbf{B} alone. VarPro essentially optimizes this new objective locally using a damped Gauss-Newton approach. This is the core routine of our algorithm which consists of the following four main steps:

1. Given $(\mathbf{B}^{(t)}, \mathbf{C}^{(t)})$ compute the maximal subgradient $\mathbf{z} \in \partial g(\gamma(\mathbf{B}^{(t)}, \mathbf{C}^{(t)}))$, using the algorithm proposed in [40].

2. Compute the approximation $r_h^{(t)}(\gamma(\mathbf{B}, \mathbf{C}))$, as in (19).
3. Run one iteration of VarPro on (20) to obtain $(\mathbf{B}^{(t+1)}, \mathbf{C}^{(t+1)})$.
4. Optional: Compute the SVD $\mathbf{X}^{(t+1)} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{X}^{(t+1)} = \mathbf{B}^{(t+1)}(\mathbf{C}^{(t+1)})^T$, and set

$$\begin{aligned} \mathbf{B}^{(t+1)} &:= U\sqrt{\mathbf{\Sigma}}, \\ \mathbf{C}^{(t+1)} &:= V\sqrt{\mathbf{\Sigma}}. \end{aligned} \tag{21}$$

The two first steps give the approximation which is used in VarPro in the third step. The fourth step is optional and is added to help avoid local minima that may occur when $a_i = 0$. In this case the regularizer is typically constant above a certain threshold, as in Figure 2a, and in such cases one may get stuck in suboptimal factorizations. Empirically, we have found that when $a_i \neq 0$ the SVD step can be omitted.

3 Main Theoretical Result

In this section we give the main technical result that makes our algorithmic approach possible.

Theorem 1. *If the sequences $(a_i)_{i=1}^n$ and $(b_i)_{i=1}^n$ are non-negative and non-decreasing then*

$$r_h(\sigma(\mathbf{X})) = \min_{\mathbf{X}=\mathbf{B}\mathbf{C}^T} r_h(\gamma(\mathbf{B}, \mathbf{C})). \tag{22}$$

The proof builds on the results of [22] which establishes a similar result but with the function $r_h(\sigma(x))$ replaced by weighted nuclear norm penalties $\mathbf{v}^T \sigma(\mathbf{X})$, where the elements of \mathbf{v} are non-negative and increasing.

3.1 Theoretical Background

Before we proceed to the proof we recall some of the theory from [22]. Here the optimization problem on the right hand side of (22) was studied by writing $\mathbf{X} = \mathbf{B}\mathbf{V}\mathbf{H}^T\mathbf{C}^T$ and varying \mathbf{V} and \mathbf{H} such that $\mathbf{V}\mathbf{H}^T = \mathbf{I}$. It was shown that the an equivalent formulation is

$$\begin{aligned} \min_{\gamma, \mathbf{M} \in \mathcal{S}} \quad & r_h(\gamma) \\ \text{s.t.} \quad & \gamma = \mathbf{M}\sigma, \end{aligned} \tag{23}$$

where

$$\mathcal{S} = \left\{ \frac{1}{2}(\mathbf{V}^T \odot \mathbf{V}^T + \mathbf{H}^T \odot \mathbf{H}^T) \mid \mathbf{V}\mathbf{H}^T = \mathbf{I} \right\}. \quad (24)$$

The set \mathcal{S} is difficult to handle since it is non-convex, but it turns out that it is contained in the set of doubly superstochastic matrices \mathcal{S}_W [3] which is convex. Moreover, the extreme points of \mathcal{S}_W are the permutation matrices which are also contained in \mathcal{S} . Therefore optimization of a linear function (weighted nuclear norm) $\mathbf{v}^T \boldsymbol{\gamma}$ over \mathcal{S} is equivalent to optimization over the convex relaxation \mathcal{S}_W . Moreover, their results show that if the elements v_i of \mathbf{v} are (strictly) increasing the unique global minimizer is $\mathbf{M} = \mathbf{I}$.

3.2 Proof of Theorem 1

Our proof relies on the observation that local approximations of $r_h(\boldsymbol{\gamma})$ are equivalent to weighted nuclear norms with coefficients $v_i = 2(z_i - \gamma_i)$, where $2\mathbf{z} \in \partial g(\boldsymbol{\gamma})$. We will show that these elements are increasing and invoke the results of [22]. For this purpose we need some knowledge about the subgradients \mathbf{z} at $\boldsymbol{\gamma}$. Note that since $\boldsymbol{\gamma}(\mathbf{B}, \mathbf{C})$ has non-negative elements we can assume the same for \mathbf{z} . Furthermore, because of the permutation invariance of r_h , we also assume that $\boldsymbol{\gamma}$ and \mathbf{z} are non-increasing.

Let $s_i = a_i + \max\{\gamma_i, \sqrt{b_i}\}$. The results of [40] show that if $2\mathbf{z} \in \partial g(\boldsymbol{\gamma})$. Then \mathbf{z}_i^* fulfills

$$\mathbf{z}_i^* = \begin{cases} s_i & s_i \in [z_{i+1}, z_{i-1}] \\ z_{i-1} & s_i \geq z_{i-1} \\ z_{i+1} & s_i \leq z_{i+1} \end{cases} \quad (25)$$

for all i where $\gamma_i \neq 0$. Note also that the values of these element are the same for all \mathbf{z} with $2\mathbf{z} \in \partial g(\boldsymbol{\gamma})$. For any i where $\gamma_i = 0$, the element z_i can take any value in $[0, s_i]$ as long as the elements of \mathbf{z} are non-increasing. In our case we will use the vector \mathbf{z} that has the largest possible elements and these fulfill (25) for all i . We will now present a fundamental property related to these sequences. A detailed proof is available in the supplementary material.

Lemma 1. *If $\mathbf{z}^* = \operatorname{argmax}_{2\mathbf{z} \in \partial g(\boldsymbol{\gamma})} \mathbf{v}^T \mathbf{z}$, where $v_i > 0$ for all $i = 1, \dots, n$ then the sequence $(z_i^* - \gamma_i)_{i=1}^n$ is non-decreasing.*

We now come to the main result.

Proof of Theorem 1. We consider the relaxation of problem (23), namely

$$\begin{aligned} \min_{\boldsymbol{\gamma}, \mathbf{M} \in \mathcal{S}_W} \quad & r_h(\boldsymbol{\gamma}) \\ \text{s.t.} \quad & \boldsymbol{\gamma} = \mathbf{M}\boldsymbol{\sigma}. \end{aligned} \quad (26)$$

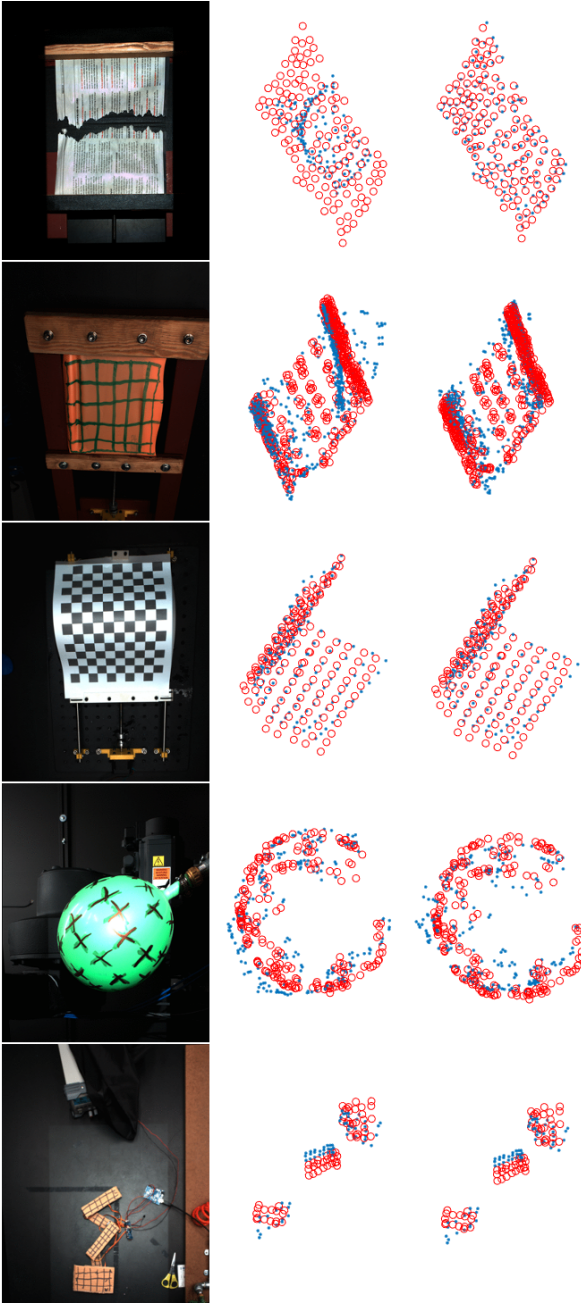


Figure 3: Qualitative results on the (left to right) Articulated, Balloon, Paper, Stretch, and Tearing datasets of the NRSfM Challenge. In red we show the provided ground-truth structure for one of the frames of the sequences. In blue we show the obtained 3D structure for the same frame using ADMM (top) and the proposed method (bottom).

We will show that minimization of $r_h(\gamma)$ over the set $\mathcal{S}_{\mathcal{W}}$ is achieved when $\gamma = \sigma$. Since $\sigma = \mathbf{I}\sigma$ and $\mathbf{I} \in \mathcal{S}$ this will also be the solution to the original problem.

For simplicity we will first consider the function $\tilde{r}_h(\gamma) = r_h(\gamma) + \langle 2\mathbf{w}, \gamma \rangle$, where $\mathbf{w} = (\varepsilon, 2\varepsilon, 3\varepsilon, \dots)$ for $\varepsilon > 0$. The directional derivatives of this function are given by

$$\begin{aligned} (\tilde{r}_h)'_d(\gamma) &= g'_d(\gamma) - \langle 2\gamma, \mathbf{d} \rangle + \langle 2\mathbf{w}, \mathbf{d} \rangle \\ &= \max_{2\mathbf{z} \in \partial g(\gamma)} \langle 2(\mathbf{z} - \gamma + \mathbf{w}), \mathbf{d} \rangle. \end{aligned} \quad (27)$$

As previously noted, g is convex, which guarantees the existence of the directional derivative. Now consider a point $\gamma^* = M\sigma$, where in $M \in \mathcal{S}_{\mathcal{W}}$. Since r_h is invariant to permutations we may assume that the elements of γ^* are non-increasing. We let $2\mathbf{z}^* \in \partial g(\gamma^*)$ with the maximal elements as in Lemma 1. This makes the sequence $z_i^* - \gamma_i^*$ non-decreasing. Therefore the sequence $z_i^* - \gamma_i^* + w_i$ will be strictly increasing. The results of [22] now show that the (unique) minimum of

$$\min_{\eta, M \in \mathcal{S}_{\mathcal{W}}} \langle \mathbf{z}^* - \gamma^* + \mathbf{w}, \eta \rangle, \quad (28)$$

$$\text{s.t.} \quad \eta = M\sigma, \quad (29)$$

is given by $\eta = \mathbf{I}\sigma$. By selecting $\mathbf{d} = \sigma - \gamma^*$ we see that there is a direction such that

$$\langle \mathbf{z}^* - \gamma^* + \mathbf{w}, \mathbf{d} \rangle = \langle \mathbf{z}^* - \gamma^* + \mathbf{w}, \sigma \rangle - \langle \mathbf{z}^* - \gamma^* + \mathbf{w}, \gamma^* \rangle < 0. \quad (30)$$

Furthermore, for any other choice $\mathbf{z} \in \partial g(\gamma^*)$ we have $\langle \mathbf{z}^*, \sigma \rangle \geq \langle \mathbf{z}, \sigma \rangle$ since the elements of σ are non-negative and $\langle \mathbf{z}^*, \gamma^* \rangle = \langle \mathbf{z}, \gamma^* \rangle$ since $z_i^* = z_i$ when $\gamma_i \neq 0$. Therefore

$$\langle \mathbf{z}^* - \gamma^* + \mathbf{w}, \mathbf{d} \rangle \geq \langle \mathbf{z} - \gamma^* + \mathbf{w}, \mathbf{d} \rangle, \quad (31)$$

which shows that \mathbf{z}^* is the maximizer in (27).

We can thus conclude that, as long as $\gamma^* \neq \sigma$, there is a direction (feasible in $\mathcal{S}_{\mathcal{W}}$) with strictly negative directional derivative. Hence σ is the minimizer of $\tilde{r}_h(\gamma)$ over $\mathcal{S}_{\mathcal{W}}$. Since $\mathbf{I} \in \mathcal{S}$ it also optimizes $\tilde{r}_h(\gamma)$ over \mathcal{S} for any $\varepsilon > 0$.

We now let $\varepsilon \rightarrow 0$ to show that σ is a (not necessarily unique) minimizer of (26). To see this we note that since $\langle \mathbf{w}, \gamma \rangle > 0$ for all feasible γ we have

$$r_h(\sigma) < r_h(\sigma) + 2\langle \mathbf{w}, \sigma \rangle \leq r_h(\gamma) + 2\langle \mathbf{w}, \gamma \rangle. \quad (32)$$

Taking the (pointwise) limit of the right hand side shows that $r_h(\sigma) \leq r_h(\gamma)$. \square

Table 2: Average log-loss and reconstruction errors (mm) on each dataset over the 6 camera paths relatively to the provided ground-truth structure.

		Artic.	Balloon	Paper	Stretch	Tearing
Log-loss	ADMM	-2.221	-2.529	-2.338	-2.395	-1.471
	Ours	-2.415	-2.657	-2.560	-2.622	-2.053
Rec. error	ADMM	14.55	9.29	6.95	7.83	29.90
	Ours	16.10	8.29	6.70	7.66	11.26

4 Experiments

4.1 Matrix Recovery

In this section, the convergence and accuracy of the proposed method, using bilinear formulation, are compared to the ADMM approach [40]. For the data term, we use the pseudo Object Space Error (pOSE) defined in Hong *et al.* [18], which fuse the Object Space Error (OSE)

$$\ell_{\text{OSE}} := \sum_{(i,j) \in \Omega} \|\mathbf{P}_{i,1:2}\tilde{\mathbf{x}}_j - (\mathbf{p}_{i,3}^T\tilde{\mathbf{x}}_j)\mathbf{m}_{i,j}\|_2^2, \quad (33)$$

and the error of an affine camera model

$$\ell_{\text{Affine}} := \sum_{(i,j) \in \Omega} \|\mathbf{P}_{i,1:2}\tilde{\mathbf{x}}_j - \mathbf{m}_{i,j}\|_2^2, \quad (34)$$

where $\mathbf{P}_{i,1:2}$ and $\mathbf{p}_{i,3}$ are, respectively, the first two and the third rows of the camera matrix \mathbf{P}_i , with $i = 1, \dots, F$. Furthermore, $\tilde{\mathbf{x}}_j$ is a 3D point in homogeneous coordinates, with $j = 1, \dots, P$, and $\mathbf{m}_{i,j}$ is the 2D observation of the j :th point on the i :th camera. The set of observable data is denoted Ω . The two terms are weighted by $\eta \in [0, 1]$, resulting in the loss

$$\ell_{\text{pOSE}} := (1 - \eta)\ell_{\text{OSE}} + \eta\ell_{\text{Affine}}. \quad (35)$$

In Iglesias *et al.* [22], the possibility of extending this framework to non-rigid structure from motion is described by replacing $\mathbf{P}_i\tilde{\mathbf{x}}_j$ by a linear combination of K shape basis, *i.e.* $\mathbf{\Pi}_i\hat{\mathbf{S}}_j$. These factors are structured as $\mathbf{\Pi}_i = [c_{i,1}\mathbf{R}_i \ \dots \ c_{i,K}\mathbf{R}_i \ \mathbf{t}_i] \in \mathbb{R}^{3 \times (3K+1)}$ and $\hat{\mathbf{S}}_j = [S_{1,j}^T \ \dots \ S_{K,j}^T \ 1]^T \in \mathbb{R}^{3K+1}$. The pOSE objective can now be written in a more compact form as

$$\ell_{\text{pOSE}}(\mathbf{X}) = \|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2, \quad (36)$$

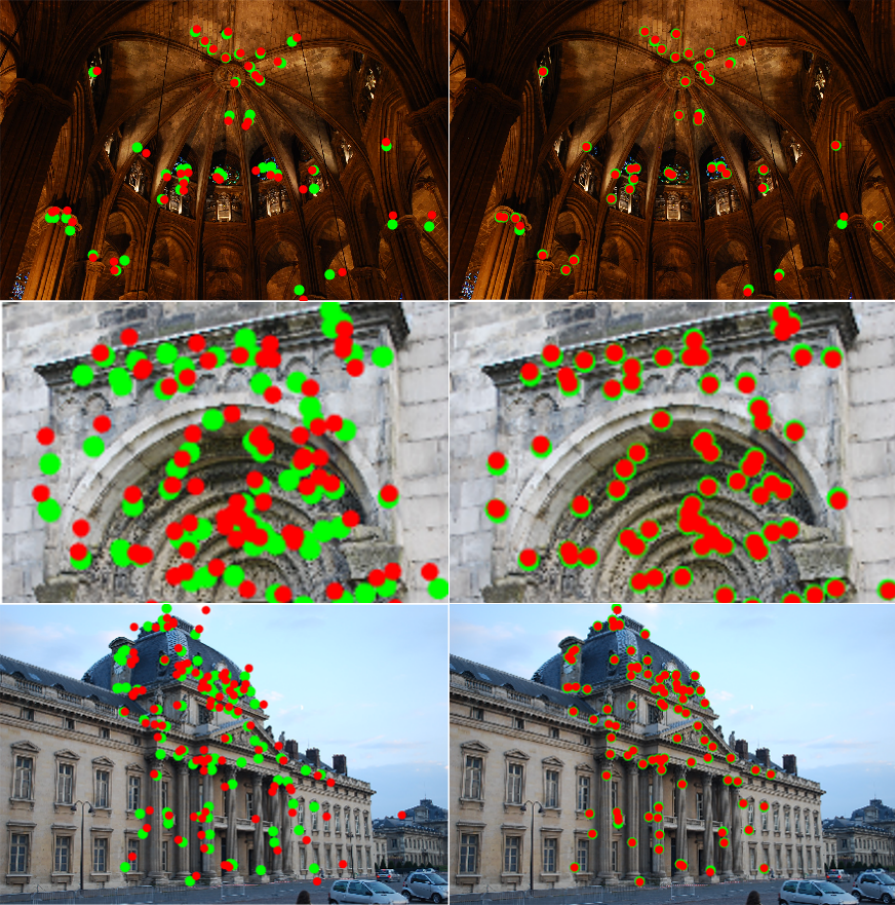


Figure 4: Qualitative results for the Cathedral, Door, and École datasets. In green we show the input image measurements, and in red the reprojected image points. (*Left*): ADMM, and (*Right*): Our method.

where $\mathbf{X} = \mathbf{\Pi}\hat{\mathbf{S}}$, and $\mathbf{\Pi}$ and $\hat{\mathbf{S}}$ are the vertical and horizontal concatenations of $\mathbf{\Pi}_i$ and $\hat{\mathbf{S}}_j$, respectively. For the ADMM formulation we consider the optimization problem

$$\min_{\mathbf{X}} \mathcal{R}_h(\mathbf{X}) + \ell_{\text{pOSE}}(\mathbf{X}), \quad (37)$$

while, for our proposed method, we have the equivalent problem

$$\min_{\mathbf{B}, \mathbf{C}} \tilde{\mathcal{R}}_h(\mathbf{B}, \mathbf{C}) + \ell_{\text{pOSE}}(\mathbf{B}\mathbf{C}^T). \quad (38)$$

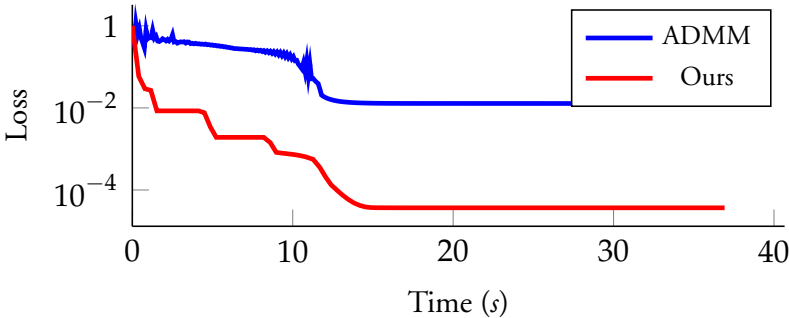


Figure 5: Convergence history for the Door dataset.

The datasets chosen for this experiment are Cathedral, École, and Door [32] (rigid scenes), Back [35], Heart [37], and Paper [42] (deformable objects). We use four different sets of sequences (a_i) and (b_i) : (i) nuclear norm (NN), with $a_i = a_{NN}$, and $b_i = 0$, (ii) weighted nuclear norm (WNN), with $a_i = \frac{a_{WNN}}{(\sigma_i(\mathbf{X}_0) + \delta)}$, and $b_i = 0$, where $\delta > 0$, (iii) linearly increasing (LI) weights with $a_i = b_i = 0, 0 \leq i \leq 4$, and $a_i = 10b_i = a_{LI}(i - 4), i \geq 5$, and (iv) singular value (SV) based sequences $a_i = 2b_i = \frac{a_{SV}}{(\sigma_i(\mathbf{X}_0) + \delta)}$. The singular values $\sigma(\mathbf{X}_0)$ are obtained from the solution \mathbf{X}_0 of $\|\mathcal{A}(\mathbf{X}) - \mathbf{b}\|^2$ (with no regularization). We set $k = 8$ and the parameters for the sequences are chosen such that the methods converge to rank 4 and 7 solutions for the rigid and deformable datasets, respectively. The values of the parameters used to define the sequences for each of the datasets are shown in the supplementary material. For increased stability regarding parameter choice across datasets, we normalize the measurement matrix M to have unit Frobenius norm.

In Table 1 we show the losses obtained by both methods, where the advantages of using our bilinear formulation become clear. In Figure 4 we also show qualitative results of the estimated reprojections on the Cathedral, Door, and École datasets. A convergence plot for the Door dataset is shown in Figure 5, and for the remaining datasets we refer to the supplementary material. The results show the improvement in accuracy that can be obtained by using the bilinear formulation. The difference in terms of loss after convergence, and how it affects the reprojection errors, is more noticeable in the rigid datasets. However, in the next section we show that even a small improvement in the reprojection errors can lead to significantly better 3D reconstructions.

4.2 Application to Non-Rigid SfM

We apply our method to the perspective datasets in the Non-Rigid Structure from Motion Challenge [23]. Each of the five datasets—Articulated, Balloon, Paper, Stretch and

Tearing—have six different sequences, consisting of six different camera paths—circle, flyby, line, semi-circle, tricky and zigzag. To deal with perspective projections, we again use the pseudo Object Space Error $\ell_{\text{pOSE}}(\mathbf{X})$ described in Section 4.1, where \mathbf{X} is now parameterized as $\mathbf{R}\mathcal{U}(\mathbf{C}\mathbf{B}^\sharp) + \mathbf{t}$, with \mathbf{t} being the vector of translations. Similarly to Iglesias *et al.* [22], we assume that the rotations are known, and that the image measurements are calibrated. The rotation matrices are recovered from 2D observations for the orthogonal camera model, while the intrinsic camera matrix is estimated using the provided ground-truth 3D structure for one of the frames of the sequences. The optimization problem can be written as

$$\min_{\mathbf{X}, \mathbf{t}} \mathcal{R}_h(\mathbf{X}^\sharp) + \ell_{\text{pOSE}}(\mathbf{R}\mathbf{X} + \mathbf{t}), \quad (39)$$

and with our bilinear formulation we solve

$$\min_{\mathbf{C}, \mathbf{B}^\sharp, \mathbf{t}} \tilde{\mathcal{R}}_h(\mathbf{C}, \mathbf{B}^\sharp) + \ell_{\text{pOSE}}(\mathbf{R}\mathcal{U}(\mathbf{C}\mathbf{B}^\sharp) + \mathbf{t}). \quad (40)$$

In our comparison, we employ the same initialization heuristic for the weights w_i on the singular values as in [24], namely

$$w_i = \frac{\xi}{\sigma_i(\mathbf{X}_0^\sharp) + \delta}, \quad (41)$$

where $\delta > 0$ is added to avoid division by zero, and $\xi > 0$. The matrix $\mathbf{X}_0^\sharp = \mathbf{R}^+ \mathbf{M}$, where \mathbf{R}^+ is the pseudo-inverse of \mathbf{R} , which is a common initialization scheme for NRSfM [11, 24, 39].

We use $\eta = 0.05$ and $K = 3$, while for the sequences, we choose $a_i = w_i$ and $b_i = w_i/2$, where w_i is defined in (41) with $\xi = 5 \times 10^{-3}$ and $\delta = 10^{-8}$. We refine the solution obtained with ADMM formulation (after convergence) by performing 100 iterations of our method. The results are summarized in Table 2, with the average log-losses and 3D reconstruction errors, respectively, over the 6 different sequences of each dataset. In Figure 3 we show qualitative results for one the sequences of each dataset, as well as examples of the convergence plots of the two methods. Our method was always able to achieve a better loss than the ADMM solution, and in 4 out of the 5 datasets that improvement in loss led to a more accurate 3D reconstruction.

5 Conclusions

In this paper we have provided a bilinear optimization framework compatible with a wide range of penalty functions. Furthermore, we have shown that the proposed regularizer

is equivalent with the linear counterpart, making the transition from a splitting scheme based methodology to our differentiable bilinear framework easier.

References

- [1] F. R. Bach. “Convex relaxations of structured matrix factorizations”. In: *CoRR* abs/1309.3117 (2013).
- [2] R. Basri, D. Jacobs, and I. Kemelmacher. “Photometric Stereo with General, Unknown Lighting”. In: *International Journal of Computer Vision* 72.3 (May 2007), pp. 239–257. ISSN: 0920-5691.
- [3] R. Bhatia and T. Jain. “On symplectic eigenvalues of positive definite matrices”. In: *Journal of Mathematical Physics* 56.11 (2015), p. 112201.
- [4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Found. Trends Mach. Learn.* 3.1 (Jan. 2011), pp. 1–122.
- [5] C. Bregler, A. Hertzmann, and H. Biermann. “Recovering non-rigid 3D shape from image streams”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2000.
- [6] A. M. Buchanan and A. W. Fitzgibbon. “Damped newton algorithms for matrix factorization with missing data”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2005.
- [7] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino. “Unifying Nuclear Norm and Bilinear Factorization Approaches for Low-rank Matrix Decomposition”. In: *International Conference on Computer Vision (ICCV)*. 2013, pp. 2488–2495.
- [8] E. J. Candès and B. Recht. “Exact matrix completion via convex optimization”. In: *Foundations of Computational Mathematics* 9.6 (2009), pp. 717–772.
- [9] L. Canyi, J. Tang, S. Yan, and Z. Lin. “Generalized Nonconvex Nonsmooth Low-Rank Minimization”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014).
- [10] M. Carlsson. “On Convex Envelopes and Regularization of Non-Convex Functionals without moving Global Minima”. In: *Journal of Optimization Theory and Applications* 183 (2019), pp. 66–84.
- [11] Y. Dai, H. Li, and M. He. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization”. In: *International Journal of Computer Vision* 107.2 (2014), pp. 101–122.

- [12] A. Eriksson, T. Thanh Pham, T.-J. Chin, and I. Reid. “The k-Support Norm and Convex Envelopes of Cardinality and Rank”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.
- [13] C. Grussler and P. Giselsson. “Low-Rank Inducing Norms with Optimality Interpretations”. In: *CoRR* abs/1612.03186 (2016). URL: <http://arxiv.org/abs/1612.03186>.
- [14] C. Grussler, A. Rantzer, and P. Giselsson. “Low-rank optimization with convex constraints”. In: *CoRR* abs/1606.01793 (2016). URL: <http://arxiv.org/abs/1606.01793>.
- [15] B. D. Haeffele and R. Vidal. “Structured Low-Rank Matrix Factorization: Global Optimality, Algorithms, and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.6 (2020), pp. 1468–1482.
- [16] J. He, L. Balzano, and A. Szlam. “Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2012, pp. 1568–1575.
- [17] J. H. Hong and A. Fitzgibbon. “Secrets of Matrix Factorization: Approximations, Numerics, Manifold Optimization and Random Restarts”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.
- [18] J. H. Hong and C. Zach. “pOSE: Pseudo Object Space Error for Initialization-Free Bundle Adjustment”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018, pp. 1876–1885.
- [19] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.
- [20] J. H. Hong, C. Zach, A. Fitzgibbon, and R. Cipolla. “Projective Bundle Adjustment from Arbitrary Initialization Using the Variable Projection Method”. In: *European Conference on Computer Vision (ECCV)*. 2016, pp. 477–493.
- [21] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. “Fast and Accurate Matrix Completion via Truncated Nuclear Norm Regularization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2117–2130.
- [22] J. P. Iglesias, C. Olsson, and M. Valtonen Örnåhag. “Accurate Optimization of Weighted Nuclear Norm for Non-Rigid Structure from Motion”. In: *European Conference on Computer Vision (ECCV)*. Aug. 2020, pp. 21–37.

-
- [23] S. H. N. Jensen, A. Del Bue, M. E. B. Doest, and H. Aanæs. “A Benchmark and Evaluation of Non-Rigid Structure from Motion”. In: *arXiv preprint arXiv:1801.08388* (2018).
- [24] S. Kumar. “A Simple Prior-Free Method for Non-rigid Structure-from-Motion Factorization : Revisited”. In: *CoRR* abs/1902.10274 (2019).
- [25] S. Kumar. “Non-Rigid Structure from Motion: Prior-Free Factorization Method Revisited”. In: *IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020*. IEEE, 2020, pp. 51–60.
- [26] H. Lai, Y. Pan, C. Lu, Y. Tang, and S. Yan. “Efficient k-Support Matrix Pursuit”. In: *European Conference on Computer Vision 2014*. Vol. 8690. 2014.
- [27] C. Lu, C. Zhu, C. Xu, S. Yan, and Z. Lin. “Generalized Singular Value Thresholding”. In: *AAAI*. 2015.
- [28] A. M. McDonald, M. Pontil, and D. Stamos. “Spectral k-Support Norm Regularization”. In: *Advances in Neural Information Processing Systems*. 2014.
- [29] T. H. Oh, Y. W. Tai, J. C. Bazin, H. Kim, and I. S. Kweon. “Partial Sum Minimization of Singular Values in Robust PCA: Algorithm and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4 (2016), pp. 744–758.
- [30] C. Olsson, M. Carlsson, and E. Bylow. “A Non-convex Relaxation for Fixed-Rank Approximation”. In: *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. 2017, pp. 1809–1817.
- [31] C. Olsson, M. Carlsson, F. Andersson, and V. Larsson. “Non-Convex Rank/Sparsity Regularization and Local Minima”. In: *Proceedings of the International Conference on Computer Vision* (2017).
- [32] C. Olsson and O. Enqvist. “Stable Structure from Motion for Unordered Image Collections”. In: *Proceedings of the Scandinavian Conference on Image Analysis (SCIA)*. 2011, pp. 524–535.
- [33] M. V. Örnhaug, C. Olsson, and A. Heyden. “Differentiable Fixed-Rank Regularisation using Bilinear Parameterisation”. In: *Proceedings of the British Machine Vision Conference (BMVC)*. 2019.
- [34] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization”. In: *SIAM Rev.* 52.3 (Aug. 2010), pp. 471–501.
- [35] C. Russell, J. Fayad, and L. Agapito. “Energy based multiple model fitting for non-rigid structure from motion”. In: 2011, pp. 3009–3016.

- [36] F. Shang, J. Cheng, Y. Liu, Z. Luo, and Z. Lin. “Bilinear Factor Matrix Norm Minimization for Robust PCA: Algorithms and Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.9 (Sept. 2018), pp. 2066–2080.
- [37] D. Stoyanov, G. Mylonas, F. Deligianni, A. Darzi, and G. Yang. “Soft-Tissue Motion Tracking and Structure Estimation for Robotic Assisted MIS Procedures”. In: vol. 8. Feb. 2005, pp. 139–146.
- [38] C. Tomasi and T. Kanade. “Shape and Motion from Image Streams Under Orthography: A Factorization Method”. In: *International Journal of Computer Vision* 9.2 (1992), pp. 137–154.
- [39] J. Valmadre, S. Sridharan, S. Denman, C. Fookes, and S. Lucey. “Closed-Form Solutions for Low-Rank Non-Rigid Reconstruction”. In: *2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. Nov. 2015, pp. 1–6.
- [40] M. Valtonen Örnthag and C. Olsson. “A Unified Optimization Framework for Low-Rank Inducing Penalties”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.
- [41] M. Valtonen Örnthag, C. Olsson, and A. Heyden. “Bilinear Parameterization for Differentiable Rank-Regularization”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2020.
- [42] A. Varol, M. Salzmann, E. Tola, and P. Fua. “Template-free monocular reconstruction of deformable surfaces”. In: *International Conference on Computer Vision (ICCV)*. 2009, pp. 1811–1818.
- [43] N. Wang, T. Yao, J. Wang, and D.-Y. Yeung. “A Probabilistic Approach to Robust Matrix Factorization”. In: *European Conference on Computer Vision (ECCV)*. 2012.
- [44] C. Xu, Z. Lin, and H. Zha. “A Unified Convex Surrogate for the Schatten- p Norm”. In: *Proceedings of the Conference on Artificial Intelligence (AAAI)*. 2017.
- [45] N. Yair and T. Michaeli. “Multi-Scale Weighted Nuclear Norm Image Restoration”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

Supplementary Material

A Proof of Lemma 1

We present the omitted proof from the main paper, but first we establish the following property.

Lemma 2. *The sequence $(s_i - \gamma_i)_{i=1}^n$ is non-decreasing.*

Proof. We choose k such that $\gamma_i \geq \sqrt{b_i}$ for $i \leq k$ and $\gamma_i < \sqrt{b_i}$ for $i > k$. Then, for $i < k$

$$\begin{aligned} s_i - \gamma_i &= a_i + \max(\gamma_i, \sqrt{b_i}) - \gamma_i \\ &= a_i \leq a_{i+1} = s_{i+1} - x_{i+1}, \end{aligned} \quad (42)$$

and, for $i = k$

$$\begin{aligned} s_k - x_k &= a_k \leq a_{k+1} \leq a_{k+1} + \sqrt{b_{k+1}} - x_{k+1} \\ &= s_{k+1} - x_{k+1}. \end{aligned} \quad (43)$$

For $i > k$ we have

$$s_i - \gamma_i = a_i + \sqrt{b_i} - \gamma_i, \quad (44)$$

which is non-decreasing since both of $(a_i + \sqrt{b_i})_{i=1}^n$ and $(-\gamma_i)_{i=1}^n$ are. \square

Now, we turn our attention to the original statement.

Proof of Lemma 1. First we note that since the elements of \mathbf{v} are positive the elements of \mathbf{z}^* that can take multiple values should be taken as large as possible. As stated above this means that (25) holds for all elements of \mathbf{z}^* . If $z_i^* = z_{i+1}^*$ then

$$z_i^* - \gamma_i = z_{i+1}^* - \gamma_{i+1} + \underbrace{\gamma_{i+1} - \gamma_i}_{\leq 0} \leq z_{i+1}^* - \gamma_{i+1} \quad (45)$$

If $z_i^* > z_{i+1}^*$ then according to (25) we have $z_i^* \leq s_i$ and $z_{i+1}^* \geq s_{i+1}$ which, together with Lemma 2, gives

$$z_i^* - \gamma_i \leq s_i - \gamma_i \leq s_{i+1} - \gamma_{i+1} \leq z_{i+1}^* - \gamma_{i+1}. \quad (46)$$

\square

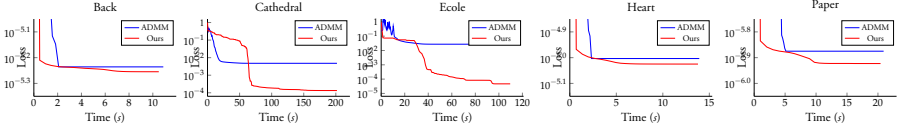


Figure 6: Convergence plots for the remaining sequences.

B Proof of the Claim in Section 2.3

We formulate the claim as a lemma.

Lemma 3. *Let $2\mathbf{z} \in \partial g(\boldsymbol{\eta})$. Then z_i can only take one value when $\eta_i \neq 0$, for all $i = 1, \dots, n$.*

Proof. By Danskin's Theorem (see e.g. [1]) it follows that $\partial g(\boldsymbol{\eta}) = 2 \operatorname{argmax}_{\mathbf{z}} \ell(\boldsymbol{\eta}, \mathbf{z})$, where

$$\ell(\boldsymbol{\eta}, \mathbf{z}) := 2\langle \boldsymbol{\eta}, \mathbf{z} \rangle - \sum_{i=1}^n \left[\left[|z_{[i]}| - a_i \right]_+^2 - b_i \right], \quad (47)$$

such that $g(\boldsymbol{\eta}) = \max_{\mathbf{z}} \ell(\boldsymbol{\eta}, \mathbf{z})$. Partition $\boldsymbol{\eta} = (\boldsymbol{\eta}', \mathbf{0})$, where all elements of $\boldsymbol{\eta}' \in \mathbb{R}_k$, $k \leq n$, has a positive magnitude. It is clear that we may choose any value $|z_{[i]}| \leq a_i + \sqrt{b_i}$ for the last $n - k$ elements of the maximizing vector, as long as $(|z_{[i]}|)_{i=1}^n$ is non-increasing. Furthermore, the first k elements are obtained by

$$\ell(\boldsymbol{\eta}', \mathbf{z}') := 2\langle \boldsymbol{\eta}', \mathbf{z}' \rangle - \sum_{i=1}^k \left[\left[|z'_{[i]}| - a_i \right]_+^2 - b_i \right], \quad (48)$$

where $\mathbf{z}' \in \mathbb{R}^k$ are the first k elements of \mathbf{z} . The objective in (48) is strictly concave in \mathbf{z}' , since all $\eta'_i > 0$, and, consequently, has a unique maximizer. In turn, the elements of the maximizing vector \mathbf{z} , can only take a single value when $\boldsymbol{\eta} \neq \mathbf{0}$. \square

C Detailed Algorithm

In this section we present a more detailed description of the algorithm. First we approximate the regularization term, as described in Section 2.3, then apply the Ruhe Wedin 2 approximation (see [2] for details) and take one step of VarPro. Since VarPro uses Jacobians, w.r.t. \mathbf{B} and \mathbf{C} , we must linearize the contribution from the regularizer. As the regularization term becomes

$$\| \operatorname{diag}(\mathbf{w}^{(t)})\mathbf{B} \|_F^2 + \| \operatorname{diag}(\mathbf{w}^{(t)})\mathbf{C} \|_F^2, \quad (49)$$

where $\text{diag}(w^{(t)})$ is a diagonal matrix with the weights $w_i^{(t)}$ in the diagonal, a natural linearization emerges by simply column stacking the variables \mathbf{B} and \mathbf{C} , respectively.

With $\mathbf{b} = \text{vec}(\mathbf{B})$ and $\mathbf{c} = \text{vec}(\mathbf{C}^T)$, we may write (49) as

$$\|\mathbf{J}_B^{\text{reg}} \mathbf{b}\|^2 + \|\mathbf{J}_C^{\text{reg}} \mathbf{c}\|^2. \tag{50}$$

Note that $\mathbf{J}_B^{\text{reg}}$ and $\mathbf{J}_C^{\text{reg}}$ are diagonal matrices.

Given a current iterate $(\mathbf{b}^{(t)}, \mathbf{c}^{(t)})$ we write the regularization term as $\|\mathbf{J}_B^{\text{reg}} \delta \mathbf{b} + \mathbf{r}_B\|^2 + \|\mathbf{J}_C^{\text{reg}} \delta \mathbf{c} + \mathbf{r}_C\|^2$, where $\mathbf{r}_B = \mathbf{J}_B^{\text{reg}} \mathbf{b}^{(t)}$, $\mathbf{r}_C = \mathbf{J}_C^{\text{reg}} \mathbf{c}^{(t)}$, $\mathbf{b} = \mathbf{b}^{(t)} + \delta \mathbf{b}$ and $\mathbf{c} = \mathbf{c}^{(t)} + \delta \mathbf{c}$.

For the data term, the residuals $\mathcal{A}\mathbf{B}\mathbf{C}^T - \mathbf{b}$ around $(\mathbf{b}^{(t)}, \mathbf{c}^{(t)})$ can be linearized, resulting in

$$\mathbf{J}_B^{\text{data}} \delta \mathbf{b} + \mathbf{J}_C^{\text{data}} \delta \mathbf{c} + \mathbf{r}^{\text{data}}. \tag{51}$$

The full objective can therefore be written as

$$\|\mathbf{J}_B \delta \mathbf{b} + \mathbf{J}_C \delta \mathbf{c} + \mathbf{r}\|^2, \tag{52}$$

where

$$\mathbf{J}_B = \begin{bmatrix} \mathbf{J}_B^{\text{reg}} \\ 0 \\ \mathbf{J}_B^{\text{data}} \end{bmatrix}, \mathbf{J}_C = \begin{bmatrix} 0 \\ \mathbf{J}_C^{\text{reg}} \\ \mathbf{J}_C^{\text{data}} \end{bmatrix}, \mathbf{r} = \begin{bmatrix} \mathbf{r}_B \\ \mathbf{r}_C \\ \mathbf{r}^{\text{data}} \end{bmatrix}. \tag{53}$$

Lastly, an optional refactorizing of the current iterate using SVD can be performed, as discussed in the main paper. We summarize these steps in Algorithm 1.

D Parameters Used in Experiments

The values of the parameters used to define the sequences for each of the datasets are shown in Table 3. In all experiments we use $\delta = 10^{-6}$.

E Convergence Plots

We display the convergence plots that were omitted in the main paper, see Figure 6. As was noted in the main text, the difference in loss is substantially larger for rigid objects; however, even small differences in loss can have a significant impact on 3D reconstructions.

Input: Robust penalty function f , linear operator \mathcal{A} and regularization parameter μ , damping parameter λ .
Initialize \mathbf{B} and \mathbf{C} with random entries
while *not converged* **do**
 Compute weights $w^{(t)}$ from current iterate (\mathbf{B}, \mathbf{C})
 Compute the vectorizations $\mathbf{b} = \text{vec}(\mathbf{B})$, $\mathbf{c} = \text{vec}(\mathbf{C}^T)$
 Compute residuals \mathbf{r}_B , \mathbf{r}_C , and Jacobians $\mathbf{J}_B^{\text{data}}$ and $\mathbf{J}_C^{\text{data}}$ depending on \mathcal{A}
 Compute residual \mathbf{r}^{reg} , and Jacobians $\mathbf{J}_B^{\text{reg}}$ and $\mathbf{J}_C^{\text{reg}}$
 Create full residual \mathbf{r} and Jacobians \mathbf{J}_B and \mathbf{J}_C
 Compute $\tilde{\mathbf{J}}^T \tilde{\mathbf{J}} + \lambda \mathbf{I} = \mathbf{J}_B^T (\mathbf{I} - \mathbf{J}_C \mathbf{J}_C^+) \mathbf{J}_B + \lambda \mathbf{I}$
 Compute $\mathbf{b}' = \mathbf{b} - (\tilde{\mathbf{J}}^T \tilde{\mathbf{J}} + \lambda \mathbf{I})^{-1} \mathbf{J}_B \mathbf{r}$ and reshape into matrix \mathbf{B}'
 Compute \mathbf{C}' by minimize the full objective (52) with fixed \mathbf{B}'
 if $\mathcal{R}(\mathbf{B}' \mathbf{C}'^T) + \|\mathcal{A}(\mathbf{B}' \mathbf{C}'^T) - \mathbf{b}\|^2 < \mathcal{R}(\mathbf{B} \mathbf{C}^T) + \|\mathcal{A}(\mathbf{B} \mathbf{C}^T) - \mathbf{b}\|^2$ **then**
 $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \text{svd}(\mathbf{B}' \mathbf{C}'^T)$
 Update $\mathbf{B} = \mathbf{U} \sqrt{\mathbf{\Sigma}}$ and $\mathbf{C} = \mathbf{V} \sqrt{\mathbf{\Sigma}}$
 Decrease λ
 else
 Increase λ
 end
end

Algorithm 1: Outline of the bilinear method.

Table 3: Parameters used for defining the sequences $\{a_i\}$ and $\{b_i\}$ in the matrix recovery experiments. [†]For the Cathedral, École, and Door datasets with WNN regularization we set $a_i = 0$ for the first four singular values and a high value for the remaining ones (truncated nuclear norm).

	a_{NN}	a_{WNN}	a_{LI}	a_{SV}
Cathe.	7.5×10^{-3}	$0/10^{\dagger}$	-	-
École	2.5×10^{-3}	$0/10^{\dagger}$	-	-
Door	1.5×10^{-3}	$0/10^{\dagger}$	-	-
Back	5×10^{-4}	1×10^{-6}	5×10^{-6}	5×10^{-9}
Heart	4.5×10^{-4}	5×10^{-6}	5×10^{-6}	1×10^{-8}
Paper	2.6×10^{-4}	2.5×10^{-7}	1×10^{-6}	5×10^{-10}

References

- [1] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.
- [2] J. H. Hong, C. Zach, and A. Fitzgibbon. “Revisiting the Variable Projection Method for Separable Nonlinear Least Squares Problems”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5939–5947.