Computer Vision Based Analysis of Animal Behavior

Palmér, Tobias

2017

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

Computer Vision Based Analysis of Animal Behavior

Tobias Palmér

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematics

Mathematics
Centre for Mathematical Sciences
Lund University
Box 118
SE-221 00 Lund
Sweden

http://www.maths.lth.se/

Doctoral Theses in Mathematical Sciences 2017:2
ISSN 1404-0034

ISBN 978-91-7753-167-8 (print)
ISBN 978-91-7753-168-5 (pdf)
LUTFMA-1063-2017

Printed in Sweden by MediaTryck, Lund 2017

Abstract

The behavior of animals is commonly studied in medicine and biology. There is a large variation in what animals are studied, in experimental paradigms and purpose. However, many studies on animal behavior have at least one thing in common – it typically involves measuring or studying the kinematics of the animal. To allow for verifiable and quantitative behavioral analysis, the experiments are recorded and kinematic data is extracted from the videos using computer vision methods. This thesis deals with the development of methods that takes recorded videos as input and provides behavioral data as output. The system of methods can be split into three parts – tracking of animal pose, extraction of kinematic features and analysis of the features. This thesis focus mainly on the two first parts. However, an important aspect in the design of the system is that *all* parts should be compatible. Therefore, all method development has been conducted in collaboration with medical/biological scientists.

This thesis contains computer vision methods for tracking rats, marmosets, zebrafish, jellyfish and zooplankton. Most of the projects are represented by a scientific paper that outlines the computer vision methods, and a paper that focus on the medical/biological application of the computer vision based system. One of the methods is applied to study the correlation between fine-kinematic behavior and neuronal activity in rats. Another method is used to characterize the long term effects of the marmoset model of Parkinson's disease. Thirdly, a high-throughput system is developed to quantify drug-induced changes in zebrafish larvae behavior. Fourthly, steps are taken towards a system that allows for studying the correlation between visual stimuli and movement output in the box jellyfish. Lastly, nonlinear positioning methods are proposed for the purpose of studying e.g. multiple threat response in zooplankton inside an aquarium.

Additionally, and seemingly an outlier, this thesis features a novel method for

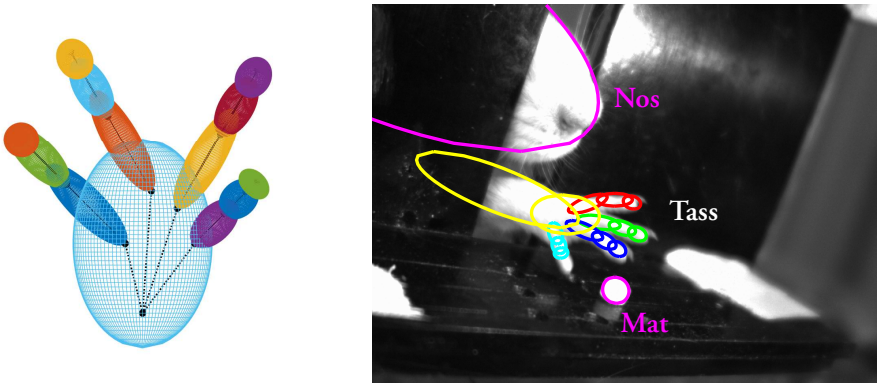
estimating relative camera motion in an underwater setting. The method is not applied for analyzing animal behavior, but it is related to the geometrical problems of refraction encountered while positioning the zooplankton. In this project, we leverage the pseudo-depth information that is contained in underwater images to design a three point relative pose algorithm.

Populärvetenskaplig sammanfattning

I medicinsk och biologisk forskning utförs ofta djurförsök. Vad för djur som studeras, hur de studeras och med vilken frågeställning varierar kraftigt. Det kan till exempel vara *dafnior* (vattenloppor) i ett akvarium som studeras för att se hur de beter sig när det finns både rovdjur och farlig UV-strålning att skydda sig från. Eller så kan det vara silkesapor med Parkinson-liknande sjukdom som studeras för att undersöka hur s.k. *deep brain stimulation* (DBS: elektrisk stimulering djupt in i hjärnan) kan lindra deras symptom. För att möjliggöra sådana studier behövs data. När en människa undersöks i ett vetenskapligt experiment finns ofta möjligheten att ställa frågor – till exempel om man upplevt obehag vid *DBS*. Men när djur undersöks är det svårare att få direkta svar på sådana frågor. Därför ligger fokus vanligtvis på att studera djurets rörelser, till exempel under tiden som djuret lär sig att navigera i en labyrint eller lär sig ett finmotoriskt beteende (se figur 1).

Automatiserad bildanalys är ofta ett oumbärligt redskap för att möjliggöra djurstudier. För det första så finns möjlighet att samla in bättre data än vad som kunnat göras manuellt, till exempel att ta fram hur individuella fingrar rör sig i 3D hos en råtta som plockar upp mat. Dessutom blir den insamlade datamängden objektiv. Till exempel att manuellt undersöka *hur* en råtta rör sig i en åtta timmar lång videoinspelning är inte bara koncentrationsmässigt utmanande utan även subjektivt och opålitligt. Med automatiserad bildanalys kan exempelvis position och riktning hos råttan beräknas och sedan studeras kvantitativt. För det andra så möjliggör automatiseringen mycket större studier. Detta medför ett större dataunderlag att analysera och därmed en ökad pålitlighet hos dessa.

Den här avhandlingen handlar i första hand om att utveckla bildanalytiska metoder som kan tillämpas för medicinska och biologiska djurförsök, men också



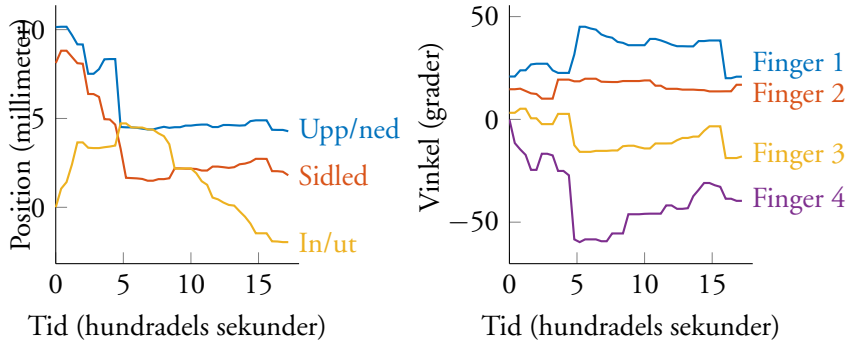
Figur 1: Vänster: tass-modellen. Höger: den fullständiga modellen ritad ovanpå en bild

om hur data från dessa metoder kan användas för att analysera djurbeteende. Bildanalytiska lösningar och tillhörande medicinska eller biologiska studier på fem olika djurarter presenteras. Även om djuren och frågeställningarna skiljer sig mycket åt så är samtliga lösningar lika på ett sätt - en matematisk modell av djuret används och parametrarna för dessa uppskattas över alla bilder i inspelningarna.

I det så kallade *skilled reaching* försöket studeras råttor som lär sig att plocka upp mat med framtassarna. I detta fallet så är det tassens rörelser som ska analyseras. Därför skapas en modell av den som anpassas efter videodata (se figur 1). Det som senare faktiskt studeras är inte hur modellen ser ut, utan hur parametrarna för den ser ut. Det vill säga det som analyseras är data som den i figur 2 som består av bland annat position för tassen (den vänstra figuren) i 3D och hur mycket fingrarna i tassen spretar (den högra figuren).

Ett annat djur som studeras är silkesapor. Målet med försöken är att se hur aporna beter sig när de genom en speciell medicinsk metod försetts med Parkinson-liknande symptom i ena hjärnhalvan. Två olika försöksuppställningar används för att karaktärisera beteeendet. I detta fallet så var det lämpligt att modellera aporna som ellipser där parametrarna ges av position, rotation och storlek. Därmed så samlas information om åt var apan befinner sig, vilket håll apan är riktad, samt hur utsträckt den är.

Djurförsök är också vanligt förekommande vid utveckling av nya läkemedel. Vanligtvis testas läkemedlen på gradvis större och större djurarter. En av de mindre arterna är däggdjuret zebrafisk. Ett av syftena med projektet är att förbättra utvär-

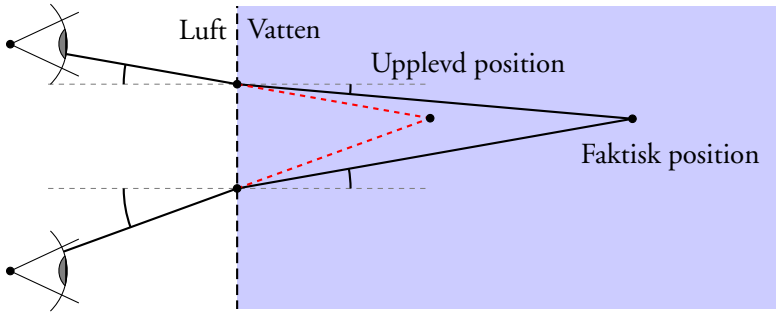


Figur 2: Delar av rörelsedata för en råtta som plockar upp mat. Vänster: position för tassen. Höger: vinklar som anger hur mycket dess fyra fingrar spretar utåt.

deringen av droger på zebrafiskar för att på så sätt undvika studier på större djur. Samtidigt som en högre kvalitet på data efterfrågas så är en stor kvantitet också av intresse. Den valda kompromissen innebär att 48 fiskar filmas samtidigt i filmer där fiskarna ser ut som böjda streck. I den bildanalytiska lösningen modelleras därför fiskarna som kurvor. På så sätt fås data om var fiskarna befinner sig, hur de är vinklade och hur deras kroppar böjer sig.

För biologer som är inriktade på syn är kubmaneten ett intressant djur att studera. Den har nämligen ögon, samtidigt som den inte har någon riktig hjärna. Därför så är (antagligen) kopplingen mellan synintryck och rörelse avsevärt enklare än hos högre stående organismer med syn. För att studera denna koppling så fixeras en manet så att den inte kan röra sig. Sedan filmas den samtidigt som den ges synintryck och försöker röra sig. Kroppen är transparent och svår att följa, men dess fyra samlingar av ögon är mycket enklare att detektera. En följning av dessa ger en bra indikation på hur maneten försöker röra sig.

Ett annat djur som studeras inom biologi är daphnier. Det är ett litet kräftdjur som i praktiken används bland annat som indikator för gifthalter i sjöar. Studien som presenteras i den här avhandlingen handlar om hur daphnior rör sig när de utsätts för multipla hot i form av UV-strålning ovanifrån och rovdjur underifrån. I de filmer som analyseras i det här projektet ser daphniorna väldigt små ut och är runda. Därför modelleras de bara som en punkt. I det presenterade projektet är dock fokus mer på den geometri som uppstår när ljusstrålar går genom vatten. När ljus inifrån akvariet passerar ut till glaset och sedan ut i luften så böjs det. Detta gör att objekt som befinner sig i vatten ser ut att vara närmare än vad de



Figur 3: Objekt i vatten är inte där man kan tro att de är.

egentligen är för den som ser det utifrån. Om man inte tar hänsyn till krökningen när man beräknar var daphniorna befinner sig så får man systematiska fel i beräknade 3D-positioner, precis som visas i figur 3. Därför presenteras en metod för att på ett korrekt sätt beräkna var någonstans daphniorna befinner sig.

Det är inte bara geometrin som påverkas av vatten. Speciellt på stora avstånd så försämras synligheten på grund av att ljuset absorberas av vattnet. Hur mycket som absorberas beror på vilken färg ljuset har (dvs. våglängd). Rött ljus absorberas snabbast och gör att undervattensbilder brukar bli mestadels blå-gröna. Men att färgerna försämras är inte bara en nackdel. Det ger även en indikation på avstånd. Detta använder vi för att beräkna så kallad *relativ rörelse* mellan två bilder, det vill säga hur kameran har förflyttat i världen sig mellan två bilder.

Sammanfattningsvis presenteras i den här avhandlingen främst nya metoder inom medicinsk och biologisk forskning, men bidrar även med matematiska metoder för undervattensbilder.

Preface

This thesis contributes to method development of computer vision and image analysis with medical and biological applications.

The contents of the thesis is based on the following ten papers. My contributions are noted below each paper. A subsidiary paper is also listed below.

Main papers

- Tobias Palmér, Kalle Åström, Olof Enqvist, Nela Ivica, and Per Petersson. "Rat Paw Tracking for Detailed Motion Analysis". In: *Visual observation and analysis of Vertebrate And Insect Behavior 2014*. 2014

  Author contributions: This paper was conceived by PP. Surgery and building of electrodes was performed by NI. Recording of experiments was done by NI. Computer vision algorithms were developed by TP under supervision of OE and KÅ. Analysis of neural data was performed by PH.

- Tobias Palmér, Martin Tamtè, Pär Halje, Olof Enqvist, and Per Petersson. "A System for Automated Tracking of Motor Components in Neurophysiological Research". In: *Journal of neuroscience methods* 205.2 (2012), pp. 334–344

  Author contributions: The paper was conceived by PP. Surgery and building of electrodes was performed by MT. Recording of experiments was done by TP and MT. Computer vision algorithms were developed by TP under supervision of OE. Analysis of neural data was performed by MT and PH.

- Tobias Palmér, Maxwell B. Santana, Romulo A. Fuentes, and Per Petersson. "Automated Tracking of Motor Behavior as a Means to Assess Severity of

Symptoms in the 6-OHDA Marmoset Model of Parkinsons Disease". In: *Visual observation and analysis of Vertebrate And Insect Behavior 2012*. 2012

Author contributions: The paper was conceived by PP and RF. Experiments were carried out by MS. MS, PP and RF performed the surgeries. Computer vision algorithms were developed by TP.

- Maxwell B. Santana, Tobias Palmér, Hougelle Simplício, Romulo A. Fuentes, and Per Petersson. "Characterization of Long-Term Motor Deficits in the 6-OHDA Model of Parkinson's Disease in the Common Marmoset". In: *Behavioural brain research* 290 (2015), pp. 90–101

  Author contributions: The paper was conceived by PP and RF. Experiments were carried out by MS. MS, HS, PP and RF performed the surgeries. Computer vision algorithms were developed by TP.

- Tobias Kjellberg, Tobias Palmér, Magnus Oskarsson, and Kalle Åström. "Tracking the Motion of Box Jellyfish". In: *Visual observation and Analysis of Vertebrate and Insect Behavior 2014*. 2014

  Author contributions: Conceived by KÅ and MO. Image analysis methods developed by TK under supervision of MO, KÅ and TP. Paper written by MO.

- Tobias Palmér, Kalle Åström, Olof Enqvist, and Per Petersson. "Visual Analysis of Zebrafish Behavior". In: *Visual Observation and Analysis of Vertebrate and Insect Behavior 2016*. 2016

  Author contributions: Conceived by PP and TP. Image analysis algorithms were developed by TP under supervision of OE. Methods for analyzing tracking data were developed by TP, KÅ, OE and PP. The paper was written by TP. The experimental set-up presented in paper VI was used. A subset of the data presented in paper VI was used.

- Tobias Palmér, Fredrik Ek, Olof Enqvist, Roger Olsson, Kalle Åström, and Per Petersson. "Action Sequencing in the Spontaneous Behavior of Zebrafish Larvae with Implications for Drug Development". *Manuscript*

  Author contributions: Conceived by PP, TP, RO and FE. Experimental set-up constructed by TP and FE. Experiments recorded by FE. Computer vision algorithms developed by TP. Behavioral analysis developed and performed by TP under supervision of PP, KÅ and OE.

- Tobias Palmér, Giuseppe Bianco, Mikael T. Ekvall, Lars-Anders Hansson, and Kalle Åström. "Calibration, Positioning and Tracking in a Refractive and Reflective Scene". In: *International Conference on Pattern Recognition*. 2016

  Author contributions: Conceived by KÅ using the pre-existing experimental set-up presented in previous studies by GB, ME and LH (e.g. [2]). The computer vision methods were developed by TP under supervision by KÅ. The paper was written by TP.

- Mikael T. Ekvall, Tobias Palmér, Giuseppe Bianco, Jan Heuschele, Johan Bäckman, Kalle Åström, and Lars-Anders Hansson. "Daphnia Response to Multiple Threats from UV-A and Predation". *Manuscript*

  Author contributions: The experiments were performed within the experimental set-up and procedures presented in previous studies by GB, ME and LH (e.g. [2]). Experiments were recorded by ME. Calibration of the cameras and scene and the tracking of *Daphnia* was performed by TP using the methods proposed in paper VIII. The paper was written by ME.

- Tobias Palmér, Kalle Åström, and Jan-Michael Frahm. "The Misty Three Point Algorithm for Relative Pose". *Manuscript*

  Author contributions: Idea of using attenuation as a depth-sensor was conceived by JF. Minimal relative-pose problem solved by KÅ and TP. Implemented and tested by TP. The paper was written by TP under supervision by KÅ and JF.

Subsidiary paper

- Kalle Åström, Magnus Oskarsson, Tobias Kjellberg, Tobias Palmér, and Dan-Eric Nilsson. "Visual Tracking of Box Jellyfish: A Real-Time Motion Tracking System". In: *Computer Vision and Pattern Recognition in Environmental Informatics* (2015), p. 107

  Author contributions: Conceived by KÅ, MO and DN. Image analysis methods developed by TK under supervision of MO, KÅ and TP. Paper written by MO and DN.

Funding

This thesis was supported by grants from KAW (Ultimate Vision), ELLIIT, eSSENCE, Bergvall, Crafoord, Kockska, Jeansson, Hjärnfonden, Michael J Fox, MultiPark, Olle Engkvist, Parkinson, Parkinson Research, Segerfalk, Åhlen and Åke Wiberg Foundation, SSMF and the Swedish Research Council (project#: K2010-62X-21400-01-3).

Acknowledgments

First of all - I'd like to thank my supervisor Kalle. Thank you for being a constant source of ideas, inspiration and help. I'd like to thank my co-supervisor Per for introducing me to all the interesting neurophysiological problems, and the collaborations we have had for the last seven years. And I'd like to thank my second co-supervisor, Olof, for being really inspirational and helpful (even after leaving the Mathematics department).

Furthermore, I'd like to thank all my colleagues at the Center for Mathematical Sciences, and the Vision group in particular. It is a truly an inspirational research environment. In particular, thank you Mårten for all the fruitful and fruitless discussions we've had in out shared office during the last five years.

Thank you Jan-Michael for supervising me and letting me be a part of the computer science group at UNC Chapel Hill during the spring semester 2015. And, of course, for the collaboration on paper X.

— — —

Jag vill tacka min familj och mina vänner för deras stöd i livet och hjälp med att tänka på annat än matematik och avhandlingsarbete. Men även för de tillfällen då de fått mig att upptäcka tillämpningar av mina bildanalyskunskaper i vardagslivet. Speciellt tack till min mamma, som gav mig idén att undersöka vad familjens kanin gjorde på nätterna – detta ledde till min första studie av djurbeteende med hjälp av bildanalys.
Sist, och mest av allt, tack Sofia.

Stockholm, January 2017

Contents

Introduction

1 Overview of the thesis

The two main topics of this thesis are on how computer vision can be applied to extract data from animal experiments and how data from the computer vision systems can be processed to enable useful behavioral analysis. Note that the two parts overlap – e.g. to enable the analysis of how the body angle of zebrafish larvae change during turning, the computer vision system needs to estimate the body angle. Furthermore, knowledge of noise and errors in the computer vision system is crucial for designing a robust behavioral analysis system. This dependency of the parts inside the systems is illustrated in Fig. 1.

The use of automatized image analysis for the purpose of analyzing animal behavior is highly popular in e.g. medical and biological sciences. It allows for higher throughput experiments with consistent results compared to the alternative of manual labeling.

The main contributions of this thesis are presented in ten papers that span six different projects and five types of animals – rats, monkeys, zebrafish, *Daphnia* and jellyfish. The scientific results for four of the animals (rats, monkeys, zebrafish and jellyfish) are represented by two papers each – one computer vision paper and one medical/biological paper. An exception among the animal projects is the jellyfish project that is only represented by a computer vision paper.

Seemingly an outlier, the sixth included project (paper X) is not related to animal experiments. The paper is instead related to the geometrical problems of underwater geometry that are considered the *Daphnia* project. In paper VIII, it is noted that the 3D geometry in underwater imaging differs from that of in-air applications. An additional difference between underwater imaging and in-air imaging is leveraged in paper X – the depth and color dependent attenuation of

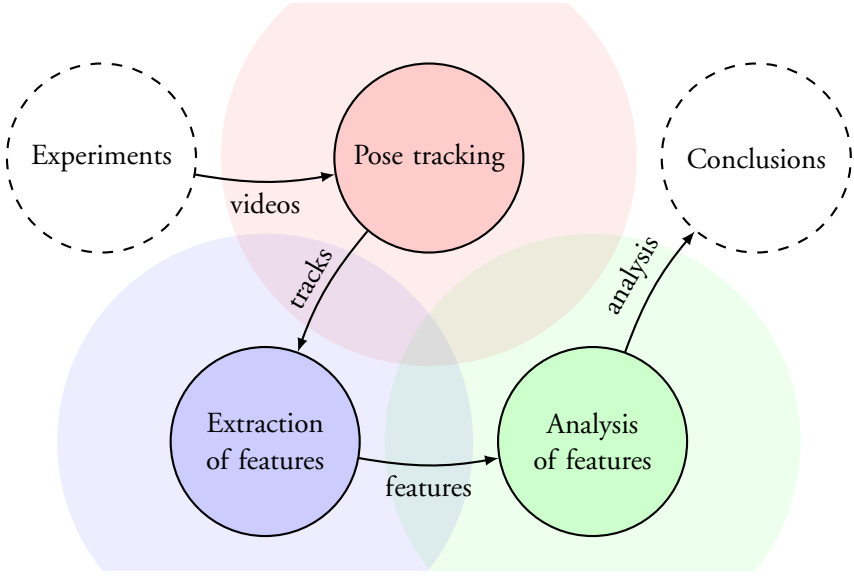


Figure 1: An overview of the systems for analyzing animal behavior included in this thesis. The created systems only transfer data as indicated by the arrows, but in the development of the systems, all parts overlap as indicated by the colors. Contributions have been made in the colored areas.

light in water provides pseudo-depth information.

This thesis is separated into four parts. In the first part, a short introduction to the theory of computer vision and image analysis is provided. Then the specific methods developed for the included papers are outlined in a different context than in the papers. In addition, some material that relates to the papers but is not included in them is presented here. The third part consists of presentations of each of the projects included in the thesis. The computer vision theory applied in each project is referenced to and the medical/biological rationale is explained. This additional material is presented for each project. The fourth part consists of the papers.

1.1 Applications for animal experiments

All motile organisms need to organize their motor output to achieve functional movements. This organization is controlled by the central nervous system. How

this is actually done is already partly known from many years of research. For example, which parts of the brain that control which parts of the body has been known at least since the 1930's. It is also well known which level of complexity different structures control. For example, the deeper parts of the brain are important for basic functions such as breathing, while the structures at the surface of the brain (the cortex) deals with more high level tasks such as dancing or playing the piano.

However, this is an oversimplification of the complex system of the brain. For example, different structures in the brain communicate with each other to organize movement. There is not one part of the brain that alone deals with, for example, controlling the muscles in the feet. A question that is far from being answered is how movements are organized by combinations of brain structures. An increased knowledge in this field would improve understanding of neural disorders, such as for example Parkinson's disease. This could potentially lead to new methods of treatment. In addition, it could lead to further development in engineering applications such as neural prosthetics and brain machine interfaces.

One approach to increasing knowledge on this topic is to conduct experiments on motile organisms with brains and measure both activity in the brain and the movement of the animal. For the purpose of understanding the human brain, studies on humans would be ideal. However, the methods that currently exist are intrusive and causes physical harm to the brain. Therefore, the harmful experiments are conducted on animals.

One of the many experiments that is commonly conducted on **rats** is *skilled reaching*. In this task, a rat is taught to reach for food rewards through a narrow slit and its movements and performance is studied. The test has proven to be sensitive in detecting for example the early stages of Parkinson's disease. A method for tracking the paw movements is proposed in the included papers (papers II and I). The computer vision method is applied on videos of rats that has microelectrodes implanted into their brains. This allows for the study of correlation between fine motor behavior and neural activity.

The **common marmoset** is one of the non-human primates that is used to study Parkinson's disease. Since the animals themselves are not known to develop this specific neural disorder, it is necessary to induce it. One of the most commonly used methods is to inject the neurotoxin 6-OHDA into the brain, with the effect that the dopamine producing cells are degenerated. This method causes a Parkinsonian behavior in the animals. One of the benefits of inducing Parkinson's

disease in animals as compared to studying pre-existing human patients is that the animals can be given the neural disorder in only one hemisphere. This allows for comparisons between the neural activity in the Parkinsonian hemisphere to that of the unharmed hemisphere. The included papers (paper IV and III) investigates the long-term effects of the 6-OHDA model of Parkinson's disease in the common marmoset.

A related topic is that of developing drugs that effects the central nervous system. In general, the animal experiments start on small and simple organisms and progress into larger and more complex animals if the results are promising. One of the smaller animals that is commonly tested is the **zebrafish**, particularly in its larval stage. The zebrafish larva is relatively simple to conduct drug testing on, since it can absorb substances directly through its skin. An improvement in detecting both efficient and inefficient substances has the potential of decreasing the amount of experiments on larger and more complex animals. So far, drug testing on zebrafish larvae have been limited to simple measures such as position and and overall distance traveled of the fish. To improve upon the quality of behavioral testing, there is need to use more advanced features for describing changes in behavior. These new features require more detailed output from the video tracking system. Therefore, the included papers (paper VI and V) propose a method that enables both high throughput and detail in the tracking data. Novel methods for analyzing the tracking output are proposed.

An animal that unlike the others featured in this thesis does not have a brain is the **box jellyfish**. Instead of a brain it has a nerve net and, remarkably among jellyfish, it has eyes. This is interesting to study since the connection between visual input and motor output can be assumed to be substantially less complex than in animals with brains. One important step towards analyzing this connection is to quantify motor output. The proposed approach (paper VII) assumes that the "head" of the jellyfish is fixated so that it can not actually move. Then the jellyfish is recorded as it receives visual stimuli and attempts to move. Its four rhopalia (clusters of eyes) are mostly clearly visible and easy to track compared to the transparent body.

Daphnia are small aquatic organisms that live in for example lakes and ponds in many parts of the world. They are sensitive to toxins in the water, which makes their behavior an indicator of toxicity levels. The included biological research paper (paper IX) focuses on the behavior of *Daphnia* under multiple threats – both UV-light from above and predation from below. A method (see paper VIII)

to estimate the 3D positions of *Daphnia* inside of the aquarium is proposed to enable to the biological study.

2 Image analysis

The difference between computer vision and image analysis is not clearly defined. The two fields use largely the same algorithms and theory. In addition, there are several other fields that are closely related and often with unclear separation – for example pattern recognition, signal processing, machine vision, image processing and photogrammetry. However, computer vision tends to focus on 3D analysis of images (e.g. structure from motion) while image analysis mainly deals with 2D pixel-wise operations (e.g. edge detection).

In this section is described the theory of 2D image analysis that is the most relevant to the included papers, as well as the basics of the field. Similarly, section 3 describes the basic and most relevant 3D computer vision theory.

2.1 Images

An image taken with an RGB camera is represented by a three dimensional data structure I on the form

$$I \in \mathbb{R}^{M \times N \times 3}, \tag{1}$$

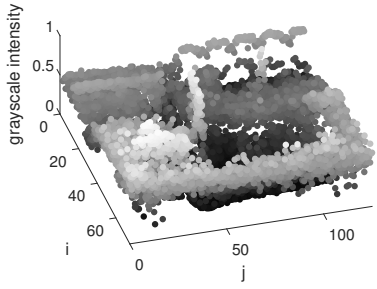
where M and N are the height and width of the image, respectively, and the third dimension consists of the three color channels – red, green and blue. In other words, an RGB image consists of MN pixels, each represented by three values. In this thesis, it is assumed that all images have pixels values that range from 0 to 1 in each color channel.

Image intensities are sometimes referenced as I_{ijk}, where i denotes the row, j the column and k the color channel. In some parts of this thesis, images are represented as a function of image coordinates (x, y). That is, an image I can also be represented by the function

$$I : (x, y) \to \mathbb{R}^3, \tag{2}$$

where (x, y) are the coordinates. Note that the coordinate system in which (x, y) is defined is not necessarily the same as (i, j).

(a) A grayscale image of Ludde the dog.



(b) A 3D scatter plot representation of (a).

Figure 2: Different representations of an image. Figure (a) and (b) contain exactly the same information to the computer, but only the left image has any semantic meaning to a human. None of the figures have semantic meaning to a computer.

Furthermore, note that there is nothing in this data structure that (without training) allows for a computer to semantically "understand" an image. That is, even though a human can see that there is a dog in Fig. 2a, there is no semantic information in the pixel values that tells the computer it is a dog (unless it has been trained to do it). What the computer "sees" is closer related to what a human can see in Fig. 2b.

2.2 Segmentation

2.2.1 Thresholding

Sometimes it is of interest to segment different parts of an image. For example, Fig. 2a can be segmented into regions in a number of ways. For example, it can be semantically based regions such as: dog, grass, paw, ear, etc. How such a segmentation can be achieved by a computer is one of the most popular subjects in the field of image analysis. It can also be more openly defined segmentation such as background and foreground. A third example that is mathematically clearly defined, but has no semantic meaning, is segmentation by applying thresholds to grayscale intensities. For example, the binary image shown in Fig. 3a is the result of classifying all pixels in Fig. 2a. Pixels that have grayscale intensities greater than 0.15 belong to class 1 and the rest belong to class 0.

(a) A binary image created by applying a threshold to the image in Fig. 2a.

(b) Erosion of (a) by three pixels.

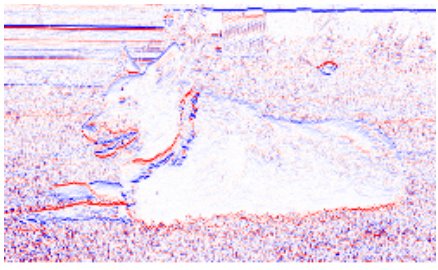
(c) Dilation of (a) by three pixels.

(d) Automatic semantic segmentation of the image in Fig. 2a.

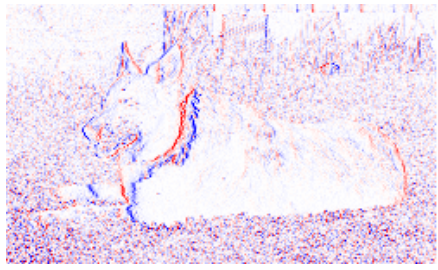
Figure 3: A binary image (a) and the results of different morphological operations applied on it in (b) and (c)). Furthermore, (d) provides an example of what methods from the field of semantic segmentation can achieve. Pixels belonging to class 0 are black and pixels belonging to class 1 are white. Note the properties of each operation – e.g. erosion removes small "islands" in (b) and dilation merges the body and head in (c).

2.2.2 Morphology

A binary image can be subject to morphological operations such as erosion and dilation. Erosion of a binary image means that the areas classified as 1 are shrunk from the borders to the regions belonging to class 0 (cf. Fig. 3a and Fig. 3b). Similarly, dilation of a binary image means an expansion of the areas classified as 1 towards the pixels belonging to class 0 (cf. Fig. 3a and Fig. 3c). The result of applying a more advanced method for segmentation is shown in Fig. 3d. Here, a method that uses convolutional neural networks (CNN) for automatic semantic segmentation is used [95]. Note that the output of the method is not only the binary image – the black area in the figure is actually classified as dog.

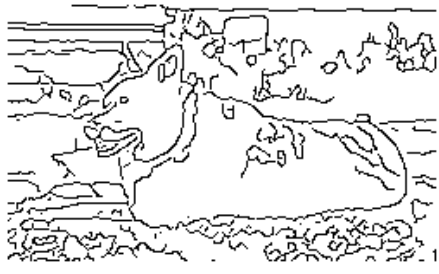


(a) Discrete derivative of 2a in vertical direction.



(b) Discrete derivative of 2a in horizontal direction.

(c) Length of discrete gradient of 2a.



(d) Result from Canny edge detector applied on 2a.

Figure 4: Edge detection. The discrete partial derivatives, (a) and (b), and the length of the gradient (c) gives an indication of edges. (d) shows the result of applying the Canny edge detector. Note that there are two different colormaps are used in the subfigures. (a)-(c) are visualized such that -1 is blue, 0 is white and $+1$ is red. Since the gradient length is non-negative, there are no blue pixels in (c). The second colormap is used in the binary image (d) – pixels that lie on an edge are black and the other are white.

2.3 Edges

Edges in an image correspond to regions of rapid changes in color and/or intensity. A simple edge detector is given by the discrete gradients in the image, that is, the discrete derivatives in the i and j directions. Fig. 4a – 4c shows an example of this, applied on the image from Fig. 2a. A threshold can be applied on the derivatives to obtain a classification of whether a pixel is an edge pixel or not.

In paper II, a method is used that computes signed edge images where the sign (-1 or 1) of a pixel depends on the orientation of the edge at the pixel. This method is later changed to on of the most commonly used edge detectors,

the Canny edge detector [12], for paper I. An example result of the Canny edge detector is shown in Fig. 4d.

2.4 Integral images

Instead of computing the discrete derivatives of an image, it is sometimes useful to compute the opposite – that is, the discrete integral of an image. They allow for fast measurements of sums over rectangular subsets of pixels in an image. The resulting integral of the image is commonly referred to as an integral image [19, 49, 88]. An example of an application where such images are useful is presented in section 4.6.1.

3 Computer vision

3.1 Cameras

This section contains descriptions of the geometry of the two camera models used in the included papers. First, the two classes of camera models – pinhole camera and generalized camera – are introduced. The introduction to cameras is then concluded by describing the intrinsic camera parameters.

3.1.1 The pinhole camera model

The pinhole camera model assumes that all rays meet at a common point, referred to as the *focal point* (or *camera center*). Assume that the focal point is located at the origin, that the camera is facing towards the positive z-axis and that the focal length is $f = 1$ (see Fig. 5). Then $z = -1$ describes the physical image plane. Note that the projection on $z = -1$ is mirrored horizontally and vertically compared to the scene. However, by instead projecting on the virtual image plane $z = 1$, the same projection as for $z = -1$ is provided but without the mirroring effects. Therefore, the plane $z = 1$ is defined as the image plane on which all points are projected.

A point $\boldsymbol{U} = (X, Y, Z)^\top$ in the scene is observed as the intersection of the line from the origin to the scene point and the plane $z = 1$. By parametrizing the line as

$$\boldsymbol{l}_{\boldsymbol{U}} : \lambda \to \mathbf{0} + \lambda(\boldsymbol{U} - \mathbf{0}) = \lambda \boldsymbol{U}, \tag{3}$$

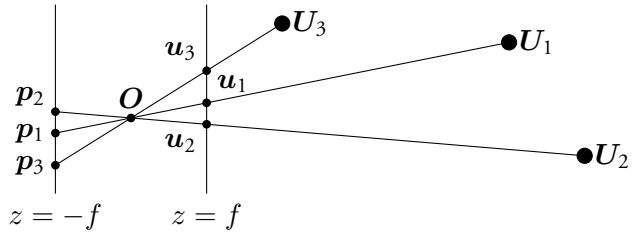


Figure 5: The geometry of the pinhole camera model. In the notation used in this thesis, $\boldsymbol{U}_k$ denotes a scene point, $\boldsymbol{u}_k$ is a normalized image coordinate and the point $\boldsymbol{p}_k$ lies on the real image plane. Note that the points $\boldsymbol{p}_k$ are never actually used. The rays from the scene points meet at the focal point $\boldsymbol{O}$.

the projection is found by picking $\lambda = 1/Z$ and inserting

$$\boldsymbol{u} = \boldsymbol{l}_{\boldsymbol{U}}(1/Z) = (X/Z, Y/Z, Z/Z)^\top = (x, y, 1)^\top . \tag{4}$$

The relation between scene points $\boldsymbol{U}$ and their projections $\boldsymbol{u}$ can be formulated as a matrix-vector multiplication as follows

$$\lambda \boldsymbol{u} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} . \tag{5}$$

3.1.2 Intrinsic parameters

So far has been described how the rays of light from the scene points are projected on the plane $z = 1$ that represents the image sensor. What has not yet been shown is how the projections relate to the image sensor elements (i.e. pixels). For this purpose, the *intrinsic* camera matrix K is defined as follows

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} , \tag{6}$$

where (f_x, f_y) is the focal length in pixels, (c_x, c_y) is the optical center in pixels and s is the skew. Note that $f_x = f_y$ if the pixels have equal sides and $s = 0$ if the pixel axes are orthogonal. The intrinsic parameters are applied to projected image points as $\boldsymbol{x} = K\boldsymbol{u}$.

3.1.3 Radial distortion

The camera lens can cause a distortion that increases with the distance from the optical center. This nonlinear effect is referred to as *radial distortion* and is in this thesis modeled by polynomials as

$$(u_x, u_y)^\top = \left(u'_x, u'_y\right)^\top \cdot (1 + k_1 r^2 + k_2 r^4 + k_3 r^6), \tag{7}$$

where (u_x, u_y) and (u'_x, u'_y) are the distorted and undistorted normalized image coordinates, respectively, and $r^2 = u_x^2 + u_y^2$.

Note that it is sometimes more convenient to define the transformation for image points $\boldsymbol{x}$. This can be achieved by first reformulating Eq. (7) using *homogeneous coordinates* as follows,

$$(u_x, u_y, 1)^\top = \left(u'_x, u'_y, \frac{1}{p(r)}\right)^\top \cdot p(r) = \Lambda_r \left(u'_x, u'_y, 1\right)^\top, \tag{8}$$

where $p(r) = 1 + k_1 r^2 + k_2 r^4 + k_3 r^6$ and $\Lambda_r = \operatorname{diag}\{p(r), p(r), 1\}$.
By multiplying both sides with $K\Lambda_r^{-1}$ and using that $\boldsymbol{u} = K^{-1}K\boldsymbol{u}$, the following equation is provided,

$$(K\Lambda_r^{-1})(\overbrace{K^{-1}\underbrace{K}}^{=I})\underbrace{(u_x, u_y, 1)^\top}_{=(x,y,1)^\top} = \underbrace{(K\overbrace{\Lambda_r^{-1})\Lambda_r}^{=I}\left(u'_x, u'_y, 1\right)^\top}_{=(x',y',1)^\top}. \tag{9}$$

In conclusion, radially distorted image points $(x, y, 1)^\top$ are related to undistorted points $(x', y', 1)^\top$ by the equation

$$\left(K\Lambda_r^{-1}K^{-1}\right)(x, y, 1)^\top = \left(x', y', 1\right)^\top. \tag{10}$$

3.1.4 Extrinsic parameters

The camera that has been used for the derivations so far has been defined in a local coordinate system. In this system, the focal point is located at the origin and the camera is directed towards the positive z-axis. Now, assume that this local coordinate system of the camera is related to the word coordinate system by a translation $\boldsymbol{t}$ followed by a rotation R. That is, the scene points $\boldsymbol{U}$ are transformed to the local coordinate system of the camera by the transformation

$$\boldsymbol{U}' = R^\top \boldsymbol{U} - R^\top \boldsymbol{t}. \tag{11}$$

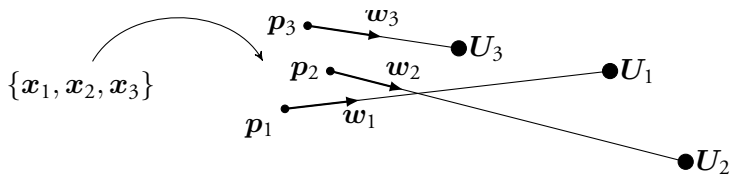


Figure 6: A generalized camera. For each image point $\boldsymbol{x}_k$, there exists a ray $(\boldsymbol{p}_k, \boldsymbol{w}_k)$ into the scene.

Thus a scene point $\boldsymbol{U}$ can be projected by first transforming it by Eq. (11) and then projecting the result, $\boldsymbol{U}'$, as in Eq. (5).

3.1.5 Summary

In conclusion, the projection of a scene point $\boldsymbol{U}$ is computed as

$$\lambda \boldsymbol{x} = K(R^\top \boldsymbol{Y} - R^\top \boldsymbol{t}) = \underbrace{KR^\top \begin{bmatrix} I & -\boldsymbol{t} \end{bmatrix}}_{P} \hat{\boldsymbol{Y}} = P\hat{\boldsymbol{U}}, \tag{12}$$

where P is the *projection matrix* and $\hat{\boldsymbol{U}} = [\boldsymbol{U}^\top \quad 1]^\top$ are the *homogeneous coordinates* [53] of $\boldsymbol{U}$. The result, $\lambda \boldsymbol{x} = (x, y, z)^\top$, is then normalized by division by z and radially distorted by Eq. (7).

3.1.6 The generalized camera model

The generalized camera model is the most general camera model. It only assumes that for each pixel, there exists a ray into the scene that originates from the pixel (see Fig. 6). No assumption is made on the continuity of the pixel-ray map. Note that in the cases where rays change direction in the scene, the *final* linear ray segment is the one that is used. An example of a generalized camera is a pinhole camera – for each pixel there exists a ray into the scene. A second example is provided by a rig of pinhole cameras whose lines of sight do not overlap. In this case, the direction into the scene for each pixel is known, and thus the rig of cameras can be modeled as a generalized camera. A third example is given by underwater cameras with flat port housings (see Fig. 7). The different physical properties of water, glass and air causes the rays from scene points to refract. Thereby, all rays do not meet at a common point. However, for each pixel, the direction into the water can be computed and thus the camera can be modeled as

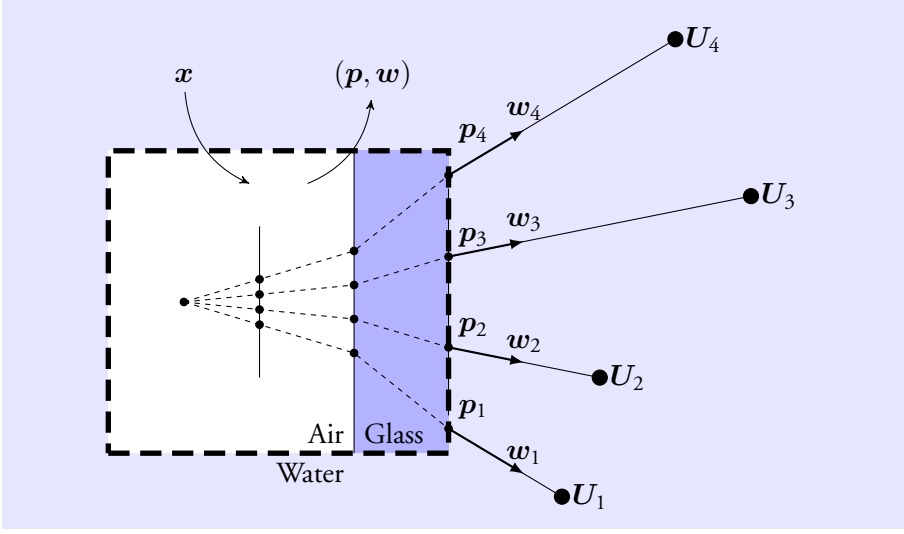


Figure 7: A pinhole camera observes a scene from behind two flat refractions. This can also be seen as a generalized camera where image points $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_4\}$ are inserted into the black (dotted) box that provide the rays $\{(\boldsymbol{p}_k, \boldsymbol{w}_k)\}$. Note that the treatment of this camera as a generalized camera discards information that could otherwise be used to add constraints for e.g. pose estimation.

a generalized camera. Additional examples are: a camera (of any kind) that views a scene through multiple non-flat reflections and five cameras that view a scene through a varying number of refractions.

Note that it is not necessarily easy to compute the mappings from the pixels to the (final) rays. And note that when treating a camera as a generalized camera, there is geometrical information that could potentially be used but is discarded. For example, the ray paths shown in Fig. 7 are constrained to meet at the focal point and the change direction according to Snell's law. Thereby, outgoing rays $(\boldsymbol{p}_k, \boldsymbol{w}_k)$ are related to each other. However, in the generalized camera model, no relations are assumed. Another example is the relative motion problem – the problem for a calibrated pinhole camera requires five pairs of corresponding points, while a generalized camera requires six pairs. The relative motion problem is further discussed in the next section.

3.2 Calibration

To calibrate means to estimate parameters of an instrument that involves the measurements or output it makes, usually to meet some standard scale. For example, a TV can be calibrated to make sure that what is intended to be displayed as red really is displayed as red and green as green, etc. In the field of computer vision, calibration most commonly refers to the estimation of a set of parameters of a camera. In this thesis, camera calibration is split up into two groups: *intrinsic calibration* and *extrinsic calibration*.

Intrinsic calibration refers to estimating the parameters describing the lens and imaging sensor, i.e. the focal length, radial distortion, tangential distortion, image sensor skew, etc. The most widely used algorithms for estimating the intrinsic parameters is based on the use of images of a checkerboard of known dimensions [6, 33, 96].

Extrinsic calibration refers to the estimation of position and orientation of a camera. This can be split into two categories – absolute pose and relative pose. The absolute pose problem is the problem of estimating the position of a camera given a set of known scene points. The relative pose problem consists of estimating the relative poses of two cameras given a set of matching image points.

There are many algorithms for extrinsic calibration. Table 8 provides an overview of some of the methods that are the most relevant for this thesis. For example, the five-point method [55, 81] uses five pairs of corresponding image points to estimate the essential matrix, assuming intrinsic calibration is done apriori. The Three Point Delta method proposed in [58] uses three pairs of corresponding image points and the difference in distance to the corresponding scene points to estimate relative pose in the generalized camera model. The direct linear transform method (DLT) (used in papers [56, 61, 62, 70]) does not model radial/tangential distortion and uses six pairs of corresponding image/world points to estimate the projection matrix of a single camera. Derivations of these methods are outlined in the following.

3.2.1 Direct linear transform

The direct linear transform (DLT) algorithm [84] is a method for computing an unknown linear transformation A in a set of similarity relations on the form

$$\boldsymbol{x}_k \propto A\boldsymbol{y}_k, \tag{13}$$

Algorithm	Assumes intrinsic calibration?	Camera model	Number of pointwise correspondences
The eight point algorithm [50, 77]	No	Pinhole	8 (x, y) points
The seven point algorithm	No	Pinhole	7 (x, y) points
The five point algorithm [55]	Yes	Pinhole	5 (x, y) points
The six point algorithm [81]	Yes	Generalized	6 (x, y) points
The Misty Three Point algorithm [58]	Yes	Generalized	3 (x, y, R, G, B) points
The Three Point Delta algorithm [58]	Yes	Generalized	3 (x, y) points and difference in depth for each pair

Figure 8: A summary of methods for relative pose estimation.

where the $\boldsymbol{x}_k$'s and $\boldsymbol{y}_k$'s are known.

The relation between scene points and image points (Eq. (12)) are on the form of Eq. (13). Consequently, the DLT can be applied to estimate the projection matrix P. In the following is shown how the DLT is applied to compute the absolute pose (i.e. the projection matrix P) using at least $n = 6$ pairs of corresponding image/scene points.

Assume that n correspondences $\{(\boldsymbol{x}_k, \boldsymbol{U}_k)\}_{k=1,\dots n}$ are known. The constraint that $\boldsymbol{x}_k \propto P\boldsymbol{U}_k$ can be rewritten as

$$P\boldsymbol{U}_k \times \boldsymbol{x}_k = 0, \tag{14}$$

since the cross-product between linearly dependent vectors is zero. The cross-product between two vectors can be formulated as a matrix-vector multiplication as follows

$$\boldsymbol{u} \times \boldsymbol{v} = [\boldsymbol{v}]_\times^\top \boldsymbol{u}, \tag{15}$$

where

$$[\boldsymbol{v}]_\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}. \tag{16}$$

Thus equation (14) can be rewritten as

$$[\boldsymbol{x}_k]_\times^\top P \boldsymbol{U}_k = 0. \tag{17}$$

This equation is sought to be reformulated on the form $M^{(k)}\boldsymbol{p} = 0$, where $M^{(k)}$ is a matrix that depends on $\boldsymbol{x}_k$ and $\boldsymbol{p}$ is a vector representation of the matrix P. To achieve this, P is first partitioned as

$$P = \begin{bmatrix} \boldsymbol{p}_1^\top \\ \boldsymbol{p}_2^\top \\ \boldsymbol{p}_3^\top \end{bmatrix}. \tag{18}$$

The expression $P\boldsymbol{U}_k$ can be rewritten as

$$P\boldsymbol{U}_k = \begin{bmatrix} \boldsymbol{p}_1^\top \boldsymbol{U}_k \\ \boldsymbol{p}_2^\top \boldsymbol{U}_k \\ \boldsymbol{p}_3^\top \boldsymbol{U}_k \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_k^\top \boldsymbol{p}_1 \\ \boldsymbol{U}_k^\top \boldsymbol{p}_2 \\ \boldsymbol{U}_k^\top \boldsymbol{p}_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_k^\top & 0 & 0 \\ 0 & \boldsymbol{U}_k^\top & 0 \\ 0 & 0 & \boldsymbol{U}_k^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \boldsymbol{p}_3 \end{bmatrix} \tag{19}$$

Thus the similarity relation between the scene point $\boldsymbol{U}_k$ and the image point $\boldsymbol{x}_k$ can be written as

$$M^{(k)}\boldsymbol{p} = 0, \tag{20}$$

where

$$\begin{aligned} M^{(k)} &= \begin{bmatrix} 0 & -x_{3k} & x_{2k} \\ x_{3k} & 0 & -x_{1k} \\ -x_{2k} & x_{1k} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_k^\top & 0 & 0 \\ 0 & \boldsymbol{U}_k^\top & 0 \\ 0 & 0 & \boldsymbol{U}_k^\top \end{bmatrix} = \\ &= \begin{bmatrix} 0 & -x_{3k}\boldsymbol{U}_k^\top & x_{2k}\boldsymbol{U}_k^\top \\ x_{3k}\boldsymbol{U}_k^\top & 0 & -x_{1k}\boldsymbol{U}_k^\top \\ -x_{2k}\boldsymbol{U}_k^\top & x_{1k}\boldsymbol{U}_k^\top & 0 \end{bmatrix}, \end{aligned} \tag{21}$$

and

$$\boldsymbol{p} = (P_{11}, P_{12}, P_{13}, P_{14}, P_{21}, P_{22}, P_{23}, P_{24}, P_{31}, P_{32}, P_{33}, P_{34})^\top. \tag{22}$$

Note that since $\mathrm{rank}([\boldsymbol{x}_k]_\times) \leq 2$, the rank of $M^{(k)}$ is also less than or equal to two. Consequently, each pair of points introduce up to two linearly independent equations. The system of equations for all k now reads

$$M\boldsymbol{p} = \begin{bmatrix} M^{(1)} \\ \dots \\ M^{(n)} \end{bmatrix} \boldsymbol{p} = \boldsymbol{0}, \tag{23}$$

and contains up to $2n$ linearly independent equations. Since $\boldsymbol{p}$ has 12 degrees of freedom, at least $n = 6$ pairs of corresponding points are required to find a solution to Eq. (23).

The actual solution $\boldsymbol{p}$ to Eq. (23) can be found using e.g. singular value decomposition (SVD) as follows:

$$M\boldsymbol{p} = U\Sigma V^\top \boldsymbol{p} = \sum_{k=1}^{n} \sigma_k \boldsymbol{U}_k (\boldsymbol{V}_k^\top \boldsymbol{p}) = \boldsymbol{0}, \tag{24}$$

where $\boldsymbol{U}_k$ and $\boldsymbol{V}_k$ are the k:th colon vectors of the matrices U and V, respectively. Since the right (and left) singular vectors form an orthogonal basis, $\boldsymbol{V}_i^\top \boldsymbol{V}_j = 0$ if $i \neq j$ and one otherwise. Therefore, assigning $\boldsymbol{p} = \boldsymbol{V}_n$ and inserting it into Eq. (24) gives

$$M\boldsymbol{V}_n = \sum_{k=1}^{n} \sigma_k \boldsymbol{U}_k (\boldsymbol{V}_k^\top \boldsymbol{V}_n) = \sigma_n \boldsymbol{U}_n. \tag{25}$$

If $\sigma_n = 0$, then $\boldsymbol{p} = \alpha \boldsymbol{V}_n$ solves Eq. (23) for any α. Even if the smallest singular value is non-zero, the corresponding singular vector is still used since it provides the best solution to Eq. (23) in a least squares sense. Thus, P can be reconstructed as

$$P = \alpha \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix}. \tag{26}$$

Since the projection of a scene point $(X, Y, Z, 1)^\top$ using P is

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{\alpha(p_9 X + p_{10} Y + p_{11} Z + p_{12})} \begin{bmatrix} \alpha(p_1 X + p_2 Y + p_3 Z + p_4) \\ \alpha(p_5 X + p_6 Y + p_7 Z + p_8) \end{bmatrix}, \tag{27}$$

then any $\alpha \neq 0$ provides a non-trivial solution to Eq. (23) and it can be assumed that $\alpha = 1$.

Lastly, if there are less than 12 linearly independent rows of M, the nullspace of M is at least two-dimensional and the infinite number of solutions can not be decreased to a finite number by setting $\alpha = 1$. Multi-dimensional nullspaces are further studied in Sections 3.2.3 and 3.2.4.

3.2.2 The eight point algorithm

The relation between two pinhole cameras is constrained by the *epipolar constraint* (see e.g. [31]). This constraint can be formulated as

$$\boldsymbol{x}^\top F \boldsymbol{y} = 0, \tag{28}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are homogeneous image coordinates in the first and second camera, respectively, that correspond to the same scene point. The 3×3 matrix F is referred to as the *fundamental matrix* and is used to compute the relative poses of the cameras.
Equation (28) can be rewritten as a scalar product equation

$$\boldsymbol{M}_1^\top \boldsymbol{f} = 0, \tag{29}$$

by first expanding it

$$\begin{aligned} &F_{11}x_1y_1 + F_{21}x_2y_1 + F_{31}x_3y_1 + \\ &+F_{12}x_1y_2 + F_{22}x_2y_2 + F_{32}x_3y_2 + \\ &+F_{13}x_1y_3 + F_{23}x_2y_3 + F_{33}x_3y_3 = 0, \end{aligned} \tag{30}$$

and then defining

$$\begin{aligned} \boldsymbol{M}_1 &= (x_1y_1, x_2y_1, \ldots, x_3y_3)^\top, \\ \boldsymbol{f} &= (F_{11}, F_{21}, \ldots, F_{33})^\top. \end{aligned} \tag{31}$$

Solutions to equation (29) can be found by computing the singular vector decomposition of $\boldsymbol{M}_1$. However, the null space of $\boldsymbol{M}_1^\top$ will be 9 dimensional since it only contains one row. Since $\boldsymbol{f}$ contains 9 elements, and a one dimensional null space desired, is a matrix M is sought that contains 8 linearly independent rows.

Therefore, seven more point correspondences are used to create $\boldsymbol{M}_2, \ldots, \boldsymbol{M}_8$ which are subsequently used to define

$$M = \begin{bmatrix} \boldsymbol{M}_1^\top \\ \ldots \\ \boldsymbol{M}_8^\top \end{bmatrix}. \tag{32}$$

The equation can be rewritten on the form

$$M\boldsymbol{f} = \mathbf{0}, \tag{33}$$

Then the vector representation of the fundamental matrix, $\boldsymbol{f}$, is found by computing the singular vector decomposition of the M, i.e.

$$M\boldsymbol{f} = U\Sigma V^\top \boldsymbol{f} = \sum_{k=1}^{9} \sigma_k \boldsymbol{U}_k (\boldsymbol{V}_k^\top \boldsymbol{f}) = \mathbf{0}. \tag{34}$$

Note that since the $\boldsymbol{V}_k$'s are linearly independent, solutions to the equation are provided by linear combinations of all $\boldsymbol{V}_k$ such that $\sigma_k = 0$. Assume that there are m non-zero singular values, then the solutions $\boldsymbol{f}$ to Eq. (29) are provided by

$$\boldsymbol{f} = \alpha_{m+1}\boldsymbol{V}_{m+1} + \cdots + \alpha_9 \boldsymbol{V}_9, \tag{35}$$

where $\alpha_m, \ldots, \alpha_n$ are arbitrary coordinates in the null space.

Using eight pairwise point correspondences, $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_8\}$ and $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_8\}$, typically gives $m = 8$ non-zero singular values. Consequently,

$$\boldsymbol{f} = \alpha \boldsymbol{V}_9. \tag{36}$$

Since Eq. (29) is unaffected by the scale of F, we can set $\alpha = 1$. F is then given by reshaping $\boldsymbol{f}$ as follows:

$$F = \begin{bmatrix} V_{1,9} & V_{4,9} & V_{7,9} \\ V_{2,9} & V_{5,9} & V_{8,9} \\ V_{3,9} & V_{6,9} & V_{9,9} \end{bmatrix}. \tag{37}$$

Thus there is typically only one solution.

3.2.3 The seven point algorithm

Assume that only seven pairs of points are used to define M (Eq. (32) and Eq. (31)). Then, typically, $m = 7$ and elements in the null space of M are given by

$$\boldsymbol{f} = \alpha_1 \boldsymbol{V}_8 + \alpha_2 \boldsymbol{V}_9. \tag{38}$$

Since Eq. (29) is unaffected by the scale of F, it can be assumed that $\alpha_1 = 1$. Thus there is one unknown, α_2, in the solution $\boldsymbol{f}$. F can now be reconstructed as

$$F = \begin{bmatrix} f_1(1,\alpha_2) & f_4(1,\alpha_2) & f_7(1,\alpha_2) \\ f_2(1,\alpha_2) & f_5(1,\alpha_2) & f_8(1,\alpha_2) \\ f_3(1,\alpha_2) & f_6(1,\alpha_2) & f_9(1,\alpha_2) \end{bmatrix}, \tag{39}$$

for any α_2.
To decrease the number of solutions to a finite number, the additional constraint is $\det F = 0$ is added. This constraint provides a third degree polynomial equation, $\det F = p(\alpha_2) = 0$, in α_2, that can be solved with up to three real solutions.

3.2.4 The five point algorithm

Consider the case where the two cameras are calibrated and that five pairs of points are given in their calibrated coordinate system (see Fig. 9). Then the matrix in the epipolar constraint (Eq. (28)) is a special case of the fundamental matrix – the *essential matrix*. The essential matrix has the additional (compared to the fundamental matrix) properties that two eigenvalues are equal and the third is zero. This constraint is formulated as a matrix equation in the following derivation.

The five point algorithm further decreases the number of points used to define M, and consequently the number of dimensions of the null space increases. Solutions $\boldsymbol{f}$ are now found on the form

$$\boldsymbol{f} = \alpha_1 \boldsymbol{V}_6 + \alpha_2 \boldsymbol{V}_7 + \alpha_3 \boldsymbol{V}_8 + \alpha_4 \boldsymbol{V}_9, \tag{40}$$

where, similar to the seven point algorithm, we can set $\alpha_1 = 1$. To decrease the number of solutions to a finite number, ten constraints are introduced [77],

$$\begin{cases} \det E = 0, \\ 2EE^\top E - \operatorname{tr}(EE^\top)E = 0. \end{cases} \tag{41}$$

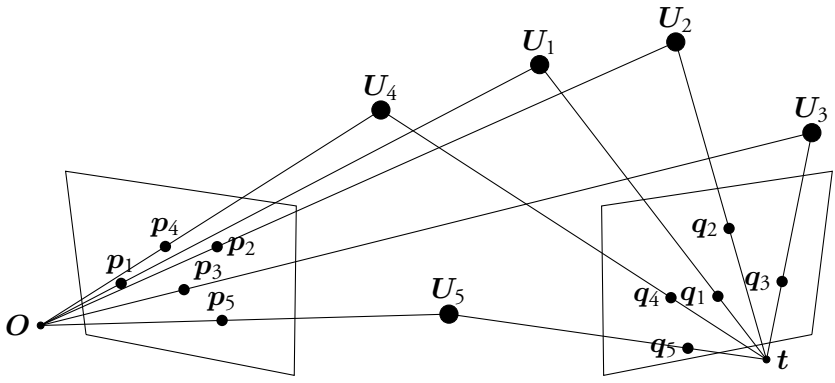


Figure 9: The five point relative pose problem. Five scene points $\boldsymbol{U}_k$ are observed as $\boldsymbol{p}_k$ and $\boldsymbol{q}_k$ in two intrinsically calibrated pinhole cameras. The position $\boldsymbol{t}$ and orientation R for the camera to the right is sought.

As in the previous derivations, one constraint is provided by $\det E = 0$. The second equation is actually a matrix equation that consists of 9 constraints. This system of equations has at most 10 real solutions. For more details, see e.g. [81].

3.2.5 The Misty Three Point algorithm

One of the contributions in this thesis is paper X, in which is proposed two new relative pose algorithms for generalized cameras that only require three pairs of image points that correspond to the same scene point. The Three Point Delta method assumes that, in addition to the points, the difference in distance from each scene point to the camera positions is known. The Misty Three Point algorithm uses three pairs of corresponding points and their respective colors to estimate the relative pose, as shown in Fig. 10. Use of the algorithm is enabled by the distance dependent degradation of light that is observed in underwater images. The method is explained in detail in section 10.2.

3.3 Bundle adjustment

The above methods (e.g. the five point algorithm) estimates the relative motion between only two cameras. However, these algorithms can be applied to estimate the relative poses of several cameras in a sequential algorithm. That is, the relative motion $(R_1, \boldsymbol{t}_1)$ between the first and second cameras, and then $(R_2, \boldsymbol{t}_2)$ between

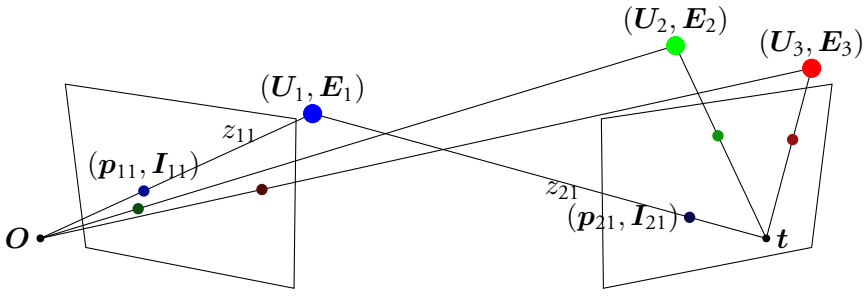


Figure 10: The relative pose problem solved in paper X. Two cameras at different locations and with different orientations observe three points of unknown position and unknown color. Note that we are not only using the direction $\boldsymbol{u}_{ik}$ from each observed point, but also the observed color. The differences in depth $\Delta z_{ik} = z_{jk} - z_{ik}$ are also crucial parts of this method. In the left camera, with larger distance to the objects, the colors look very similar. In the right camera, however, the observed colors are more similar to the actual color of the object. This distance dependent degradation in color is what enables the estimation of depth differences.

the second and third cameras can be estimated. The relative pose of the third camera (in the coordinate system defined by the first camera) can then be computed by combining the two estimated relative poses. This sequential algorithm provides an estimate of the relative pose of camera n that depend on all $n-1$ previous estimates. Therefore, the n'th estimate is subject to the cumulative errors in the previous $n-1$ estimated relative poses. In this section is outlined a widely used method for minimizing the reprojection errors in all cameras simultaneously and removing the cumulative character of the errors.

Let $\boldsymbol{U}_k$ define a set of N_p scene points that correspond to image points $\boldsymbol{x}_{ik}$ observed in N_c cameras, where $k = 1, \ldots, N_p$ and $i = 1, \ldots, N_c$. Let $P_i = [R_i \quad \boldsymbol{t}_i]$ be the extrinsic camera parameters and $\boldsymbol{\Omega}_i$ the set of intrinsic camera parameters of each camera. Bundle adjustment in this case is the simultaneous optimization of each $\boldsymbol{U}_k$, P_i and $\boldsymbol{\Omega}_i$ with respect to the reprojection errors relative to $\boldsymbol{x}_{ik}$. More generally, bundle adjustment can be defined as the simultaneous optimization of scene structure, relative camera motion and intrinsic camera parameters with respect to reprojection errors.

An example of bundle adjustment for a set of intrinsically calibrated pinhole cameras is provided in the following. Assume that the cameras and observed points are normalized. Let $\boldsymbol{f} : \mathbb{R}^M \rightarrow \mathbb{R}^N$ be a vector valued error function that

provides, among others, the reprojection errors of all points. Each scene point and each camera provides a vector of errors $\boldsymbol{f}_{\text{reproj}}(R_i, \boldsymbol{t}_i, \boldsymbol{U}_k, \boldsymbol{x}_{ik})$. There are several possible definitions of such an error function, for example

$$\boldsymbol{f}_{\text{reproj}}(R, \boldsymbol{t}, \boldsymbol{U}, \boldsymbol{x}) = \begin{bmatrix} \frac{R_{11}U_1 + R_{12}U_2 + R_{13}U_3 + t_1}{R_{31}U_1 + R_{32}U_2 + R_{33}U_3 + t_3} \\ \frac{R_{21}U_1 + R_{22}U_2 + R_{23}U_3 + t_2}{R_{31}U_1 + R_{32}U_2 + R_{33}U_3 + t_3} \end{bmatrix} - \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{42}$$

Furthermore, it is of importance that each R_i is a rotation matrix. This can be achieved by for example reformulating the constraint $R^\top R = I_{3\times 3}$ and defining the error function $\boldsymbol{f}_{\text{rot}}(R)$ as

$$\boldsymbol{f}_{\text{rot}}(R) = \begin{bmatrix} R_{11}^2 + R_{21}^2 + R_{31}^2 - 1 \\ R_{11}R_{12} + R_{21}R_{22} + R_{31}R_{32} \\ R_{11}R_{13} + R_{21}R_{23} + R_{31}R_{33} \\ R_{12}^2 + R_{22}^2 + R_{32}^2 - 1 \\ R_{12}R_{13} + R_{22}R_{23} + R_{32}R_{33} \\ R_{13}^2 + R_{23}^2 + R_{33}^2 - 1 \end{bmatrix}. \tag{43}$$

Let $\boldsymbol{Y}$ be a colon vector of length M that contains all the variables. Then, the error function $\boldsymbol{f}$ is defined as

$$\boldsymbol{f}(\boldsymbol{Y}) = \begin{bmatrix} \boldsymbol{f}_{\text{rot}}(R_1) \\ \cdots \\ \boldsymbol{f}_{\text{rot}}(R_{N_c}) \\ \boldsymbol{f}_{\text{reproj}}(R_1, \boldsymbol{t}_1, \boldsymbol{U}_1, \boldsymbol{x}_{11}) \\ \cdots \\ \boldsymbol{f}_{\text{reproj}}(R_{N_c}, \boldsymbol{t}_{N_c}, \boldsymbol{U}_{N_p}, \boldsymbol{x}_{N_c N_p}) \end{bmatrix}. \tag{44}$$

Given an error function $\boldsymbol{f}$, that for each M-vector of variables $\boldsymbol{Y}$ provides an N-vector of errors, the optimization problem can be defined as

$$\boldsymbol{Y}^* = \underset{\boldsymbol{Y}}{\operatorname{argmin}} \|\boldsymbol{f}(\boldsymbol{Y})\|_2 = \underset{\boldsymbol{Y}}{\operatorname{argmin}} (\boldsymbol{f}(\boldsymbol{Y}))^\top (\boldsymbol{f}(\boldsymbol{Y})). \tag{45}$$

A commonly used method for minimizing the nonlinear function $\|\boldsymbol{f}(\boldsymbol{Y})\|_2$ is the Levenberg-Marquardt algorithm. It is an iterative algorithm that, assuming

that an initial vector $\boldsymbol{Y}_0$ is defined, in each step updates the current solution $\boldsymbol{Y}_k$ to

$$\boldsymbol{Y}_{k+1} = \boldsymbol{Y}_k + \boldsymbol{\delta}_k, \tag{46}$$

where $\boldsymbol{\delta}_k$ is an unknown update step at iteration k. The Levenberg-Marquardt method provides a specific method for picking a suitable $\boldsymbol{\delta}$. This is derived as follows. The updated error, $\boldsymbol{f}(\boldsymbol{Y}_k + \boldsymbol{\delta}_k)$, can be linearized as

$$\boldsymbol{f}(\boldsymbol{Y}_k + \boldsymbol{\delta}_k) \approx \boldsymbol{f}(\boldsymbol{Y}_k) + J\boldsymbol{\delta}_k, \tag{47}$$

where J is the Jacobian matrix of $\boldsymbol{f}$, that is, J is a matrix that contains all partial derivatives of $\boldsymbol{f}$. Note that J is a sparse matrix, since most parts of the error functions only depend on a few variables. Setting the updated error to zero thus provides an equation in $\boldsymbol{\delta}_k$ as

$$\boldsymbol{f}(\boldsymbol{Y}_k) + J\boldsymbol{\delta}_k = \boldsymbol{0}. \tag{48}$$

This equation has the least squares solution

$$\boldsymbol{\delta}_k = -\left(J^\top J\right)^{-1} J^\top \boldsymbol{f}(\boldsymbol{Y}_k). \tag{49}$$

The Levenberg-Marquardt algorithm provides a more stable solution to the equation by adding a damping factor $\lambda > 0$,

$$\boldsymbol{\delta}_k = -\left(J^\top J + \lambda \operatorname{diag}\left(J^\top J\right)\right)^{-1} J^\top \boldsymbol{f}(\boldsymbol{Y}_k), \tag{50}$$

where $\operatorname{diag}\left(J^\top J\right)$ is a diagonal matrix where the diagonal elements are those of $J^\top J$.

3.4 Solving systems of polynomial equations

In paper X, an encountered relative-pose problem was formulated as a system of polynomial equations. This section outlines the fundamentals of the theory of polynomial solving that is the most relevant for this particular problem. Furthermore, how the theory is applied to solve this particular system of polynomial equations is outlined. Note that it is assumed in this section that all polynomials are defined over $\mathbb{C}$, but the same theory holds for any algebraically closed field.

Polynomial solvers, and in particular the action matrix method, are important tools in computer vision. Some of the earliest applications of the action matrix method for solving polynomial equations in computer vision were proposed by Stewenius [80]. At that time, most methods were hand-crafted [40, 79]. However, tools have been developed to automatize the method [44]. Furthermore, there have been some developments in making the methods more numerically stable [10, 42] and faster [9, 43]. For an overview of the theory of algebraic geometry, see e.g. [17, 54].

A monomial in the n variables $x_1, \ldots, x_n$ with integer exponents $\alpha_1, \ldots, \alpha_n$ is written on the compact form

$$\boldsymbol{x}^{\boldsymbol{\alpha}} = x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_n^{\alpha_n}. \tag{51}$$

A polynomial $p(\boldsymbol{x})$ in the variables $x_1, \ldots, x_n$ is defined as a finite linear combination of monomials, that is

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} a_{\boldsymbol{\alpha}} \boldsymbol{x}^{\boldsymbol{\alpha}}, \tag{52}$$

where $a_{\boldsymbol{\alpha}}$ is the coefficient for the monomial of the exponential vector $\boldsymbol{\alpha}$ and $\mathcal{A}$ is a finite set of exponents. The set of all polynomials in $x_1, \ldots, x_n$ with coefficients in $\mathbb{C}$ is denoted $\mathbb{C}[x_1, \ldots, x_n]$.
As an example, if $\boldsymbol{x} = (x, y, z)$, $\mathcal{A} = \{(2,1,0), (1,1,1), (0,0,1), (0,0,0)\}$ and $\boldsymbol{a} = \{a(2,1,0), a(1,1,1), a(0,0,1), a(0,0,0)\} = \{\pi, -5, \sqrt{3}, 1\}$, then

$$p(x, y, z) = \pi x^2 y - 5xyz + \sqrt{3}z + 1. \tag{53}$$

A system of polynomial equations in $\boldsymbol{x}$ is a system on the form

$$\begin{cases} p_1(\boldsymbol{x}) = 0, \\ \quad \vdots \\ p_m(\boldsymbol{x}) = 0. \end{cases} \tag{54}$$

The set of solutions $\boldsymbol{x} \in \mathbb{C}^n$ to the system in Eq. (54) is denoted the *affine variety* of the set of polynomials, i.e.

$$V(p_1(\boldsymbol{x}), \ldots, p_m(\boldsymbol{x})) = \{\boldsymbol{y} \in \mathbb{C}^n \quad | \quad p_1(\boldsymbol{y}) = 0, \ldots, p_m(\boldsymbol{y}) = 0\}. \tag{55}$$

Examples of affine varieties in one polynomial are presented in Fig. 13 and Fig. 14, where $V((x/2)^2 + y^2 - z)$ and $V((x/2)^2 + y^2 - z)$ are visualized.

An *ideal* generated by $p_1(\boldsymbol{x}), \ldots, p_m(\boldsymbol{x})$ is the set of all sums of polynomials multiplied with $p_1(\boldsymbol{x}), \ldots, p_m(\boldsymbol{x})$. This can be written as

$$\langle p_1, \ldots, p_m \rangle = \left\{ \sum_{i=1}^{m} h_i q_i \quad \middle| \quad h_1, \ldots, h_m \in \mathbb{C}[x_1, \ldots, x_n] \right\}, \tag{56}$$

where the h_i's are any polynomials over the field $\mathbb{C}$. An important property of ideals is that the affine variety of the ideal is equal to the affine variety of the polynomials that generate the ideal, that is

$$V(\langle p_1, \ldots, p_m \rangle) = V(p_1, \ldots, p_m). \tag{57}$$

Furthermore, an ideal is a *radical ideal* if it contains all polynomials that vanish on $V(I)$. For example, $\langle x^2 \rangle$ is not radical since both $V(x^2) = \{0\}$ and $V(x) = \{0\}$ while $x \notin \langle x^2 \rangle$. Two examples of radical ideals are $\langle x^2 - 1, x + 1 \rangle$ and $\langle x^7, x \rangle$.

Two polynomials p and q are equivalent modulo I if $p - q \in I$, where I is an ideal. The set of polynomials in $\mathbb{C}[\boldsymbol{x}]$ that are equivalent to p is denoted the *equivalence class* of p and is defined as

$$[p] = \{ q \in \mathbb{C}[\boldsymbol{x}] \quad | \quad p - q \in I \}. \tag{58}$$

Furthermore, the *quotient space* $\mathbb{C}[\boldsymbol{x}]/I$ is defined as the set of all equivalence classes modulo I, that is

$$\mathbb{C}[\boldsymbol{x}]/I = \{ f \in [g] \quad | \quad g \in \mathbb{C}[\boldsymbol{x}] \}. \tag{59}$$

The action matrix method is an algorithm that can be used to solve systems of polynomial equations. Assume that a system of polynomial equations have been reformulated by applying row-operations to the form

$$a(\boldsymbol{x})\boldsymbol{b}(\boldsymbol{x}) = M\boldsymbol{b}(\boldsymbol{x}), \tag{60}$$

where $a(\boldsymbol{x})$ is a monomial, $\boldsymbol{b}(\boldsymbol{x})$ is a vector of basis polynomials and M a constant matrix. Then $a(\boldsymbol{x})$ is an eigenvalue and $\boldsymbol{b}(\boldsymbol{x})$ an eigenvector of M. Techniques from linear algebra can be applied to compute the eigenvalues and eigenvectors of M, thus providing potential solutions to the initial system of equations.

The action matrix method is used in chapter paper X to solve the following system of equations,

$$\begin{cases} A_{11}xy & & +A_{14}x + A_{15}y & + A_{17} = 0, \\ A_{21}xy + & A_{22}xz + A_{23}yz & +A_{24}x + A_{25}y & +A_{26}z + A_{27} = 0, \\ & A_{32}xz & +A_{34}x & +A_{36}z + A_{37} = 0, \end{cases} \tag{61}$$

where x, y and z are the unknown variables and $A_{11}, \ldots, A_{37}$ are known coefficients. How these equations can be solved are outlined in the following.

A consequence of Eq. (57) is that if two bases $f_1, \ldots, f_s$ and $g_1, \ldots, g_t$ generate the same ideal, then the varieties over the two bases are equal. That is, if $\langle f_1, \ldots, f_s \rangle = \langle g_1, \ldots, g_t \rangle$, then $V(f_1, \ldots, f_s) = V(g_1, \ldots, g_t)$.
This can be applied to solve Eq. (61) as follows. First, define

$$\begin{aligned} p_1(x,y,z) =& A_{11}xy + A_{14}x + A_{15}y + A_{17} = 0, \\ p_2(x,y,z) =& A_{21}xy + A_{22}xz + A_{23}yz + \\ & A_{24}x + A_{25}y + A_{26}z + A_{27} = 0, \\ p_3(x,y,z) =& A_{32}xz + A_{34}x + A_{36}z + A_{37} = 0. \end{aligned} \tag{62}$$

Since $\{p_1, p_2, p_3\}$ and $\{p_1, p_3, p_2A_{11}A_{32} - p_1A_{21}A_{32} - p_3A_{22}A_{11}\}$ generate the same ideal, the corresponding affine varieties are equal. Note that also Gauss elimination can be applied to systems of polynomial equations. Furthermore, each polynomial can divided by the leading term coefficients without any effect on the ideal it generates. Thus a set of polynomials that generate the same ideal as $\{p_1, p_2, p_3\}$ is provided by

$$\begin{aligned} \hat{p}_1(x,y,z) = xy & + \hat{A}_{12}x + \hat{A}_{13}y & + \hat{A}_{15} &= 0, \\ \hat{p}_2(x,y,z) = xz & + \hat{A}_{22}x & + \hat{A}_{24}z + \hat{A}_{25} &= 0, \\ \hat{p}_3(x,y,z) = yz & + \hat{A}_{33}y & + \hat{A}_{34}z + \hat{A}_{35} &= 0. \end{aligned} \tag{63}$$

Note that for this particular problem, it turns out that the coefficient for x in the third polynomial, $\hat{p}_3(x,y,z)$, is zero. The term is therefore omitted from Eq. (63) without further explanation.

Since $zp_1, yp_2, xp_3 \in \langle p_1, p_2, p_3 \rangle$, then also

$$V(p_1, p_2, p_3) = V(p_1, p_2, p_3, zp_1, yp_2, xp_3), \tag{64}$$

and thus the system can be rewritten as

$$\begin{aligned}
\hat{p}_1(x,y,z) = xy \quad & +\hat{A}_{12}x + \hat{A}_{13}y & & + \hat{A}_{15} &= 0,\\
\hat{p}_2(x,y,z) = xz \quad & +\hat{A}_{22}x & & +\hat{A}_{24}z + \hat{A}_{25} &= 0,\\
\hat{p}_3(x,y,z) = yz \quad & & + \hat{A}_{33}y & +\hat{A}_{34}z + \hat{A}_{35} &= 0,\\
\hat{p}_4(x,y,z) = xyz \quad & & + \hat{A}_{12}xz & +\hat{A}_{13}yz + \hat{A}_{15}z &= 0,\\
\hat{p}_5(x,y,z) = xyz \quad & +\hat{A}_{22}xy & & +\hat{A}_{24}yz + \hat{A}_{25}y &= 0,\\
\hat{p}_6(x,y,z) = xyz \quad & +\hat{A}_{33}xy + \hat{A}_{34}xz & & + \hat{A}_{35}x &= 0.
\end{aligned} \tag{65}$$

Note that $\hat{p}_1$ can be used to reduce monomials containing xy to first degree polynomials in x, y and z. Similarly, $\hat{p}_2$ and $\hat{p}_3$ can reduce xz and yz to first degree polynomials.
Gauss elimination can be used to rewrite the system on the form

$$\begin{cases}
M_{11}y + M_{12}z \quad & +M_{13} = xy,\\
M_{21}y + M_{22}z \quad & +M_{23} = xz,\\
M_{31}y + M_{32}z \quad & +M_{33} = x,
\end{cases} \tag{66}$$

i.e. as the eigenvalue equation

$$a(\boldsymbol{x})\boldsymbol{b}(\boldsymbol{x}) = M\boldsymbol{b}(\boldsymbol{x}), \tag{67}$$

where $a(\boldsymbol{x}) = x$ and $b(\boldsymbol{x}) = (y, z, 1)^\top$. The reason that the equation system in Eq. (61) can be formulated on the form of Eq. (66) is that the dimension of the quotient space for the ideal $\langle p_1, p_2, p_3\rangle$ is finite dimensional (it is in fact three dimensional). Thereby, each eigenvalue λ and associated eigenvector $\boldsymbol{v} = (v_1, v_2, v_3)^\top$ provide a potential solution,

$$(x, y, z)^\top = \left(\lambda, \frac{v_1}{v_3}, \frac{v_2}{v_3}\right)^\top. \tag{68}$$

Note that Eq. (66) has three solutions. However, in this particular application, x, y and z correspond to distances and must therefore be positive. Consequently, the number of solutions is typically decreased.

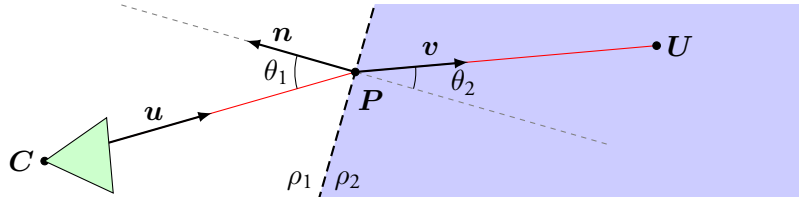


Figure 11: Snell's law. A ray originating from the camera center C with direction $\boldsymbol{u}$ undergoes a change of direction according to $\rho_1 \sin\theta_1 = \rho_2 \sin\theta_2$. This causes the usually linear equations for projections, for example, to become nonlinear and much harder to solve.

3.5 Underwater imaging

When rays of light travel through a medium, they interfere with the molecules in the medium. For example, the speed of light varies depending on the materia. The difference in speed causes light rays entering one medium from another to change direction. This phenomena is called *refraction*. The relation between the angle θ_1 of an impinging ray $\boldsymbol{u}$, with respect to the refractive plane normal, is related to the angle θ_2 of the outgoing ray $\boldsymbol{v}$ by the equation known as Snell's law,

$$\rho_1 \sin(\theta_1) = \rho_2 \sin(\theta_2), \tag{69}$$

where ρ_1 and ρ_2 are the refractive indices of the surfaces, as visualized by Fig 11. See section 9 and paper VIII for more details on the geometrical effects of refractions.

Furthermore, a photon can be absorbed and possibly subsequently released in an arbitrary direction by molecules in the medium. In a homogeneous material, the number of photons being absorbed is proportional to the distance traveled in the medium, i.e.

$$\Phi(z) = -\mu d\Phi(z), \tag{70}$$

where $\Phi(z)$ is the light intensity at depth z and μ is the attenuation coefficient of the material. A solution to 70 is given by

$$\Phi(t) = \Phi(0)e^{-\mu z}, \tag{71}$$

clearly showing the exponential loss of intensity over distance.

The absorption and arbitrarily directed emission of photons also cause an effect known as *backscatter* – photons not originally in the camera's line of sight can be absorbed by a molecule in the line of sight and then released in the direction of the camera. This adds another term in the equation describing the measured light intensity at each pixel. Denote the ambient radiance of water B. Then the observation of the background is

$$B(1 - e^{-\mu z_k}). \tag{72}$$

Thus, the measured light intensity I_k at pixel position k is modeled as

$$I_k = E_k e^{-\mu z_k} + B(1 - e^{-\mu z_k}), \tag{73}$$

which is a the simplified version of the Jaffe-McGlamery equation [35, 37].

4 Video tracking

4.1 Background models and segmentation

A background model is a tool to help segment observed objects in a scene into two classes – background and foreground. What background model is appropriate depend on the application.

For example, in some cases of traffic surveillance, a useful background model describes how a scene without cars, pedestrians, etc. appears. In other traffic surveillance applications, cars can be considered to be part of the background if they have been immobile for a long time (e.g. parked).

One of the simplest background models consists of a *background image* B and a threshold τ. Foreground and background is classified in an image I by applying the threshold on the distance from the pixel values in the image to the background image. That is, each pixel (i, j) in an image is classified as

$$F_{ij} = \begin{cases} 1, & \text{if } |I_{ij} - B_{ij}| > \tau, \\ 0, & \text{otherwise,} \end{cases} \tag{74}$$

where the result of classifying all pixels is the binary foreground image F. Note that also the non-binary *difference image* $F = I - B$ is well suited for certain applications, since it is more smooth than a binary foreground image.

Other models use a statistical framework. For example, the foreground and background in each pixel can be modeled as Gaussian distributions $\mathcal{N}(\mu_{ij}^F, \sigma_{ij}^F)$

and $\mathcal{N}(\mu_{ij}^B, \sigma_{ij}^B)$, respectively. In this model, the probability that an observed pixel (i, j) belongs to the background is $P(I_{ij} \in \mathcal{N}(\mu_{ij}^B, \sigma_{ij}^B))$ and the probability that it belongs to foreground is $P(I_{ij} \in \mathcal{N}(\mu_{ij}^F, \sigma_{ij}^F))$. The classification of pixels as being foreground or background can then be defined as

$$F_{ij} = \begin{cases} 1, & \text{if } P(I_{ij} \in \mathcal{N}(\mu_{ij}^F, \sigma_{ij}^F)) > P(I_{ij} \in \mathcal{N}(\mu_{ij}^B, \sigma_{ij}^B)), \\ 0, & \text{otherwise.} \end{cases} \tag{75}$$

This model can be extended to any type of distributions for background foreground. Note that a native problem of using background/foreground estimation is that objects that have the same color as the background (at the current location) are typically not detected as foreground.

There exists a large number of background estimation algorithms. Many of them consider complicated conditions such as cluttered backgrounds and foregrounds. In the presented papers, however, the experimental set-ups, including the cameras, have been designed to make the conditions for detecting foreground objects as good as possible. Static cameras and static backgrounds are used. In all papers where background estimation has been performed use a similar approach. Each pixel is modeled using a bimodal Gaussian distribution – i.e. a pixel value observed at each point in time belongs to one of two Gaussian distributions. In some of the papers, the actual variation is estimated and subsequent object detection is performed by computing the probability of pixels belonging to the foreground distribution. In other papers, the variation is approximated as being equal and constant for all distributions. In this case, comparing the probabilities of a pixel intensity belonging to foreground or background distributions is equivalent to comparing the distances to the observed pixel intensity.

4.2 Concerning motion models

In the papers included in this thesis, there is an absence of traditional tracking methods. This is motivated as follows. Kinematically-based tracking methods pose constraints on how the tracked objects (e.g. marmoset body, rat paw, etc.) can move.Many applications are well suited for using such models. For example, tracking multiple people in a store for the purpose of knowing which aisles they visit might be aided by a motion model for the purpose of keeping track of individuals. That is, applications where it is more important to know coarsely where subjects are and avoid identity switches, and less important to know *how* the individuals move. However, when attempting to characterize *how* a zebrafish larva

moves – short burst of swimming, quick turns, etc. – movement constraints in the video tracking algorithm would intrude on the data analysis that is to follow. Furthermore, the presented applications in papers I-VI only feature single target tracking, thus identity switches are avoided.

In summary, tracking methods that pose constraints on the movements of the animals are not used in the included papers.

4.3 Quadrics

Quadrics are useful tools in computer vision for describing some basic geometrical shapes, such as spheres and ellipses. Their usefulness is partly due to the property that perspective projections of quadrics surfaces are conic sections, which can then be computed analytically. This thesis contains several applications of quadric surfaces and conic sections. Ellipsoids, spheres and elliptical paraboloids are used to model rigid elements of a rat paw, a food pellet and the nose in paper I and paper II. Ellipsoids are used to geometrically model marmosets in paper III and paper IV. The methods employed to estimate the parameters of the quadrics are outlined in the following sections. Section 4.4 presents methods to generate quadrics with chosen parameters of position, orientation and size. A method to compute the perspective projection of quadrics is outlined in section 4.5. Lastly, methods for fitting quadric surfaces to images is discussed in section 4.6.

A quadratic equation in one variable x is an equation on the form

$$ax^2 + bx + c = 0, \tag{76}$$

where a, b and c are constants and $a \neq 0$. This equation is easy to check for real solutions and to solve. The equation can be reformulated on matrix form as

$$\begin{bmatrix} x \\ 1 \end{bmatrix}^\top \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} = 0. \tag{77}$$

A generalization to N dimensions is denoted a quadric. This can be written on the form

$$\boldsymbol{x}^\top Q \boldsymbol{x} = 0, \tag{78}$$

where $\boldsymbol{x} = (x_1, \ldots, x_{N+1}, 1)^\top$ is a $(N+2)$-dimensional homogeneous coordinate vector and Q is a $(N+2) \times (N+2)$ matrix of constants. Note that

the dimension of $\boldsymbol{x}$ is $N + 2$ and that the non-homogeneous solutions are N-dimensional in a $(N + 1)$-dimensional space. For example, if $N = 1$, then the solutions $(x(s), y(s))$ are curves in $\mathbb{R}^2$ that are parametrized by s. Furthermore, if $N = 2$, then the solutions $(x(s,t), y(s,t), z(s,t))$ are surfaces in $\mathbb{R}^3$ that are parametrized by (s,t).

What geometrical shape the quadric describes depends on the dimensions and values of the matrix Q. As previously noted, a one-dimensional quadric is a curve in the euclidean plane. However, what shape the curve has depends on the values of the elements of Q – the curve may be a parabola, an ellipse or a hyperbola, as shown in Fig. 12. Furthermore, a two-dimensional quadric is a surface in the euclidean space. A few of the possible shapes that Q can represent are ellipsoids, elliptic paraboloids, hyperbolic paraboloids or elliptic hyperboloids, two of which are shown in Fig. 13 and Fig. 14.

4.4 How to generate a quadric matrix

Ellipse

The unit circle can be described by the quadratic equation

$$x^2 + y^2 - 1 = 0, \tag{79}$$

which can be rewritten on the form of Eq. (78),

$$\boldsymbol{y}^\top Q_0 \boldsymbol{y} = 0, \qquad Q_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{80}$$

The circle can be transformed to an ellipse of arbitrary position (x_0, y_0), axis lengths (a_x, a_y) and rotation ϕ as follows. First, the circle is scaled by a_x in x-direction and a_y in y-direction. Then, it is rotated by the angle ϕ, followed by translation by (x_0, y_0).
This corresponds to transforming the *points* on the circle by the transformation matrices

$$T_{\mathrm{s}} = \begin{bmatrix} a_x & 0 & 0 \\ 0 & a_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, T_{\mathrm{r}} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, T_{\mathrm{t}} = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{81}$$

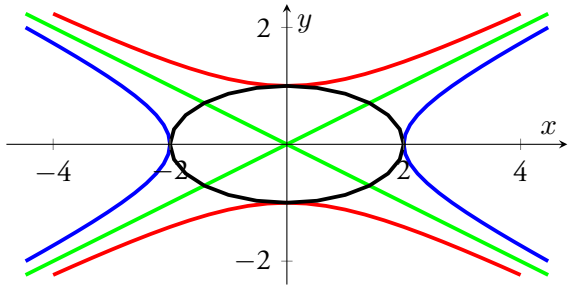


Figure 12: Examples of conic sections. A black ellipse $(x/2)^2 + y^2 - 1 = 0$, blue hyperbola $(x/2)^2 - y^2 + 1 = 0$, red hyperbola $(x/2)^2 - y^2 - 1 = 0$ and green asymptotes $x^2 - y^2 = 0$.

That is, the points $\boldsymbol{y}$ on the unit circle are transformed to

$$\boldsymbol{x} = T_{\mathrm{t}} T_{\mathrm{r}} T_{\mathrm{s}} \boldsymbol{y} = T \boldsymbol{y}. \tag{82}$$

Note that the column vectors of the rotational transformation T_{r} correspond to the direction of the two axes.
The sought quadric matrix Q can now be computed by solving Eq. (82) for $\boldsymbol{y}$ and inserting it into Eq. (79),

$$\left(T^{-1}\boldsymbol{x}\right)^{\top} Q_0 \left(T^{-1}\boldsymbol{x}\right) = \boldsymbol{x}^{\top} \underbrace{T^{-\top} Q_0 T^{-1}}_{=Q} \boldsymbol{x} = 0. \tag{83}$$

In conclusion, $Q = T^{-\top} Q_0 T^{-1}$, where Q_0 is the quadric matrix for the unit circle and $T = T_{\mathrm{t}} T_{\mathrm{r}} T_{\mathrm{s}}$ is provided by Eq. (81).

Ellipsoid

The unit sphere is described by the quadratic equation

$$x^2 + y^2 + z^2 - 1 = 0, \tag{84}$$

which can be rewritten on the form of Eq. (78),

$$\boldsymbol{y}^{\top} Q_0 \boldsymbol{y} = 0, \qquad Q_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}, \tag{85}$$

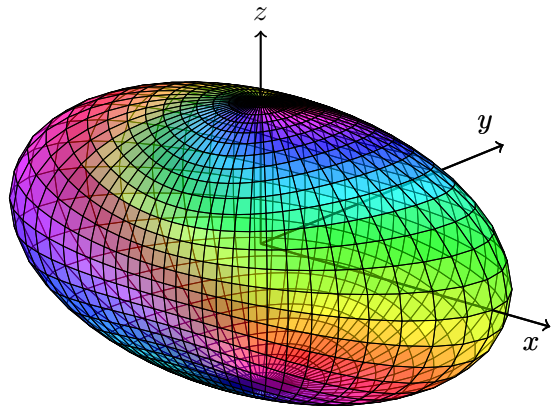


Figure 13: An ellipsoid defined by the equation $(x/2)^2 + (y/1.5)^2 + z^2 - 1 = 0$.

Analogously to the circle, the sphere can be transformed to an ellipsoid $\mathcal{Q}$ of arbitrary position (x_0, y_0, z_0) and axis lengths (a_x, a_y, a_z). In this case, however, the orientation is described by the directions of the axes, $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3\}$, where $\boldsymbol{v}_i^\top \boldsymbol{v}_j = 0$ if $i \neq j$.

Thus, the matrix for the ellipsoid $\mathcal{Q}$ is provided by $Q = T^{-\top} Q_0 T^{-1}$, where $T = T_\mathrm{t} T_\mathrm{r} T_\mathrm{s}$ and

$$T_\mathrm{s} = \begin{bmatrix} a_x & 0 & 0 & 0 \\ 0 & a_y & 0 & 0 \\ 0 & 0 & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_\mathrm{r} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \boldsymbol{v}_3 & \mathbf{0} \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_\mathrm{t} = \begin{bmatrix} 1 & 0 & 0 & x_0 \\ 0 & 1 & 0 & y_0 \\ 0 & 0 & 1 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{86}$$

Elliptic paraboloid

A circular paraboloid is described by the equation

$$x^2 + y^2 - z = 0, \tag{87}$$

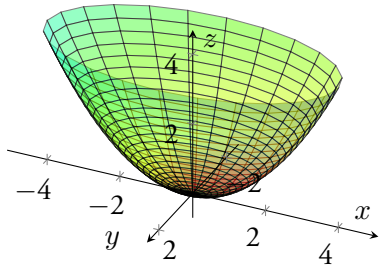


Figure 14: An elliptical paraboloid, $(x/2)^2 + y^2 - z = 0$.

which can be rewritten on the form of Eq. (78),

$$\boldsymbol{y}^\top Q_0 \boldsymbol{y} = 0, \qquad Q_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{2} \\ 0 & 0 & -\frac{1}{2} & 0 \end{bmatrix}, \tag{88}$$

The paraboloid can be transformed to any position, orientation and size by applying the same method that was used on ellipsoids.

4.5 Projections of quadric surfaces

It can be shown that the projection of a quadric surface is a filled conic section. In this section is derived a method for finding the matrix of a conic section that describes the projection of the silhouette of a quadric surface. See also e.g. [78].

First, assume that a quadric matrix Q is given, and let it be partitioned as

$$Q = \begin{bmatrix} A & \boldsymbol{b} \\ \boldsymbol{b}^\top & c \end{bmatrix}, \tag{89}$$

where A is a 3×3 matrix, $\boldsymbol{b}$ a 3-element colon vector and c a scalar. Note that it can be assumed that any quadric surface $\boldsymbol{x}^\top Q \boldsymbol{x} = 0$ is described by a symmetric matrix Q, since $\boldsymbol{x}^\top Q \boldsymbol{x} + (\boldsymbol{x}^\top Q \boldsymbol{x})^\top = \boldsymbol{x}^\top \left(Q^\top + Q\right) \boldsymbol{x} = 0$.

Secondly, to find a point on the silhouette of the quadric, parametrize a ray from the origin through an image point $\boldsymbol{x}$ in homogeneous coordinates,

$$\boldsymbol{U}(t) = \begin{bmatrix} \boldsymbol{x} \\ t \end{bmatrix}. \tag{90}$$

The ray meets the quadric when

$$\boldsymbol{U}(t)^\top Q \boldsymbol{U}(t) = \boldsymbol{x}^\top A \boldsymbol{x} + 2\boldsymbol{b}^\top \boldsymbol{x} t + ct^2 = 0. \tag{91}$$

Note that this is a second order polynomial equation in t, and as such it has at most two real solutions. In this application, the point $\boldsymbol{x}$ for which there is only one unique solution is sought, since such a point lies on the silhouette of the projected quadric surface.

Assume that $c \neq 0$. Then solutions to the equation are provided by

$$t = -\frac{\boldsymbol{b}^\top \boldsymbol{x}}{c} \pm \frac{1}{c}\sqrt{(\boldsymbol{b}^\top \boldsymbol{x})^2 - c\boldsymbol{x} A \boldsymbol{x}}. \tag{92}$$

Eq. (91) has one unique solution when the discriminant is zero,

$$(\boldsymbol{b}^\top \boldsymbol{x})^2 - c\boldsymbol{x}^\top A \boldsymbol{x} = \boldsymbol{x}^\top \underbrace{\left(\boldsymbol{b}\boldsymbol{b}^\top - cA\right)}_{=C} \boldsymbol{x} = 0. \tag{93}$$

Thus the points $\boldsymbol{x}$ that define rays that meets the tangents of the quadric surface can be described by the equation

$$\boldsymbol{x}^\top C \boldsymbol{x} = 0, \tag{94}$$

where

$$C = \boldsymbol{b}\boldsymbol{b}^\top - cA. \tag{95}$$

The depths of a point $\boldsymbol{x}$ on the conic section is provided by inserting $\boldsymbol{x}^\top C \boldsymbol{x} = 0$ into Eq. (92), i.e.

$$t = -\frac{\boldsymbol{b}^\top \boldsymbol{x}}{c}. \tag{96}$$

Note that Eq. (92) only has real solutions when the discriminant is positive. Therefore, the filled conic section that is the projection of the quadric surface is provided by the inequality

$$\boldsymbol{x}^\top C \boldsymbol{x} \geq 0. \tag{97}$$

The above derivations are based on the use of a nominal projection matrix, i.e. $P = [I \quad 0]$. A generalization to arbitrary projection matrices $P = K\,[R \quad \boldsymbol{t}]$ is derived in the following. First, the projection matrix P is expanded as

$$T = \begin{bmatrix} KR & K\boldsymbol{t} \\ \boldsymbol{0}^\top & 1 \end{bmatrix}. \tag{98}$$

Homogeneous points on the quadric surface, i.e. $\boldsymbol{U}$ such that $\boldsymbol{U}^\top Q \boldsymbol{U} = 0$, are transformed to the local coordinate system of the camera by T as $\hat{\boldsymbol{U}} = T\boldsymbol{U}$. The matrix for the quadric surface that contains the transformed points $\hat{\boldsymbol{U}}$, i.e. $\hat{Q}$ such that $\hat{\boldsymbol{U}}^\top \hat{Q} \hat{\boldsymbol{U}} = 0$, is sought. Since $\boldsymbol{U}^\top Q \boldsymbol{U} = 0$ and $\hat{\boldsymbol{U}} = T\boldsymbol{U}$, then

$$\left(T^{-1}\hat{\boldsymbol{U}}\right)^\top Q \left(T^{-1}\hat{\boldsymbol{U}}\right) = \hat{\boldsymbol{U}}^\top \underbrace{T^{-\top} Q T^{-1}}_{=\hat{Q}} \hat{\boldsymbol{U}}. \tag{99}$$

The provided matrix $\hat{Q} = T^{-\top} Q T^{-1}$ is then projected as outlined above.

4.6 Methods for ellipse fitting

Model fitting typically means to find the parameters for which the model is as similar to the observed data as possible. This can be expressed as a mathematical optimization problem,

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, f(\boldsymbol{\theta}, X), \tag{100}$$

where $\boldsymbol{\theta}$ are the parameters of the model, X are the measurements and $f(\boldsymbol{\theta})$ is the error function. Note that this optimization problem can be equivalently defined as the maximization of a goodness-of-fit function.

In the case of fitting a geometrical shape to observed data, the squared sum of euclidean distances is typically used as an error function, i.e.

$$f(\boldsymbol{\theta}) = \sum_{k=1}^{N} \min_{\boldsymbol{x} \in M(\boldsymbol{\theta})} \|\boldsymbol{x} - \boldsymbol{y}_k\|_2^2, \tag{101}$$

where the $\boldsymbol{y}_k$'s are the measured data points and $M(\boldsymbol{\theta})$ is the set of points in the model given the parameters $\boldsymbol{\theta}$. Furthermore, the distance from a data point $\boldsymbol{x}_k$ to the model is defined as the minimal distance from $\boldsymbol{x}_k$ to the model.

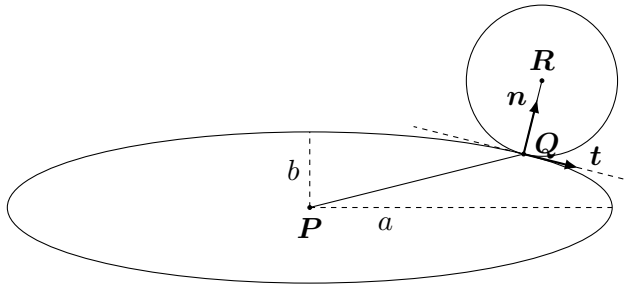


Figure 15: The problem of computing the distance from a point to an ellipse. The ellipse is centered at $\boldsymbol{P}$, and has axis directions as shown in the figure and axis lengths a and b. The distance from the point $\boldsymbol{R}$ to the ellipse is sought. Note that the closest point on the ellipse, $\boldsymbol{Q}$, does not lie on the straight line from $\boldsymbol{P}$ to $\boldsymbol{R}$.

For certain geometrical shapes such as lines, circles and rectangles, the shortest distance between a point and the model can be directly computed. The shortest distance between a point and an ellipse, however, is in general not as simple to compute. A visualization of this is provided by Fig. 15. In the figure is shown an example when the closest point $\boldsymbol{Q}$ on the ellipse is not on the line from the data point $\boldsymbol{R}$ to the center $\boldsymbol{P}$ of the ellipse. To compute the euclidean distance between a point and an ellipse is a problem for which there is no closed form solution. There are, however, iterative algorithms that estimate this distance. Furthermore, there are methods for ellipse fitting that minimize other error functions, for example the algebraic error [25].

4.6.1 A method to measure goodness-of-fit for quadric surfaces

This section provides a outline of the error functions used for fitting quadric surfaces to foreground that are applied in papers I, II, III and IV. Note that all four papers use quadrics as models, but that there are some differences in implementation.

Consider a quadric surface $\mathcal{Q}(\boldsymbol{\theta})$ that is defined by the parameters $\boldsymbol{\theta}$ corresponding to position, rotation and size. The methods outlined in section 4.4 are used to create a quadric matrix Q that represents the quadric surface. Then, the methods presented in section 4.5 are used to compute the projection of $\mathcal{Q}(\boldsymbol{\theta})$ by a known projection matrix. The result is a filled conic section $\mathcal{C}(\boldsymbol{\theta})$ described by

the matrix C,

$$\boldsymbol{x}^\top C \boldsymbol{x} \leq 0. \tag{102}$$

The quadric surface is then fitted to foreground images F by optimizing the parameters $\boldsymbol{\theta}$ with respect a goodness-of-fit function that is defined for the conic section.

In the related applications (papers I, II, III and IV), the goodness-of-fit function between a filled conic section $\mathcal{C}(\boldsymbol{\theta})$ and a foreground image F is defined as

$$q_A(\boldsymbol{\theta}) = \frac{\sum_{x,y} \min(S(x,y|\boldsymbol{\theta}), F(x,y))}{\sum_{x,y} \max(S(x,y|\boldsymbol{\theta}), F(x,y))}, \tag{103}$$

where $S(\boldsymbol{\theta})$ is a binary image such that $S(x,y|\boldsymbol{\theta}) = 1$ if (x,y) is inside the filled conic section and $S(x,y|\boldsymbol{\theta}) = 0$ otherwise. That is, the goodness-of-fit function measures how many pixels that are classified as foreground in both $S(\boldsymbol{\theta})$ and F, divided by the number of pixels that are classified as foreground in either model or data. Note that in paper II, the foreground image F is binary and thereby $q_A(\boldsymbol{\theta})$ provides values between 0 and 1. The remaining three papers use difference images without applying a threshold – this makes the matching quality $q_A(\boldsymbol{\theta})$ depend more smoothly on the parameters $\boldsymbol{\theta}$.
The binary image $S(x,y|\boldsymbol{\theta})$ is generated from $\mathcal{C}(\boldsymbol{\theta})$ and defined as an $M \times N$ image, where

$$S(x,y|\boldsymbol{\theta}) = \begin{cases} 1, & \text{if} \quad \boldsymbol{x}^\top Q \boldsymbol{x} \leq 0, \\ 0, & \text{else}, \end{cases} \tag{104}$$

for all $(x,y) \in [1,\ldots,M] \times [1,\ldots,N]$ and where $\boldsymbol{x} = (x,y,1)^\top$. Thereby, the image $S(\boldsymbol{\theta})$ can be computed by evaluating each of the MN pixels in the image. The result can then be inserted into Eq. (103) to receive a measure of the matching quality. This is the method used in paper II.

Note that the method can be made more computationally efficient by avoiding the independent computation of $S(\boldsymbol{\theta})$, and instead compute the numerator and denominator of $q_a(\boldsymbol{\theta})$,

$$\begin{aligned} f_{\text{num}}(\boldsymbol{\theta}) &= \sum_{x,y} \min(S(x,y|\boldsymbol{\theta}), F(x,y)), \\ f_{\text{den}}(\boldsymbol{\theta}) &= \sum_{x,y} \max(S(x,y|\boldsymbol{\theta}), F(x,y)), \end{aligned} \tag{105}$$

directly. This is described in the following.

Consider the inequality that describes a filled conic section,

$$F(\boldsymbol{a}, (x, y)) = ax^2 + x(by + d) + cy^2 + ey + f \leq 0. \tag{106}$$

That is, a point (x, y) is covered by the filled conic section if $F(\boldsymbol{a}, (x, y)) \leq 0$, and not covered otherwise. Note that for each fixed y, this is a second degree polynomial inequality in x, which in general has zero, one or two intervals as solutions. For the intended applications, however, it is assumed that the constants $\boldsymbol{a}$ of the conic section are such that the solutions x to the inequality are contained in at most one connected set $[x_1, x_2]$.

There exists real solutions x to the inequality only if the discriminant is positive, i.e. if

$$(by + d)^2 - 4a(cy^2 + ey + f) \geq 0. \tag{107}$$

The inequality in Eq. (106) is then solved for each y that satisfies the inequality in Eq. (107). Thereby is provided a either an empty set or a single interval $[x_1, x_2]$ of solutions for each y.

The intersection between the background F and the filled conic section – for a given y – is provided by $I(x_2, y) - I(x_1, y)$, where I is the integral image of F. Furthermore, the union of the background and model can be computed by noting that $|S(\boldsymbol{\theta}) \cup F| = |F| + |S(\boldsymbol{\theta}) \setminus F|$. For a fixed y, $|F|$ corresponds to the interval $x_2 - x_1$, and $|S(\boldsymbol{\theta}) \setminus F|$ corresponds to $(I(M, y) - I(x_2, y)) + (I(x_1, y) - I(M, y))$. Therefore, Eq. (103) can be reformulated as

$$q_A(\boldsymbol{\theta}) = \frac{\sum_{y=1}^{N} I(x_2(y), y) - I(x_1(y), y)}{\sum_{y=1}^{N} (x_2(y) - x_1(y)) + (I(M, y) - I(x_2(y), y)) + (I(x_1(y), y) - I(M, y))}, \tag{108}$$

where $x_1(y)$ and $x_2(y)$ are solutions x to $F(\boldsymbol{a}, (x, y)) = 0$ such that $x_1(y) \leq x_2(y)$. If there are no real solutions to $F(\boldsymbol{a}, (x, y)) = 0$, then $x_1(y)$ and $x_2(y)$ are set to zero. Eq. (108) is summarized in Algorithm 1.

Input: Integral image I and conic parameters $\boldsymbol{a} = (a, b, c, d, e, f)^\top$.
Output: Goodness-of-fit q

$q_{\text{num}} \leftarrow 0$
$q_{\text{den}} \leftarrow 0$
for $y \leftarrow 1, \ldots, N$ **do**
    Evaluate the discriminant $d \leftarrow (by + d)^2 - 4a(cy^2 + ey + f)$
    if $d \geq 0$ **then**
        $\hat{a} \leftarrow a$
        $\hat{b} \leftarrow by + d$
        $\hat{c} \leftarrow cy^2 + ey + f$
        Solve $\hat{a}x^2 + \hat{b}x + \hat{c} = 0$ for x and denote the solutions $x_1 < x_2$.
        $q_{\text{num}} \leftarrow q_{\text{num}} + (I(x_2, y) - I(x_1, y))$
        $q_{\text{den}} \leftarrow q_{\text{den}} + (x_2 - x_1) + (I(M, y) - I(x_2, y)) + (I(x_1, y) - I(x_1, y))$
    else
        $q_{\text{den}} \leftarrow q_{\text{den}} + (I(M, y) - I(0, y))$
    end if
end for
$q \leftarrow \frac{q_{\text{num}}}{q_{\text{den}}}$

Algorithm 1: A method for evaluating ellipse goodness-of-fit.

In addition to the area matching quality, paper I and paper II use an edge matching quality. The two versions of are outlined in the following.

In paper II, an edge image E is computed from the binary foreground image F. Each pixel (x, y) that is classified as foreground (i.e. $F(x, y) = 1$) is considered as part of an edge if there is at least one background pixel in its 8-neighborhood. The direction of the edge at (x, y) is estimated and stored in the edge image as $E(x, y) = -1$ if the angle is between 0 and 180° and $E(x, y) = 1$ otherwise. An edge image $\hat{E}$ that corresponds to the rat model is created analogously from the binary image of the projected quadrics. The edge matching quality is then defined as

$$q_{\text{edge}} = \frac{\sqrt{\max\left(0, \sum_{x,y} E(x, y)\hat{E}(x, y)\right)}}{\sum_{x,y} \left|\hat{E}(x, y)\right|}. \tag{109}$$

Paper I use another definition of the edge matching quality. First, the Canny edge detector (see section 2.3) is used to create an edge image E from an observed image. Then, for each edge point (x, y) on the projected quadrics, the shortest euclidean distance to an edge in E is computed. The edge matching quality is then defined as

$$q_E = \frac{1}{|\partial S|} \sum_{(x,y)\in\partial S} \frac{1}{D(x,y)+1}, \tag{110}$$

where ∂S is the set of edge points on the projected quadrics and $D(x, y)$ is the shortest distance from the point (x, y) to an edge pixel in E.

Method development

5 Robust pose estimation for animals

Human pose estimation is a topic that has been studied extensively [5, 68, 71, 76]. Applications include for example human-computer interaction [36, 67], markerless motion capture [21, 75] and clinical gait analysis [63, 92]. There exists datasets and challenges in which researchers can evaluate the performance of their algorithm. One example is the Human 3.6M dataset by Ionescu et al. that contains 3.6 million human poses in different scenarios [13, 34]. In contrast, pose estimation in any other kind of animal has been largely neglected by the computer vision community. As a consequence, the majority of the papers included in this thesis consider problems where there are neither established challenges nor datasets that allows for comparison between different solutions.

Even though parts of the theory and methods developed for humans can be applied for other animals, there are some practical differences to consider that depend on the animal and application. These similarities and differences, as well as proposed solutions, are outlined in this section.

The problem of estimating the pose of an animal depends on the geometry and texture of the animal, the shape and color of the arena, the light conditions, etc. For example, in the included papers, zebrafish larvae only rarely cause self-occlusion (from the camera's point of view), while the marmosets regularly occlude their legs, arms and/or tail significantly. Furthermore, which geometrical model is suitable for an animal depends on the application and the video or image data. The possibilities for the video(s) depend on the properties of the data acquisition system – for example number of cameras, video resolution, video frame rate and sensor type (color, IR, depth, etc.). It also depends on factors such as staining (for example, *Daphnia* are colored by quantum dots in papers VIII and IX) and

the tracking arena (flat or 3D, small or large, shape, etc.). Thus, the chosen model is a compromise between what information is possible to extract from the video(s) and what data are beneficial for the subsequent behavioral analysis.

For example, a deformable mesh model is used by Fontaine et al. [26] on high resolution videos of zebrafish larvae, while a skeleton model that consists of eight points is used on low resolution videos in paper V and paper VI. Additionally, the turning behavior of marmosets is of interest for the neurophysiological study in paper IV, while only the positions of the *Daphnia* is necessary for the analysis in paper IX.

For example, the common marmoset is to a large degree similar to humans. Most notably, it has a head, a similarly shaped torso, two legs, two arms, elbows, knees, hands and feet. The major difference in a skeleton model of a marmoset compared to a human is the tail (which can simply be added to the model). The differences in shape and size of corresponding body parts are minor and can be adjusted for. Consequently, an adaptation of algorithms in articulated human pose estimation [5, 68, 71, 76] for marmosets could potentially provide good results. However, this is generally not the case due to a non-geometric difference in marmosets – they move and behave differently than humans. For example, in applications such as in paper III and paper IV, the arms and legs of the animal are rarely seen separated from the body. In other words, the poses that are sought to be estimated are more challenging than the commonly considered poses in human pose estimation. Thus for paper III and paper IV, a more suitable geometrical model consists of only the torso and tail.

5.1 Related work

Rat tracking

There are many proposed methods for tracking of rats (*Rattus*) and mice (*Mus*) [7, 20, 27, 46, 93], commonly with the intended application of studying social behavior. In common for all of these methods is that they track the whole body of the animals. This clearly differs from our proposed methods (paper I and paper II) that track the articulated 3D-pose of the front paws of rats.

A popular subject within the field of rat tracking is whisker tracking [15, 28, 89]. Even though finer movements of rats are tracked, there is no tracking of paws.

However, methods for tracking rat paws have been proposed in the context

of gait analysis [22, 29]. These methods use images of the footprints of the paws to analyze gait patterns. Note that this method is only applied for the case where the paws touch the ground. Our method, however, estimates the 3D-poses of rat paws while in the air during grasping motions.

In conclusion, our proposed methods (paper I and paper II) track the articulated front paws of the rats during a reaching task. To our knowledge, there is no previous automatic method that tracks the articulated poses of front paws of rats.

Marmoset tracking

The common marmoset (*Callithrix jacchus*) is a widely used non-human primate in neurophysiological and medical research [16]. Despite the widespread use of this animal, there is a lack of proposed methods for tracking of marmosets. To our knowledge, there are no publication that predates our proposed methods (paper III and paper IV). Since the proposals of our algorithms, there have been two additional publications on this topic [8, 87].

Zebrafish tracking

Zebrafish (*Danio rerio*) larvae is a commonly used test animal primarily in medical research. Its widespread use is motivated by the fact that it is a vertebrate, it is small and easy to maintain and can be tested in large numbers. Furthermore, it is particularly useful in drug screening applications since it absorbs medical substances directly from the water it is swimming in.

There are several previously proposed methods and commercial products for the tracking and behavioral analysis of zebrafish (larvae). The methods generally focus on either quantity or quality, i.e. to analyze large numbers of fish/larvae at a coarse scale or small number or single fish/larva at a fine scale. The parameters of recording duration, video resolution, frame rate, number of fishes and well size need to be tuned accordingly.

For example, Fontaine et al. record single fish at 1500 hz with 1024×1024 pixels resolution [26]. The high spatial resolution gives the possibility to fit a detailed geometric model to the fish and the high temporal resolution gives the possibility to analyze fine movements. However, the recordings are limited to short durations due to the high data rate.

There are also methods that estimate the 3D position of zebrafish. For example, Tian et al. and Cachat et al. use two cameras to estimate 3D-poses and

trajectories [11, 86] of single fish.

Other methods are focused more towards high throughput. For example Creton track the position and orientation of larvae in 12 well microtiter plates [18] with a spatial resolution of 9 MP.

Our proposed method (paper V and paper VI) estimates the positions and skeleton curves of multiple larvae simultaneously in low resolution videos. The method is applied on videos recorded at 300 hz with 640×480 resolution. In these videos, the larvae are approximately 10 pixels long. The proposed method provides a novel compromise between quality and quantity.

Jellyfish tracking

Tripedalia cystophora is a box jellyfish that is unique among other jellyfish since they have eyes. A total of 24 eyes are divided into four clusters that are denoted *rhopalia*.

In paper VII is proposed a method for real-time tracking of the position of the rhopalia in an experiment where the bell of the jellyfish is fixated.

Daphnia tracking

The zooplankton *Daphnia magna* is a small aquatic organism that inhabits freshwater environments and is an important part of the ecosystem. Therefore, the behavior of *Daphnia* is of great interest in biology. *Daphnia* are very small (up to 5 mm long) and translucent. Consequently, there are some issues to consider when designing a system to track its behavior.

Kunze et al. track the movements of multiple *Daphnia* in 2D by applying Kalman filters to the tracked contours of images after subtracting the estimated background [45].

The group of L-A. Hansson propose a method where the *Daphnia* are colored using quantum dots [48] and tracked in 2D. The method is improved upon by Ekvall et al. and Bianco et al. [4, 23] by e.g. tracking the *Daphnia* in 3D. Note that in these papers, the refractions caused by the aquarium glass and water are not modeled.

Our proposed method further improves upon the quantum dot based methods [4, 23, 48]. We propose the first, to our knowledge, 3D-tracking system for *Daphnia* that accurately model the geometric effects caused by refractions and reflections (paper VIII and paper IX).

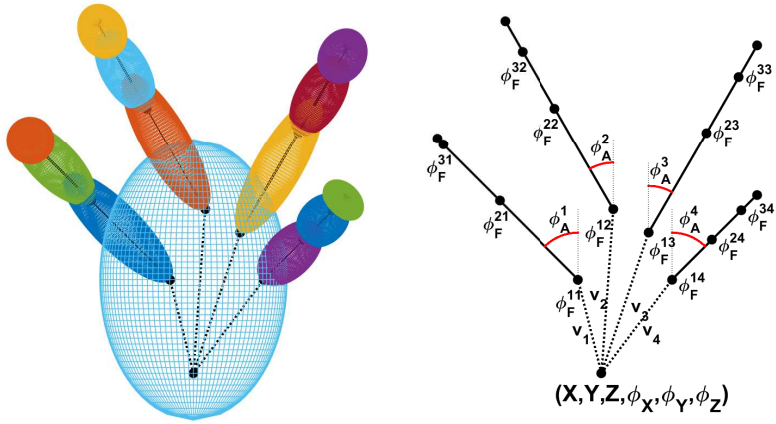


Figure 16: The ellipsoid based hierarchical rat paw pose model used in [56]. The position of the paw is (X, Y, Z) and the orientation is defined by the angles (ϕ_X, ϕ_Y, ϕ_Z). Digit k is described by the angles of adduction ϕ_A^k and flexion ϕ_F^{1k}, ϕ_F^{2k}, ϕ_F^{3k}. The sizes of the ellipsoids and the metacarpal vectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, $\boldsymbol{v}_3$ and $\boldsymbol{v}_4$ are constant.

5.2 Rats

In the skilled reaching experiment, the articulated poses of rat paws are sought to be analyzed. From a neurophysiological point of view, it is of interest to explore potential correlations between neuronal activity and the fine kinematics of the paw during reaching. It is our hypothesis that the angles of the digits and, in particular, the dynamics between those angles and the position of the paw are of interest for this task.

5.2.1 The model

The rat paw is modeled hierarchically such that the pose of the palm is defined in the global coordinate system and the phalanges (parts of the digits) are defined locally. A proximal phalanx can rotate along two axes – the axis of adduction/abduction and the axis of flexion/extension (the ϕ_A's and ϕ_F's, respectively, in Fig 16). The remaining phalanges are limited to rotation by their flexion/extension axes. Due to difficulties in estimating the poses of the metacarpal

bones (the bones inside the palm), the corresponding vectors from the palm position to the proximal phalanges are manually set to constant values for each animal. Furthermore, it is assumed that the sizes of the phalanges are constant. Using all of the above variables and constants, the global pose of each phalanx can be computed.

In total, the articulated pose of a rat paw is defined by 22 variables: $3 + 3$ for the pose of the palm and $2 + 1 + 1$ for each digit. Each of these variables is anatomically valid and believed to be of use for subsequent neural and behavioral analysis. However, note that the angle of flexion/extension in the most distal joint of each digit is not actually estimated. This is because the most distal phalanx is so short and difficult to estimate – these angles are instead considered as a function of the flexion angle of the second joint on each digit [47].

5.2.2 Pose estimation

Four cameras are used to record the experiments, where the result is frames such as shown in Fig. 17 and Fig. 32. The cameras are calibrated using DLT (see section 3.2.1). For each frame in a video, the quadric parameters that minimize the error function are estimated through an iterative method. The error function is defined as in section 4.6.1 and the iterative scheme is outlined in paper I and paper II.

5.3 Marmosets, box

In the box experiment, the locomotive behavior of marmosets inside a box is the target of study. Measurements of movement speed and overall activity is of particular interest for the neurophysiological study.

Due to the difficult problem of estimating the poses of arms and legs sufficiently accurate, a simple model that can be tracked robustly is needed. For this purpose, the marmoset is modeled by a single ellipsoid of variable position, orientation and size.

Two cameras are used to record the experiments, and examples from the video data are shown in Fig. 18 and Fig. 35. The cameras are calibrated using DLT (see section 3.2.1).

For each frame in a video, the parameters of the quadric surface for which the error function is minimized are sought. The error function is defined by Eq. (103) and the iterative estimation method is outlined in paper III.

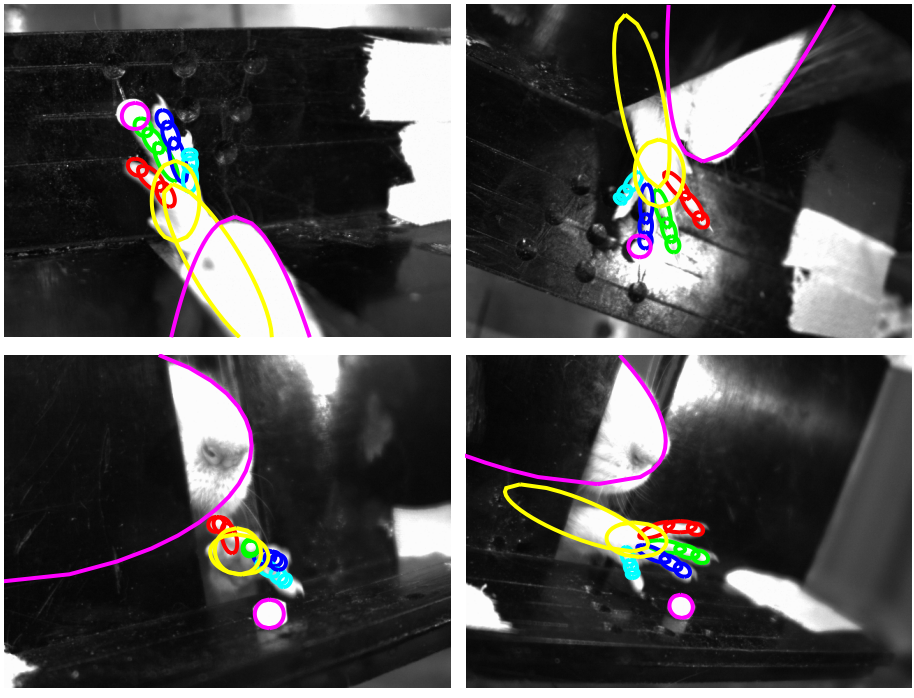


Figure 17: Images from the four cameras used in the reaching experiment. The estimated paw model has been projected and plotted in the images. The digits are plotted in red, green, blue and cyan, and the palm and forearm in yellow. The snout and pellet are both plotted in magenta.

5.4 Marmosets, tower

The setup of the tower experiment consists of a 2.2 meter high testing chamber in which 7 horizontal sticks are placed at different heights. The main locomotive goal of the experiment is to characterize vertical locomotion behavior of marmosets. Thus only the position of the marmoset at each time frame is of interest to analyze. The animals are modeled as single ellipsoids in this set-up as well.

A single camera is used to record the experiments, where the result is frames such as shown in Fig. 19 and Fig. 34. The camera is calibrated using DLT (see section 3.2.1).

The same error function and the same method for fitting ellipsoids that was used for the box experiments is used for the tower experiment. That is, the error

Figure 18: Images from two cameras that correspond to the same point in time during a marmoset box experiment. Model fitting results have been superimposed in blue.

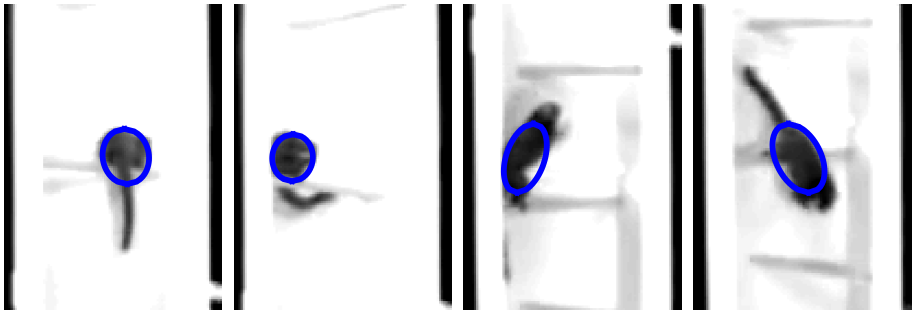


Figure 19: Images from four points in time during a marmoset tower experiment. Model fitting results have been superimposed in blue.

function is defined by Eq. (103) and the iterative estimation method is outlined in paper III. Note that since only one camera is used, it is assumed that the z-position of the animal is fixed, and that rotation only occurs along the z-axis. That is, the z-position and rotation along the x and y-axes are not estimated.

5.5 Zebrafish

The purpose of the zebrafish experiment is to evaluate the effect of drugs that target the central nervous system, for example drugs that could potentially be used in treatment of schizophrenia or anxiety.

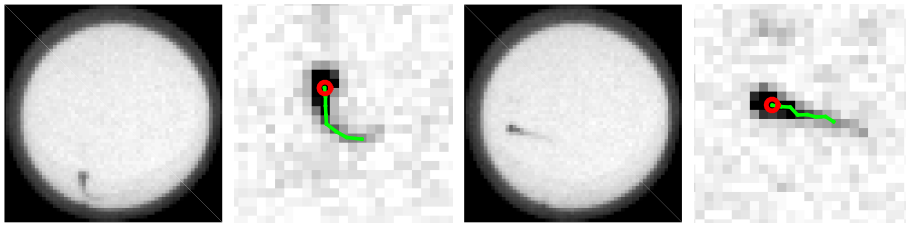


Figure 20: Two pairs of images from two points in time during a zebrafish experiment. The leftmost image in each pair is the original video. The rightmost images focus on small regions around the larvae. Model fitting results have been superimposed as a red circle (head) and green curve (body).

5.5.1 The model

In the recorded videos, the zebrafish larvae appear as thin lines or curves (see Fig. 20). The typical observation of a larva is approximately ten pixels long, three to four pixels wide at the head and thinner towards the tail. Therefore, a suitable model for the larva in this application is a curve with no width and where the start (i.e. the head) is at the thicker side of the curve.

5.5.2 Pose estimation

The pose of a zebrafish larva is estimated in a two-step process. First, the position of the head is estimated as the position of the maximum intensity (in the region of the image where the larva can be located). The position of maximum intensity is computed robustly by first applying a Gaussian filter. Furthermore, the initial estimated pixel position of the head is improved upon by fitting second order polynomials in the x and y-directions. The head position is defined as coordinates of the maxima of the fitted polynomials. Thus the head position is estimated in sub-pixel resolution. Then, the curve of the body is estimated as the positions of maximal intensities on concentric circles, centered at the head position. Similarly to the head position, these are computed in sub-pixel resolution, but by fitting a polynomial on each circle.

5.5.3 Additional results

In addition to the results presented in paper V and paper VI, a test was conducted on the quality of the pose estimation results. For that purpose, 500 frames were

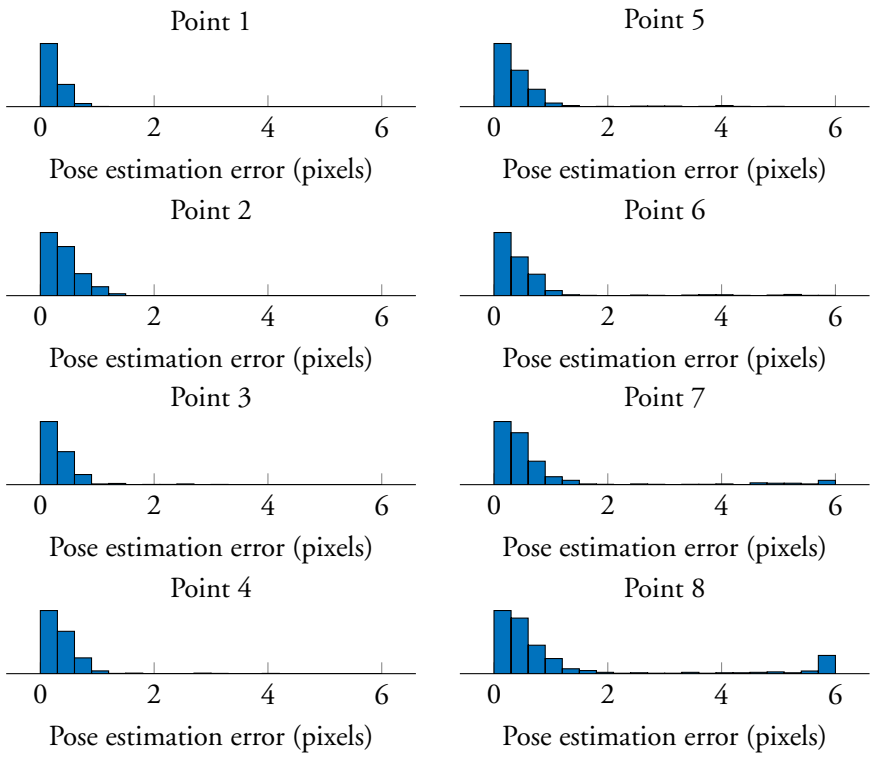


Figure 21: Distributions of pose estimation errors for each of the eight tracked points on zebrafish larvae. The tracking error of a point is defined as the Euclidean distance from the point to the spline fitted to the ground truth points.

randomly selected. In each frame, a larva was selected at random, and points on the zebrafish larva were manually selected. A spline was fitted to the selected points, and then the distance from the pose estimation results were computed. The results of this evaluation are presented as eight histograms – one per point on the estimated skeleton model of the larvae – in Fig. 21. Note that the error for the head ("Point 1") is typically less than one pixel, and that the errors further out on the tail are larger but typically less than 2 pixels.

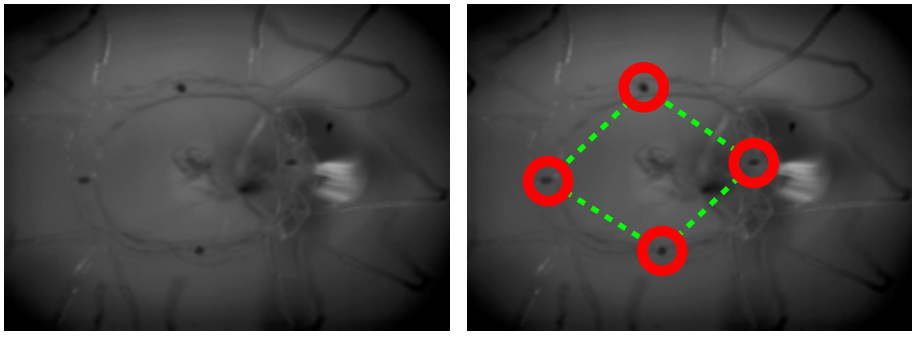


Figure 22: An image from a video of box jellyfish. Model fitting results have been superimposed as red circles (rhopalia) and dashed green lines (body).

5.6 Jellyfish

The box jellyfish is transparent and, consequently, challenging to estimate the full body pose of. Fortunately, it has four clusters of eyes, *rhopalia*, that are easier to detect (see Fig. 22). These are positioned approximately equidistantly on the bell, thus making a good indication of how the actual bell is moving. The rhopalia mostly appear as circular blobs, and there is no interest in analyzing their shape in the intended application. Therefore, the box jellyfish is modeled as consisting of four circular rhopalia.

The experiments on the box jellyfish are recorded by a single camera, and no methods of calibration are applied in the included paper (paper VII). In the included paper, a system is proposed that detects and tracks the rhopalia. For a description of the system, see paper VII.

5.7 *Daphnia*

As previously noted, what model is suitable for an animal depend on for example the video data. In this application, another factor is of importance – namely staining. The zooplankton *Daphnia* is a small and transparent animal that is hard to see unless viewing at a high resolution. A solution that allows for the observation of *Daphnia* at greater distance is given by using quantum dots to color the *Daphnia* [48]. In the videos of the presented application (papers VIII and IX), the *Daphnia* are small, but clearly visible as small dots, as presented in figure 23. A suitable model in this case is as a dot with color. Potential *Daphnia* positions

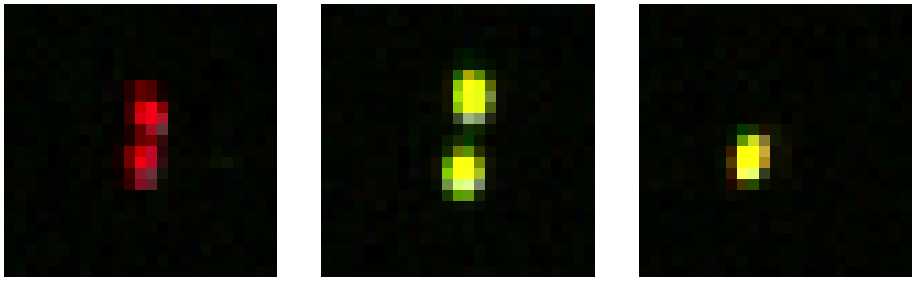


Figure 23: Examples of frames from a video of *Daphnia*. In the leftmost and center images can be seen both a direct view and a reflection. The rightmost image only contains the direct view.

are found in the videos at the locations of local maxima that have colors similar to the introduced quantum dot colors. Note that the coloring by quantum dots can potentially simplify tracking of multiple *Daphnia* if they are assigned different colors.

The *Daphnia* experiments are recorded by four cameras, which are calibrated using the methods outlined in section 9 and paper VIII.

6 Robust feature extraction from frame-by-frame pose estimation data

As explained and motivated in Section 4.2, the outputs from the tracking systems outlined above are not constrained by motion models. Consequently, the tracking data are not necessarily smooth and may contain outliers. Methods for extracting features from such frame-by-frame pose estimation data need to be robust to the characteristic noise from the tracking system to avoid inducing errors in the behavioral analysis. An example of this is provided by Fig. 24, where the tracked head position of a zebrafish larva is plotted. In this figure can be seen time intervals that contain pixel-sized oscillatory noise (Fig. 24b) during a movement and larger oscillatory noise (Fig. 24c) while stationary. A method that provides kinematic features (such as distance or speed) for behavioral analysis need to be robust to such noise in order to give reliable data. Failing to do this may lead to vastly different results, as shown in Section 6.3.1. Furthermore, the extracted features should be useful for subsequent behavioral analysis.

This section provides an outline of the methods for robust feature extraction that are proposed in the included papers on rats, marmosets and zebrafish. The subsequent analysis of the extracted features is commented upon in Section 7, but mostly contained in the included papers.

Note that this thesis does not contain any proposed methods for extracting or analyzing kinematic features from experiments on box jellyfish. Furthermore, no methods are proposed for extracting kinematic features from the frame-by-frame tracking data of *Daphnia*. The analysis of *Daphnia* behavior was performed using the nlme package [66] and R [85].

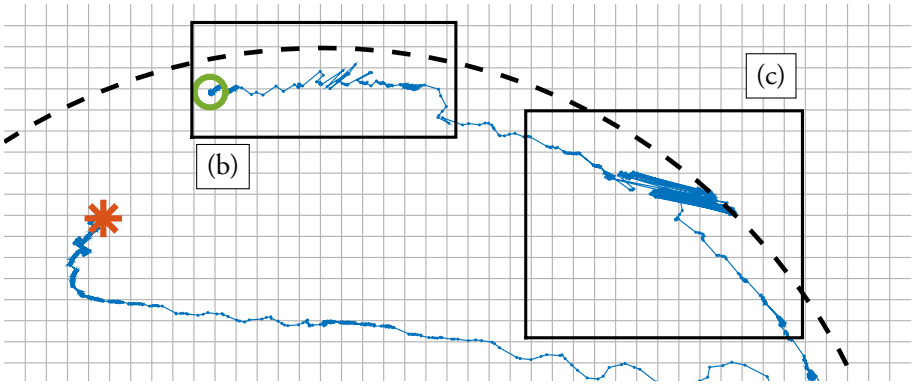
6.1 Reaching

The part of the reaching data that corresponds to the paw pose is represented by a 19-dimensional vector, that contains the angles of flexion/extension (8 values), adduction/abduction (4 values), wrist position (3 values), palm orientation (3 values) and a deformation value of the paw. Note that additional values (such as nose, pellet, etc.) are also tracked, but are not subject to analysis here.

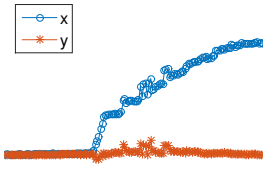
6.1.1 Motor components

The reaching movement can be seen as separated into three different phases – advance, arpeggio and grasp [69]. First, the paw is advanced through the slit and usually far out into the scene. At this moment, the paw is typically at least 45 degrees from being parallel to the shelf. Then, the paw is moved down, towards the shelf. As it hits the shelf, the palm turns until it is parallel to the shelf. This downwards and turning phase is denoted *arpeggio*. Then, the rat clenches the paw in a *grasping* movement. After these three phases, the paw is withdrawn from the scene, with or without food.

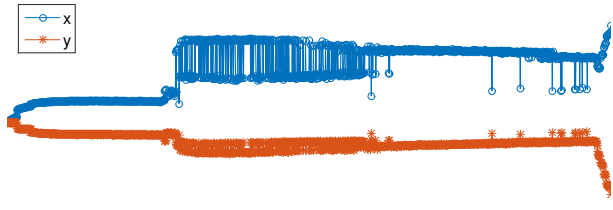
In the included papers, detectors for advance, arpeggio and grasp were constructed by manually selecting intervals corresponding to each of the phases in a number of reaching attempts. Then the mean value of those were used to create advance, arpeggio and grasp templates of length 6 and 8, respectively. The detectors consist of template matching by correlation maximization, using the semi-automatically defined templates.



(a) The tracked head position (blue) of a zebrafish larva. The dotted black line gives an indication of where the wall of the well is. The green circle and the red asterisk denote the start and end, respectively, of the 16 seconds long interval. Two intervals are studied in detail in (b) and (c) – these intervals are marked in the figure as black rectangles.



(b) Small oscillatory noise while moving.



(c) Oscillatory noise while stationary.

Figure 24: Example output from the zebrafish tracking algorithm. (a) shows the tracked head position in 2D. (b)–(c) shows the tracked x and y-coordinates of the head independently, for two intervals of different tracking quality. (b) shows an interval where the larva is moving and there is noise of size $\sim$ 1 pixel. In (c), there can be seen oscillatory noise while the larva is (most likely) not moving.

6.2 Marmosets

The frame-by-frame tracking data of a marmoset is on the form

$$\{\dots, (\boldsymbol{x}_k, \boldsymbol{\psi}_k, \boldsymbol{a}_k), \dots\}, \tag{111}$$

where, for each k, $\boldsymbol{x}_k$ is the position in 2D or 3D, $\boldsymbol{\psi}_k$ is the orientation and $\boldsymbol{a}_k$ are the axis lengths. However, the orientation and axis lengths are not actually used in the presented applications (paper IV and III) and therefore discarded for

the remainder of this section. Note that the animals are modeled as ellipsoids in both experiments, but that the z-position and rotation along the x and y axes are fixated for the tower experiment. Therefore, the tracked position and angle in the tower experiment are considered in 2D.

6.2.1 Walk bouts

Even though marmosets do not have as clearly discretized movement as e.g. the zebrafish larvae, their full body movements can be discretized into bursts of movements nonetheless. Such movements are here referred to as *walk bouts*, and are defined analogously to swim bouts (see section 6.3.2). For this application, the set of time differences used in Eq. (116) are given, in seconds, by $\mathcal{I} = \{0.25, 0.50, 1.0, 1.5\}$.

6.2.2 Features of the walk bouts

The separation of movements into walk bouts provide tools to extract meaningful behavioral data. For example, the maximal movement speed, distance traveled and duration can be measured for each bout. Furthermore, how often the animal initiates a walk bout is an informative measure.

An important aspect of the studied marmoset model of Parkinson's disease is that it should decrease the speed of the animals. For that purpose, the *maximal* speed for each walk bout was measured in paper IV. A robust measure of speed that gives qualitatively good results is achieved by applying a median filter (see e.g. [3]) on the tracked sequence of positions, followed by estimating speed as

$$v(t_k) = v_k = \frac{\|\boldsymbol{x}_{k+N} - \boldsymbol{x}_k\|_2}{t_{k+N} - t_k}, \tag{112}$$

where N is defined such that the time difference $t_{k+N} - t_k$ is 0.1 seconds. The speed function $v(t)$ is then filtered by a Gaussian filter, and the maximal speed is defined as the maximal value of the filtered speed function.

In addition, the distance traveled in each walk bout is also expected to decrease after inducing Parkinsonism in the animals. A robust measure of distance that gives qualitatively good results is achieved by fitting splines to the positions, using the estimated goodness-of-fit as spline weights. The distance is then defined as the sum of euclidean distances between all subsequent times. Note that this measure of distance is different from the measure proposed in section 6.3.1. The method

that is applied for measuring distance in marmosets has a higher precision and is convenient to use here. However, this method is badly suited for the zebrafish data, since it is more sensitive to noise. Furthermore, the method is much slower compared to the applied method. This is of practical importance since the vector of tracked head positions for each of the 48 zebrafish is approximately 40 times longer.

Lastly, the duration of a walk bout is defined as the length (in seconds) of the detected walk bout interval.

6.2.3 Levels in the tower experiment

The tower is a narrow and high testing chamber that has 7 sticks at varying heights, as shown in Fig. 34. The marmosets are believed to prefer to be positioned relatively high up in the tower. However, the distance between sticks increases as the height over the floor increases. It is hypothesized that Parkinsonian marmosets are less likely to be located higher up. For this purpose, the distribution of time that the marmosets spend at the different sticks is estimated. This is straight forward to compute using the tracked y-coordinate of the animals.

Furthermore, the marmosets do not always transition between the closest sticks – they sometimes travel multiple levels at a time. If the ability to move is impaired in the Parkinsonian marmosets, they should not only stay at the lower levels – they should also not jump between multiple sticks at a time. This effect is not captured by the above mentioned distributions of time on each level. Therefore, the level transitioning behavior is also characterized, where the transitional probabilities are measured using the maximum likelihood estimate.

6.3 Zebrafish

The position data provided by the zebrafish tracking algorithm (see section 5.5) is on the form

$$\{\dots, (\boldsymbol{x}_{1k}, \dots, \boldsymbol{x}_{8k}), \dots\}, \tag{113}$$

where, for each time index k, $\boldsymbol{x}_{1k}$ is the head position and $\boldsymbol{x}_{jk}$ are points on the tail that are increasingly more distant as j increases. In the proposed methods for analysis, however, this detailed data on the shape of the larvae is simplified. Instead, a straight line is fitted to the 8 points and the angle of the line is used.

Thereby, the zebrafish tracking data that is considered for the remainder of this section is on the form

$$\{\dots, (\boldsymbol{x}_k, \phi_k), \dots\}, \tag{114}$$

where $\boldsymbol{x}_k$ is the head position and ϕ_k is the angle.

6.3.1 Distance

The distance between two points is easily defined – for example, the euclidean distance is often suitable. However, the traveled distance of an object along a trajectory is not as simple to define. For example, consider the trajectory shown in Fig. 24a. The length of the trajectory could be defined as the euclidean distance between the start point and end point. Or it could be defined as the sum of euclidean distances between all subsequent points in the trajectory. Neither of these definitions is practically useful – the former measure is too coarse and subject to noise in the start and end points, and the latter is too sensitive to e.g. oscillatory noise in the data.

To design a measure for distance that is robust to the typical noise in this tracking data, the noise first needs to characterized. For the zebrafish tracking data, the noise is separated into two types. First, the tracking errors that may occur when a larva is close to the wall are modeled as independent frame-to-frame jumps of at most $\sigma = 6$ pixels between two positions. Secondly, the tracking errors encountered when a larva is completely occluded larvae are seen as frame-to-frame jumps of more than σ' pixels. One way of constructing a metric that is robust to such noise is to create a sequence that starts at the index $i_1 = 1$, and then add the next point that is at least σ and less than σ' pixels away from the previous point. Thus is created a sequence of points $\{\boldsymbol{x}_{i_1}, \dots, \boldsymbol{x}_{i_N}\}$, where $\|\boldsymbol{x}_{i_{k+1}} - \boldsymbol{x}_{i_k}\|_2 \geq \sigma$ for each $k = 1, \dots, N-1$.

Note that σ can be decreased by first applying, for example, a rectangular mean value filter to the tracked x and y coordinates. Since the oscillatory noise is modeled as time independent jumps between two positions $\boldsymbol{p}_0$ and $\boldsymbol{p}_1$, a rectangular mean value filter would instead produce the mean value of $\boldsymbol{p}_0$ and $\boldsymbol{p}_1$. The oscillatory noise is then empirically considered to be small enough to decrease the threshold σ to half of the initial noise size.

The measure used in the included papers is defined as follows. The x and y coordinates are filtered by a rectangular mean value filter of length 4, and then the above mentioned measure is used with the threshold $\sigma = 3$.

6.3.2 Swim bouts

The zebrafish larva moves in discrete bursts of movement – these are denoted *swim bouts*. Some of the features that are extracted from the swim bouts are outlined in the following sections. See also paper V and paper VI.

A swim bout is defined as a interval in time with a momentarily high speed and a net movement. That is, time t_k corresponds to a swim bout if both

$$\frac{\|\boldsymbol{x}_{k+n_0} - \boldsymbol{x}_k\|}{n_0} > v_{\text{thr}}, \quad \text{and} \quad \|\boldsymbol{x}_{k+n_1} - \boldsymbol{x}_k\| > d_{\text{thr}}, \tag{115}$$

where v_{thr} and d_{thr} are speed and distance thresholds, respectively, and n_0 and n_1 are time differences where $n_1 > n_0$. The second inequality can be reformulated as a speed-inequality by dividing by n_1. Furthermore, if the speed thresholds are equal, then

$$\frac{\|\boldsymbol{x}_{k+n} - \boldsymbol{x}_k\|}{n} > v_{\text{thr}}, \quad \forall n \in \mathcal{I}, \tag{116}$$

where $\mathcal{I} = \{n_0, n_1\}$ is the set of time differences. Note that $\mathcal{I}$ is not limited to contain only two values – any number of time differences can be used. The included papers (paper V and VI) use the empirically defined set $\mathcal{I} = \{12, 24, 48\}$, which provides qualitatively good results.

6.3.3 Angles

How and when the zebrafish larva turns is considered important for the analysis that is to follow. Therefore, the turning for each swim bout is measured. The starting point of a swim bout is not clearly defined and sensitive to errors in the pose estimation and, thereby, also the starting angle is not clearly defined and sensitive to noise. For that purpose, the maximal angular *change* is instead measured. This is achieved by, for each swim bout, finding the maximal and minimal cumulative angles that are attained. The angular change is then defined as the first encountered angle of the two extreme angles minus the second. Note that this is typically a value between -180° and 180°.

6.3.4 Clusters

A method that is different from manually defined and traditional kinematic measures of speed, distance, acceleration, angular change, etc. is to define a metric

between swim bouts, and then group the data according to distance. What each cluster describes depend on the definition of the metric and the clustering method.

For that purpose, papers V and VI propose a non-parametric method for grouping similar swim bouts. This method for classifying swim bouts is described in detail in paper V.

6.3.5 Sequencing

A sequence of classified swim bouts can be modeled as a (higher order) Markov chain where the states are the swim bout classes. Thereby, the transitional probabilities of the model can be estimated. In paper V and paper VI, the acquired clusters correspond to actual movements such as left turn, short forwards swimming. Therefore, the transitional probabilities describe actual sequencing of motor behavior. This is described in more detail in paper VI, and the subsequent behavioral analysis is also presented in paper VI.

7 Analysis of extracted feature data

How extracted feature data can be analyzed is proposed primarily in papers II, IV, VI and IX, but also briefly outlined in papers I, III and V. While the computer vision and image analysis methods (e.g. animal pose estimation or camera calibration) can be described in general, the analysis component is implemented specifically for each study to answer particular medical or biological questions. This part is very dependent of the size, individual variability and characteristics of the dataset. An example of how variable the data can be is provided by Fig. 25. Here, the traveled distance per swim bout is presented as histograms for four individual zebrafish larvae in a control experiment. Note the large variability in the number of swim bouts and the small variations in the shapes of the distributions.

The different projects are subject to their own characteristic datasets. Therefore, the analyses of extracted features are not further presented here – the reader is directed to the papers instead. Descriptions of the methods for analyzing rats are outlined in paper II, marmosets in paper IV, zebrafish in paper VI and *Daphnia* in paper IX.

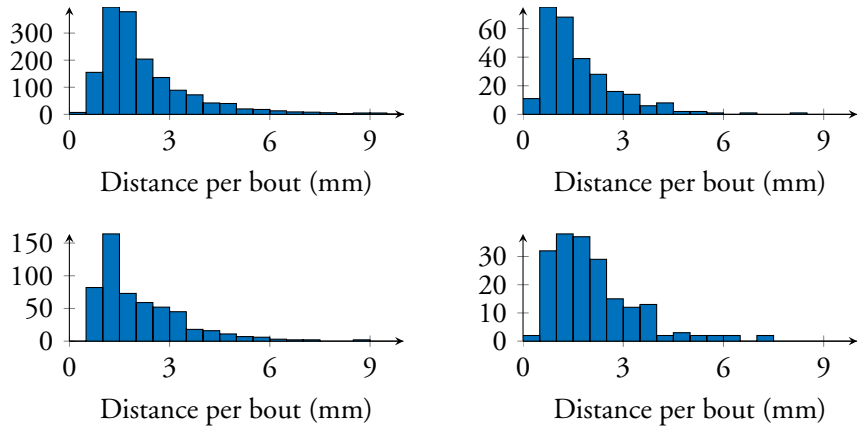


Figure 25: Distance traveled per swim bout for four individual zebrafish larvae in a control experiment. Note the variation in activity of the larvae – the most active executes more than 8 times as many swim bouts as the least active.

8 Modeling of motor program initiation with the Ornstein-Uhlenbeck process

The Ornstein-Uhlenbeck process is a stochastic process that can be defined as the solution to the following stochastic equation

$$dx_t = \theta(\mu - x_t)dt + \sigma dW_t, \tag{117}$$

where θ, μ and σ are constant parameters of the process and W_t is a Wiener process. That is, in each time step t, the process x_t takes a step $\theta(\mu - x_t)dt$ towards the value μ and a random step σdW_t. One solution is given by a random walk process that starts at zero and at each time step is updated by a random walk step and a drive step.

The Ornstein-Uhlenbeck *distribution* is defined by the expected time for an Ornstein-Uhlenbeck process, $x^{(t)}$, that starts at zero with drive parameter c_{drift} and step size c_{diff} to reach the threshold $c_{\text{threshold}}$. The associated probability density function can be expressed as

$$p(t) = P(x^{(t)} \geq c_{\text{threshold}}). \tag{118}$$

In general, there exists no closed-form solution to the equation. Instead, the distribution is usually estimated by simulating the Ornstein-Uhlenbeck process

sufficiently many times and then estimating the expected arrival times numerically. In the following is proposed a method for computing the distribution directly.

Consider the stochastic process described by

$$\begin{cases} x^{(t)} = x^{(t-1)} + dW_t + c_{\text{drift}}, \\ x^{(0)} = 0, \end{cases} \tag{119}$$

where dW_t is a random walk process with discrete time steps of length c_{diff}. At time $t = 1$, the expected position of $x^{(t)}$ is given by

$$\begin{aligned} P(x^{(1)} = c_{\text{drift}} + c_{\text{diff}}) &= 1/2, \\ P(x^{(1)} = c_{\text{drift}} - c_{\text{diff}}) &= 1/2, \end{aligned} \tag{120}$$

for $t = 2$ by

$$\begin{aligned} P(x^{(2)} = 2c_{\text{drift}} + 2c_{\text{diff}}) &= 1/4, \\ P(x^{(2)} = 2c_{\text{drift}} + 0) &= 1/2, \\ P(x^{(2)} = 2c_{\text{drift}} - 2c_{\text{diff}}) &= 1/4, \end{aligned} \tag{121}$$

and for $t = 3$ by

$$\begin{aligned} P(x^{(3)} = 3c_{\text{drift}} + 3c_{\text{diff}}) &= 1/8, \\ P(x^{(3)} = 3c_{\text{drift}} + c_{\text{diff}}) &= 3/8, \\ P(x^{(3)} = 3c_{\text{drift}} - c_{\text{diff}}) &= 3/8, \\ P(x^{(3)} = 3c_{\text{drift}} - 3c_{\text{diff}}) &= 1/8. \end{aligned} \tag{122}$$

Note that in each time step, the probabilities are the convolution of the previous probabilities and $(-1, 1)$. For example, the probabilities for $t = 1$, $\boldsymbol{p}^{(1)} = (1/2, 1/2)$, are updated to $\boldsymbol{p}^{(2)} = (1/4, 1/2, 1/4) = \boldsymbol{p}^{(1)} * (1, 1)$. Also note that in each time step, the positions of non-zero probability are updated as follows: the drive step is added to all elements, a positive random walk step is added to the first element and a negative step is added to the remaining elements.

Thus is provided the first component of a method for simulating the expansion

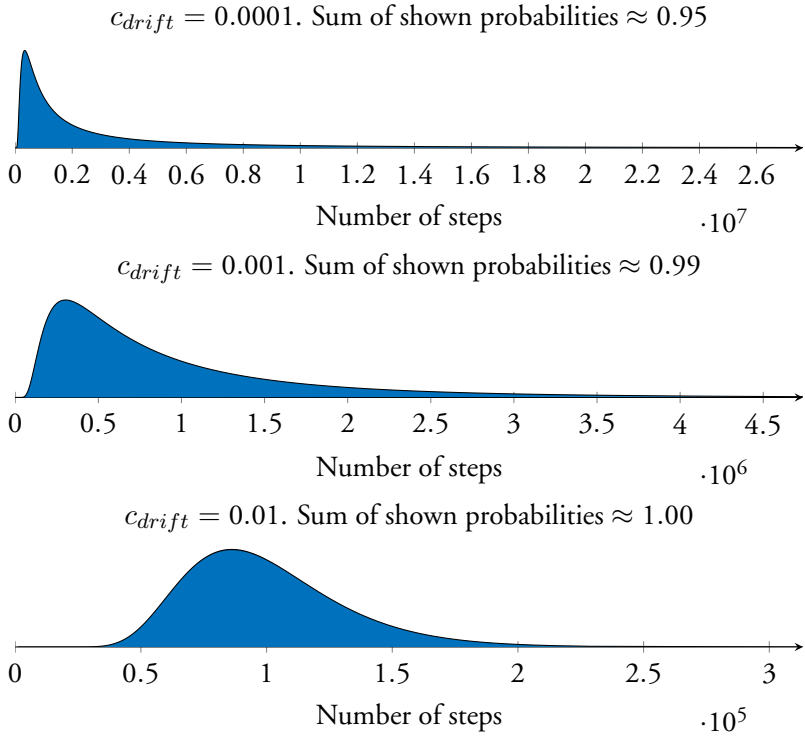


Figure 26: Examples of simulated Ornstein-Uhlenbeck distributions with varying drive parameters c_{drift}. Note that the tails are particularly heavy for low drive values and that the distribution is more symmetric for high drive values.

of the probability over time as

$$\begin{cases} \boldsymbol{p}^{(0)} = (1), \\ \boldsymbol{x}^{(0)} = (0), \\ \boldsymbol{p}^{(k+1)} = \boldsymbol{p}^{(k)} * (1,1)/2, \\ \boldsymbol{x}^{(k+1)} = \Big(x_1^{(k)} + c_{\text{diff}} + c_{\text{drift}}, x_1^{(k)} - c_{\text{diff}} + c_{\text{drift}}, \\ \qquad\qquad x_2^{(k)} - c_{\text{diff}} + c_{\text{drift}}, x_3^{(k)} - c_{\text{diff}} + c_{\text{drift}}, \dots \Big) . \end{cases} \tag{123}$$

Note that the vectors of probabilities $p^{(k)}$ and positions $x^{(k)}$ increase in size each iteration.

Input: Drift parameter c_{drift}, number of iterations N and random walk step length c_{diff}.
Output: Vector of probabilities $\boldsymbol{P}$.

$\boldsymbol{P} = \mathbf{0}_{N\times 1}$
$\boldsymbol{p}^{(1)} = (1)$
$\boldsymbol{x}^{(1)} = (0)$
for $k = 2, \ldots, N$ **do**
    $n \leftarrow \text{length}(\boldsymbol{p}^{(k)})$
    if $x_1^{(k)} \geq 1$ **then**
        $P_k \leftarrow p_1^{(k)}$
        $\boldsymbol{p}^{(k)} \leftarrow \left(p_2^{(k)}, \ldots, p_n^{(k)}\right)$
        $\boldsymbol{x}^{(k)} \leftarrow \left(x_2^{(k)}, \ldots, x_n^{(k)}\right)$
    end if
    $\boldsymbol{p}^{(k+1)} \leftarrow \boldsymbol{p}^{(k)} * (1,1)/2$
    $\boldsymbol{x}^{(k+1)} \leftarrow \Big(x_1^{(k)} + c_{\text{drift}} + c_{\text{diff}}, x_1^{(k)} + c_{\text{drift}} - c_{\text{diff}},$
        $x_2^{(k)} + c_{\text{drift}} - c_{\text{diff}}, \ldots, x_n^{(k)} + c_{\text{drift}} - c_{\text{diff}}\Big)$
end for

Algorithm 2: Simulation of the Ornstein-Uhlenbeck distribution. Note that the threshold can be set $c_{\text{threshold}} = 1$ by rescaling c_{drift} and c_{diff}.

The second component describes what happens when the threshold is reached. Assume that at time $t = n$, the first element of the position vector is larger than or equal to the threshold, i.e. $x_1^{(n)} \geq c_{\text{threshold}}$. Then the probability of reaching the threshold in n iterations is given by $p_1^{(n)}$. Since the sought distribution describes the probabilities of reaching the threshold in any given time, the first element of both $\boldsymbol{x}^{(n)}$ and $\boldsymbol{p}^{(n)}$ are removed after saving the value of $p_1^{(n)}$. Then the iteration scheme continues as before.

Thus has been outlined a method for computing the Ornstein-Uhlenbeck distribution. This is summarized in Algorithm 2. Distributions for a few selected drift parameter values are shown in Fig. 26.

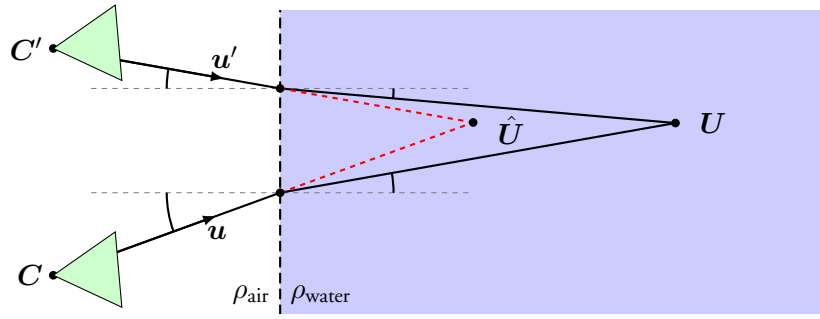


Figure 27: Triangulation in water. The ray paths that correspond to the observations $\boldsymbol{u}$ and $\boldsymbol{u}'$ of a scene point $\boldsymbol{U}$ in two cameras are plotted. Note that the rays parametrized as $\boldsymbol{C} + s\boldsymbol{u}$ and $\boldsymbol{C}' + t\boldsymbol{u}'$ meet at $\hat{\boldsymbol{U}}$, which is not the position of the observed scene point.

9 Calibration and positioning with refractive and reflective planes

The problem of estimating the position of objects behind refractive and reflective interfaces from observed image points is considered in paper VIII. That is, the 3D positions of *Daphnia* in an aquarium are sought from recorded videos of a biological experiment (see paper IX).

To estimate the 3D position of a point in air corresponding to image point observations in at least two cameras is simple if the cameras are intrinsically and extrinsically calibrated. For each image point, there is a ray into the scene that can be computed. The observed 3D point corresponding to a set of image points is found at the intersection of all rays from those image points. For the intended application, however, the 3D points are in water and the cameras in air. The light ray from a scene point pass through the flat glass of the aquarium before arriving at the camera. A ray of light from a scene point in water is refracted (i.e. change direction) as it passes through one medium to another. The ray path from a scene point to a camera is therefore first refracted at the water-glass interface, and then again at the glass-air interface. As a result, the scene point is typically not where in-air triangulation would estimate it to be (see Fig. 27). How the angle of a ray changes as it passes through the interface from one medium to another is described by Snell's law, which is visualized in Fig. 28. Note that in the proposed

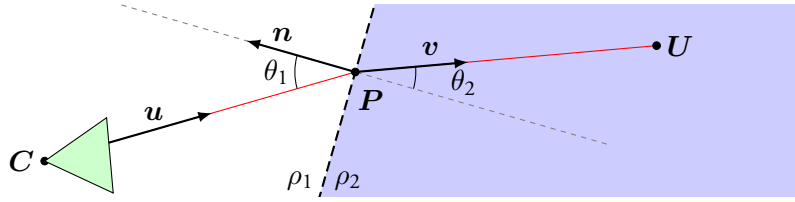


Figure 28: Snell's law. A ray originating from the camera center C with direction $\boldsymbol{u}$ undergoes a change of direction according to $\rho_1 \sin\theta_1 = \rho_2 \sin\theta_2$. This causes the usually linear equations for projections, for example, to become nonlinear and much harder to solve.

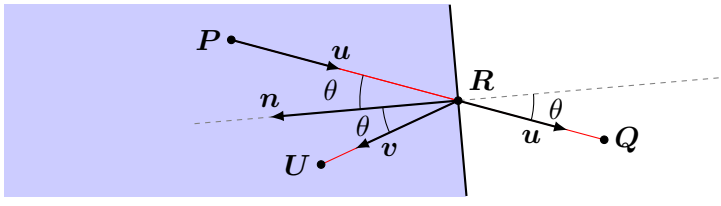


Figure 29: Reflections. A ray with direction $\boldsymbol{u}$ originating from a point $\boldsymbol{P}$ intersects a reflective plane $(\boldsymbol{n}, d)$ at a point $\boldsymbol{R}$, where the ray changes direction to $\boldsymbol{v}$ and subsequently intersects a point $\boldsymbol{U}$. Note that the ray segment from $\boldsymbol{R}$ to $\boldsymbol{U}$ is the mirror image of the virtual ray segment from $\boldsymbol{R}$ to $\boldsymbol{Q}$.

methods, it is assumed that the glass is thin enough to make a single refraction (water-air) a good enough approximation.

In addition to refraction, there is the possibility that a ray is reflected in a wall or the water surface before it arrives at the camera. Reflections are modeled using *virtual points* – this allows for reflecting points instead of directions. Thereby, the point where a ray is reflected is not used. An example is visualized in Fig. 29. For more details on the modeling of reflections, see paper VIII.

Thus far has been described the angular changes that the light ray of an observed *Daphnia* undergoes before it meets the camera. These are formulated as equations that depend on refractive and reflective plane parameters, refractive indices, scene points, points of refraction, virtual points, image points and intrinsic and extrinsic camera parameters. Additional equations are formulated to constrain projections to the measured image points and the plane normals to have

unit length. All of these equations are reformulated as error functions and implemented in a bundle adjustment framework (see section 3.3). This framework can be used to, for example, calibrate the refractive and reflective plane parameters, estimate the 3D position corresponding to a set of observed image points and project scene points in the cameras.

The details of each of these steps are presented in paper VIII.

9.1 Additional results

In addition to the results presented in paper VIII, an evaluation was performed to investigate how triangulation errors are related to the number of reflective planes. Therefore, a practically relevant evaluation was done by generating scene points in the model of the actual set-up used for the *Daphnia* experiments. That is, data such as the observed data in the *Daphnia* experiment was synthetically generated. The scene points were projected, using the previously outlined methods, in different reflective plane configurations. Since the real set-up has three reflective surfaces (two walls and the surface of the water), there are three possible planes that a ray can be reflected exactly one time. For the case of more than two reflections, only subsets of the possible configurations were considered. The results of triangulation using these different configuration of reflection are presented as the distance to ground truth in Fig. 30. From the figure can be concluded that the triangulation of a scene point is less sensitive to noise if it is (in addition to a direct view) also observed through reflections.

10 The Misty Three Point algorithm

The Misty Three Point algorithm is a method that estimates the relative motion of a camera from three pairs of image coordinates that correspond to the same scene point, and the observed colors at those coordinates.

10.1 Related work

To estimate the relative motion of two cameras is a well-studied topic that has been researched for more than a century [14, 41]. In 1855, Chasles proposed solutions to the relative pose problem. In 1913, Kruppa proved that the five-point problem has at most 11 solutions. A more thorough survey of the history of computer vision is given by Sturm [82].

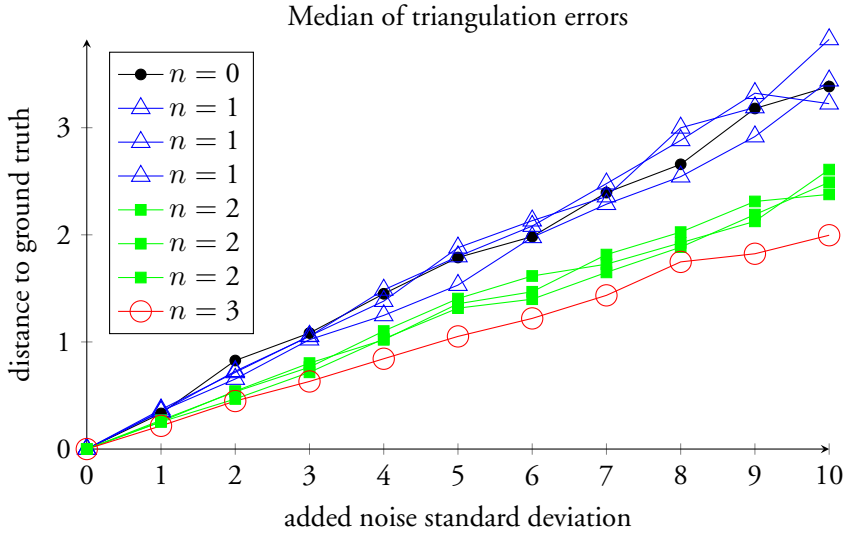


Figure 30: Median of triangulation error for different reflective configurations as a function of added image point noise variance x^2. $n = 0$ means that triangulation is performed using only direct views of the point. $n > 1$ means that both direct views and n additional observations through reflection, for each camera, are used for triangulation.

However, the actual problem to solve depends on a number of assumptions and what parameters are known. Nistér proposes an efficient solution to the minimal case (five pairs of corresponding image points) for a calibrated pinhole camera [55]. Stewénius et al. first show that at least six pairs of corresponding image points are needed to estimate the relative poses of two calibrated generalized cameras and then solves the minimal case [81].

A less studied topic is that of estimating structure and motion from underwater images. Jordt and Koch propose a method for estimating relative pose where refractions caused by a flat port housing are handled using a virtual camera [38]. Here, a non-minimal relative pose solution is proposed that improves upon the results that are obtained by a simple pinhole camera model. The problem of estimating the absolute pose of a camera under flat refractive interfaces is solved minimally by Haner and Åström [30].

There are also proposed methods for restoring images that have been degraded by attenuation in water. For example, Schechner and Karpel propose methods

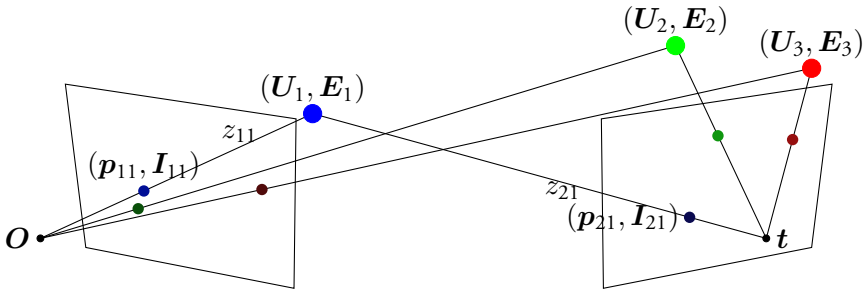


Figure 31: The relative pose problem solved in paper X. Two cameras centered at the origin and t, respectively, observe three points of unknown position and unknown color. Note that we are not only using the direction $\boldsymbol{u}_{ik}$ from each observed point, but also the observed color. The differences in depth $\Delta z_{ik} = z_{jk} - z_{ik}$ are also crucial parts of this method. In the left camera, with larger distance to the objects, the colors look very similar. In the right camera, however, the observed colors are more similar to the actual color of the object. This distance dependent degradation in color is what enables the estimation of depth differences.

for recovering the colors and estimating depth maps in single underwater images using polarized light [72, 73]. This problem is also considered by Jordt [37]. The related topic of restoring the colors in hazy images has gained more attention that underwater applications, but often use similar physical models [32, 74, 83]. For example, Bahat and Irani propose a method for dehazing foggy images using recurring image patches [2].

In paper X, we consider the case where pairs of corresponding image points in two generalized cameras are known and the difference in distance to each of the corresponding scene points are known. We show that at least three pairs of image points are required and solve the minimal case. Furthermore, we propose a method to estimate the differences in depth from underwater images. As a result, we propose a method for estimating the relative poses of two underwater cameras using three points and their colors.

10.2 The algorithm

Assume that three scene points $\boldsymbol{U}_k$ with irradiance $\boldsymbol{E}_k$ are observed in two cameras as the points $\boldsymbol{x}_{ik}$ with colors $\boldsymbol{I}_{ik} = (I_{ik}^{\text{red}}, I_{ik}^{\text{green}}, I_{ik}^{\text{blue}})^\top$, where k denote the point index and i the camera index. The observed colors of the points are

modeled by a simplified version of the Jaffe-McGlamery equation,

$$I_{ik}^{\lambda} = \alpha_{\lambda} \left(E_k^{\lambda} e^{-\eta_{\lambda} z_{ik}} + B_{\infty}^{\lambda} (1 - e^{-\eta_{\lambda} z_{ik}}) \right) + \beta_{\lambda}, \tag{124}$$

where $\lambda \in \{\text{red}, \text{green}, \text{blue}\}$ is the color channel, α_{λ} and β_{λ} are color correction coefficients [37], η_{λ} is the attenuation coefficient of the water, z_{ik} is the distance from the outer refraction plane of camera i to point k and B_{∞}^{λ} is the "veiling light".

It is noted that the Jaffe-McGlamery equation can be solved for the term $\alpha_{\lambda} \left(\boldsymbol{E}_k - B_{\infty}^{\lambda} \right)$, which is equal in both observations of point k. This leads to the equation

$$\frac{I_{ik}^{\lambda} - \gamma_{\lambda}}{I_{jk}^{\lambda} - \gamma_{\lambda}} = e^{\eta_{\lambda} \Delta z_{ijk}}, \tag{125}$$

where $\Delta z_{ijk} = z_{jk} - z_{ik}$ is the difference in distance between the two observations and $\gamma_{\lambda} = \alpha_{\lambda} B_{\infty}^{\lambda} + \beta_{\lambda}$ is the observation of the veiling light. Thus, it has been shown that there is a quantifiable pseudo-depth information in underwater stereo images.

A method (the Three Color Depth Difference algorithm) is proposed to solve Eq. (125) for the unknown difference in depth, and to estimate the underwater imaging parameters γ_{λ} and η_{λ}. This is done by defining an error function,

$$r_{\lambda}(\gamma, \eta, \Delta z_{ijk}) = I_{ik}^{\lambda} - \gamma_{\lambda} - (I_{jk}^{\lambda} - \gamma_{\lambda}) e^{\eta_{\lambda} \Delta z_{ijk}}, \tag{126}$$

and performing a nonlinear optimization routine using Gauss-Newtons algorithm.

A method (the Three Point Delta algorithm) is proposed that takes image points $\boldsymbol{x}_{ik}$ and differences in depth $\Delta z_{ijk} = z_{jk} - z_{ik}$ as input, and gives an estimate of the relative motion between camera i and j as output. The method is derived for a generalized camera that is assumed to be calibrated. That is, for each image point $\boldsymbol{x}_{ik}$, a 3D-point $\boldsymbol{p}_{ik}$ and 3D-direction $\boldsymbol{u}_{ik}$ can be computed. In the case of underwater cameras – i.e. where the difference in depth is the difference in distance between the two camera refractive planes and the scene point – $\boldsymbol{p}_{ik}$ is assumed to be a point on the outer refractive plane of camera i.

For each pair of observed image points $(\boldsymbol{x}_{ik}, \boldsymbol{x}_{jk})$ that correspond to the same scene point $\boldsymbol{U}_k$, the rays into the scene can be parametrized by the distances z_{ik}

and z_{jk} to the scene point,

$$\begin{aligned} \boldsymbol{U}_k &= \boldsymbol{p}_{ik} + z_{ik}\boldsymbol{u}_{ik}, \qquad z_{ik} \in \mathbb{R}, \\ \boldsymbol{U}'_k &= \boldsymbol{p}_{jk} + z_{jk}\boldsymbol{u}_{jk}, \qquad z_{jk} \in \mathbb{R}. \end{aligned} \tag{127}$$

Note that these parametrizations are given in the local coordinate systems of each camera. Since the poses of the cameras is unknown, there is an unknown rigid transformation between $\boldsymbol{U}_k$ and $\boldsymbol{U}'_k$. Therefore, the Gramians for the two sets of points $\mathcal{U} = \{\boldsymbol{U}_1, \boldsymbol{U}_2, \boldsymbol{U}_3\}$ and $\mathcal{U}' = \{\boldsymbol{U}'_1, \boldsymbol{U}'_2, \boldsymbol{U}'_3\}$ are equal [94]. The Gramian for $\mathcal{U}$ is defined as $V^\top V$, where

$$V = \begin{bmatrix} \boldsymbol{U}_2 - \boldsymbol{U}_1, & \boldsymbol{U}_3 - \boldsymbol{U}_1 \end{bmatrix}. \tag{128}$$

Inserting the expressions for the $\boldsymbol{U}_k$'s and $\boldsymbol{U}'_k$'s gives the Gramians

$$\begin{aligned} V &= \begin{bmatrix} z_{i2}\boldsymbol{u}_{i2} - z_{i1}\boldsymbol{u}_{i1} + p_{i2} - p_{i1} \\ z_{i3}\boldsymbol{u}_{i3} - z_{i1}\boldsymbol{u}_{i1} + p_{i3} - p_{i1} \end{bmatrix}^\top, \\ V' &= \begin{bmatrix} (z_{i2} + \Delta z_{ij2})\boldsymbol{u}_{j2} - (z_{i1} + \Delta z_{ij1})\boldsymbol{u}_{j1} + p_{j2} - p_{j1} \\ (z_{i3} + \Delta z_{ij3})\boldsymbol{u}_{j3} - (z_{i1} + \Delta z_{ij1})\boldsymbol{u}_{j1} + p_{j3} - p_{j1} \end{bmatrix}^\top. \end{aligned} \tag{129}$$

Thus the constraint that the Gramians are equal amounts to

$$V^\top V - (V')^\top (V') = 0, \tag{130}$$

which provides three equations that are quadratic in the three unknowns z_{ik},

$$\begin{cases} A_{11}xy & & +A_{14}x + A_{15}y & + A_{17} = 0, \\ A_{21}xy + & A_{22}xz + A_{23}yz & +A_{24}x + A_{25}y & +A_{26}z + A_{27} = 0, \\ & A_{32}xz & +A_{34}x & +A_{36}z + A_{37} = 0, \end{cases} \tag{131}$$

where $x = z_{i1}$, $y = z_{i2}$, and $z = z_{i3}$. How this system of three polynomial equations in three unknown can be solved is outlined in section 3.4.

Given a solution (z_{i1}, z_{i2}, z_{i3}) to Eq. (131), the scene points $\boldsymbol{U}_k$ and $\boldsymbol{U}'_k$ can be computed. What remains is then to estimate the rigid transform $(R, \boldsymbol{t})$ from $\boldsymbol{U}_k$ to $\boldsymbol{U}'_k = R\boldsymbol{U}_k + \boldsymbol{t}$. The rigid transform is the sought relative motion between the first and second camera.

Lastly, a method (the Misty Three Point algorithm) is proposed that takes image points $\boldsymbol{x}_{ik}$ and their colors $\boldsymbol{I}_{ik}$ from two cameras as input, and provides an estimate on the relative poses of the cameras. This method is a combination of the Three Color Depth Difference algorithm and the Three Point Delta algorithm.

Applications

11 Skilled reaching in rats

The main neurophysiological rationale for studying the *skilled reaching* behavior in rats is that it allows for studying the central nervous system as it initiates and controls fine motor behavior. In this experimental paradigm, the animals learn to pick up food rewards by their front paws as shown in Fig. 32.

Some of the earliest neurophysiological studies on rat reaching was performed in the 1930's [64, 65] for the purpose of investigating handedness. Skilled reaching in rats has been studied extensively by I.Q. Whishaw since the 1980's [1, 51, 52, 69, 90, 91]. Two important observations in these papers is that skilled reaching appears be a sensible tool for e.g. detecting early stages of Parkinson's disease [51, 52] and that skilled reaching in rats is similar to skilled reaching in humans [69, 91]. Therefore it is of great scientific and practical interest to further investigate both the behavioral and neuronal dynamics of skilled reaching in rats.

The skilled reaching tests were conducted as follows. Each reaching trial starts at the back side of the box (the rightmost side of Fig. 33b). Then the rat climbs over the obstacle in the center of the figure, and approaches the left side of the image where it can reach for food. After it has made a reaching attempt, it has to return to the back side of the box before new food is placed on the shelf. This part of the procedure ensures that the time interval for each reaching trial is separated from the others. Thus, the neural activity during a reaching movement can be compared to the baseline activity before the attempt.

The dataset for each recording consists of video data and neural data. Four cameras are directed at the reaching scene (see Fig. 33a and Fig. 32) and a fifth camera (not used in the included papers) views the rectangular box from the side (see Fig. 33b). The neural data were provided by micro-electrodes with 64 chan-

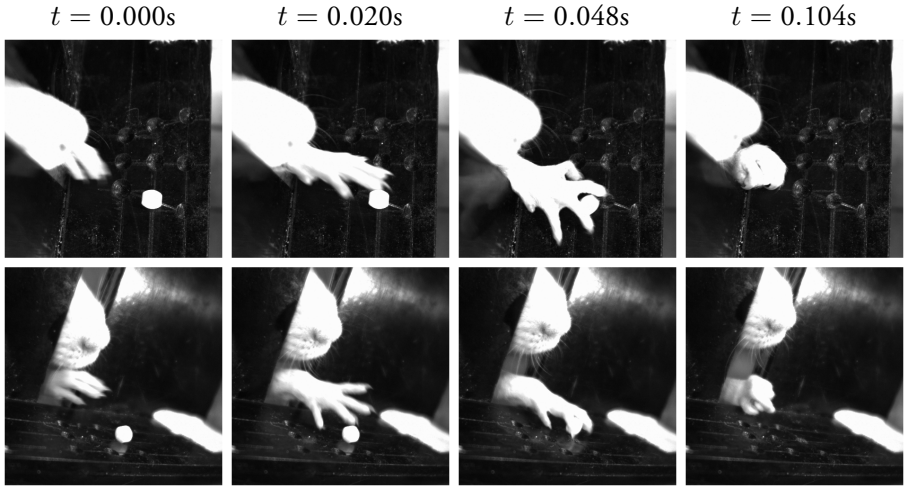
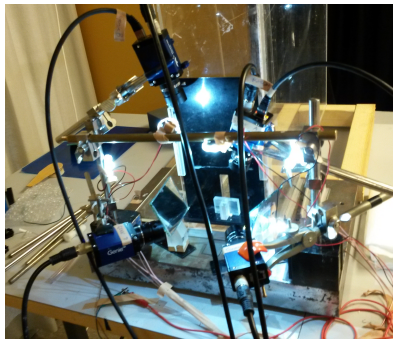


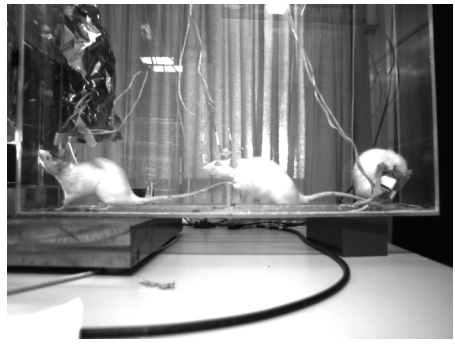
Figure 32: The reaching movement of a rat. Images from two cameras are shown from four points in time during a reaching attempt.

nels, that were implanted into the primary motor cortex and striatum in both of the hemispheres of the animals. Each channel is sampled at 32 kHz, and each sample is represented by a measurement of the voltage. Both the activity of single neurons (referred to as the *single unit activity*) and the low-pass filtered signals of each channel (referred to as the *local field potential*) are extracted from these measurements. The single unit activity is represented by the sequence of times when *action potentials* are generated, and the local field potentials are represented by the sequence of measured voltages. These data are subsequently analyzed together with the behavioral tracking data.

The behavioral data is extracted as follows. First, the scene is calibrated using DLT (see section 3.2.1) and the background is modeled as described in section 4.1. The animal is geometrically modeled and its pose is estimated in each frame as outlined in section 5.2. Methods for generating and projecting quadric surfaces are introduced in sections 4.4 and 4.5. Thirdly, features were extracted from the data as presented in section 6.1. Lastly, the behavioral data and the correlation between the neural data and the extracted features is analyzed as proposed in paper II.



(a) Front view



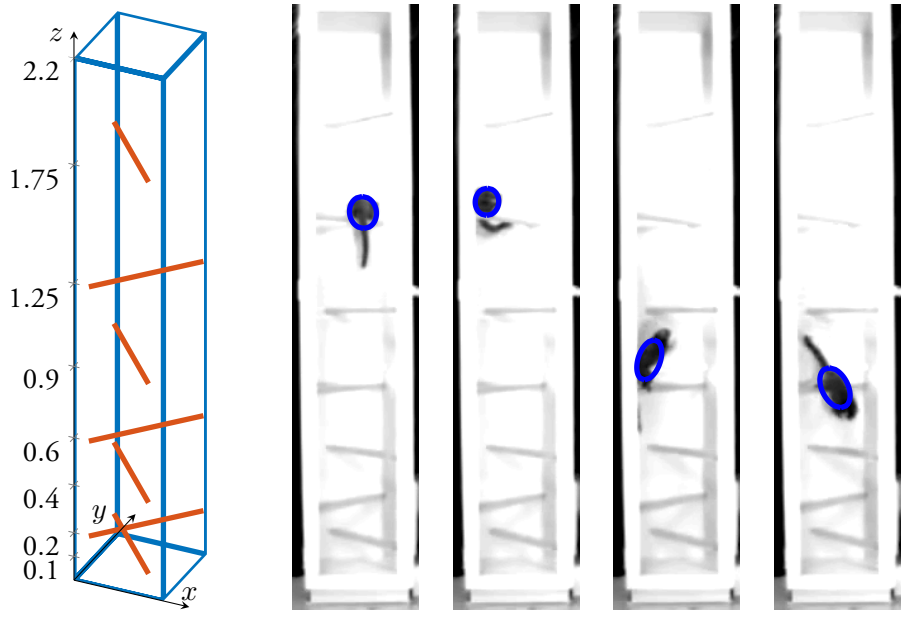
(b) Side view

Figure 33: (a): an overview of the experimental set-up. Four cameras are recording a scene in which the rat can reach for food rewards (cf. Fig 32). A calibration object – i.e. an object with known dimensions and geometry that is used to calibrate the cameras – is placed in the scene. (b): three merged images from a side-view camera that is not seen in the left figure. In this image, the slit through which the rat can reach for food rewards is on the left hand side. Note that there is an obstacle that the middle rat is climbing over - this obstacle is used to force the rat to do a reaching-like movement but for a different purpose that receiving food rewards.

12 Box and tower experiments in the common marmoset

The common marmoset (*Callithrix jacchus*) is a widely used model species for medical studies. In this thesis is included two papers [61, 70] that studies the long term effects of motor impairment induced by a 6-OHDA model of Parkinson's disease. The proposed computer vision based methods are used together with manual scoring, and shows that the induced Parkinsonism persists for several months. Thus is implied that this non-human primate model of Parkinson's disease is well suited for long-term evaluation of novel therapies for treatment of Parkinson's disease.

To enable the study of the motor symptoms, two experimental set-ups – *box* and *tower* – were constructed. In the box experiment, a marmoset is placed inside a box, as depicted in Fig. 35, where it is free to move around. There are small circular holes on one side of the box and shelves on the outside. These were used in a parallel study where animals were trained to reach for and grasp food rewards – this behavior was not evaluated in the included studies and the shelves were not baited. In the tower experiment, a marmoset is placed inside a vertical cage that



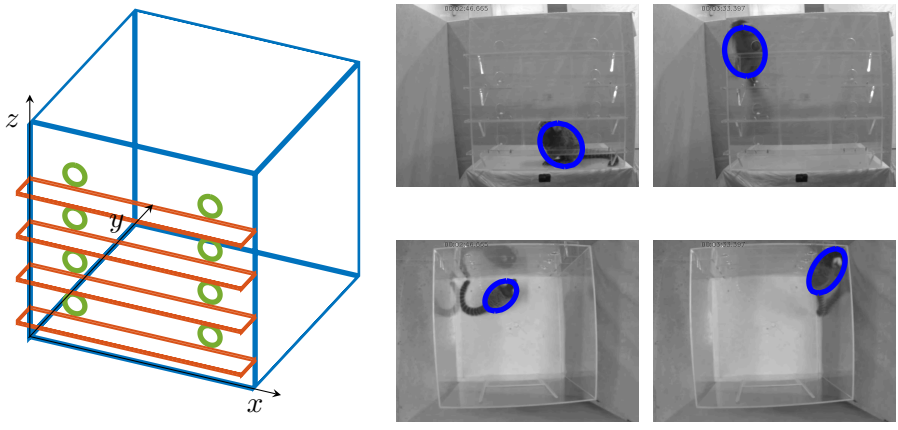
(a) The tower set-up.

(b) Example video data and pose estimation results for four frames.

Figure 34: The tower set-up is plotted in (a) and example frames from the recorded videos are shown in (b). The estimated geometrical model of the animal has been plotted in blue on the images.

has 7 sticks at different heights above the floor, as shown in Fig. 34. The distance between the sticks increases as the distance from the floor increases.

The most relevant theory of image analysis and computer vision, and the specific methods applied for the box and tower experiments are listed in the following. The background is modeled as described in section 4.1. Geometrical modeling and tracking of the marmosets is motivated and outlined in sections 5.3 and 5.4. Methods for generating and projecting quadric surfaces are introduced in sections 4.4, 4.5 and fitting quadrics to images in section 4.6.1. Feature extraction is performed as proposed in section 6.2. Analysis of the extracted feature data is presented in paper IV.



(a) The box set-up.

(b) Example data and projected pose estimation results for two frames (left and right columns).

Figure 35: The box set-up is plotted in (a) and examples of recorded video data are shown in (b). Note that the plotted pose estimation results are the projections of the estimated 3D-pose. Also note that there are holes in one of the walls and shelves on the outside of the box – these are not used for the presented study.

13 Tracking the rhopalia of box jellyfish

The box jellyfish is a simple organism that does not have a brain (it only has a simple nerve net). What is remarkable, however, is that it has eyes. An animal that has both of these properties is very useful for studies on vision and systems of vision. Due to the absence of a brain, it can be assumed that the neural decision making processes that initiates movement are relatively simple, as compared to e.g. rats or marmosets.

One of the goals of this project is to study the correlation between visual stimuli and motor output. For that purpose, an experimental set-up has been built where the jellyfish are fastened at the top of the bell. The animals are video recorded while they attempt to move (see Fig. 36). In addition, there are lighting elements that can provide visual stimuli (not used in this study).

To enable the study of jellyfish behavior, the attempted movements of the animals need to be quantified. In paper VII is proposed a method for tracking the rhopalia of the box jellyfish in this particular set-up. The proposed method is efficient and could potentially be implemented for real-time applications.

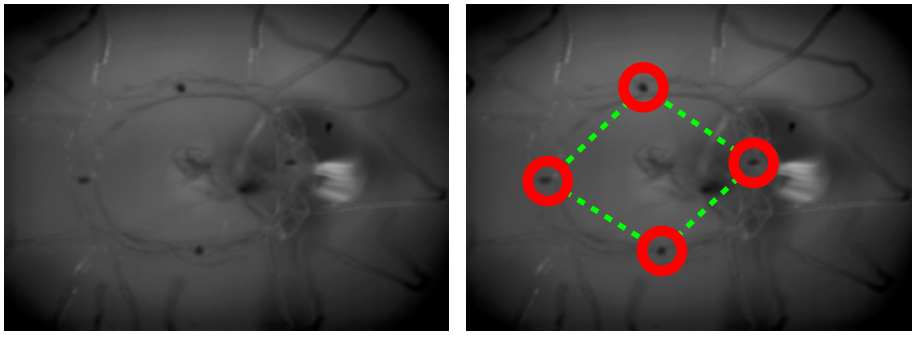


Figure 36: An image from a video of box jellyfish. Model fitting results have been superimposed as red circles (rhopalia) and dashed green lines (body).

A possible use of a real-time tracking application is to create a virtual reality system for the jellyfish – i.e. a system where the attempted movements of the jellyfish is forwarded to the light elements that provide the visual stimuli. For example, the lights could change according to the estimated direction of the attempted swim movements. This could potentially let biological scientists create virtual mazes, for example, where the jellyfish attempt to navigate.

14 Drug induced behavior in zebrafish

The zebrafish (*Danio rerio*) is a commonly used animal in the field of medicine, but up until recently the commercially available automatic tracking software have tracked only one position per fish, consequently greatly limiting the behavioral analysis. In order to allow for more advanced analysis of the behavior of zebrafish larvae, more advanced tracking data are required. Therefore, papers V and VI propose a system for tracking the body shape of zebrafish larvae, and analyzing the results, in a high-throughput application. This system allows for experiments on large numbers of individuals and, at the same time, it provides a high level of detail.

The experimental set-up used for the zebrafish experiments consists of two microtiter plates placed in a water bath (to maintain 28°C temperature). A camera records the scene from above in 300 frames per second at a resolution of 640×480 pixels (see Fig. 37). Lighting of the scene is provided by LED strips positioned below the plates.

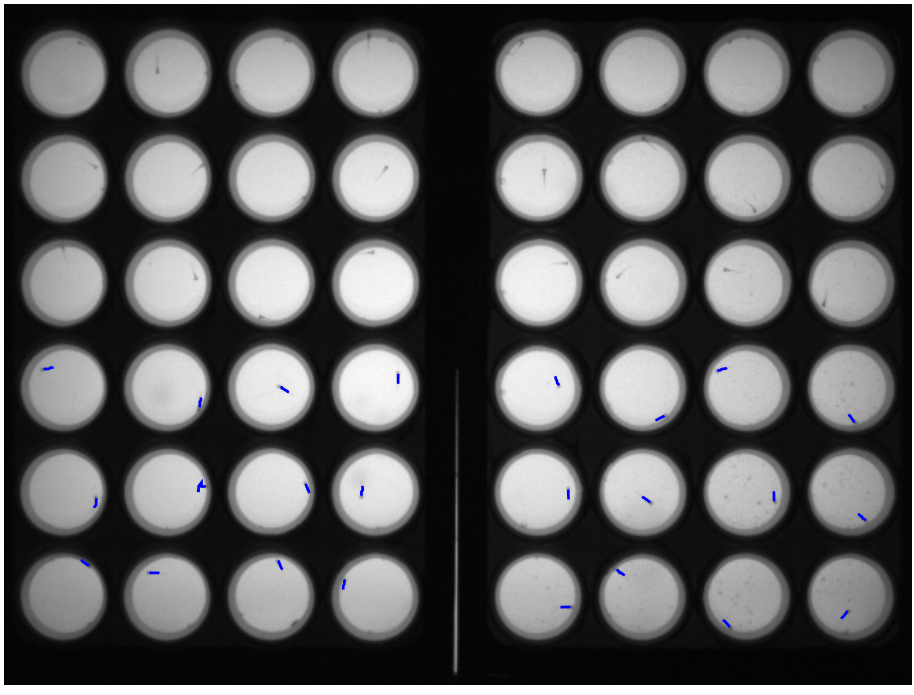


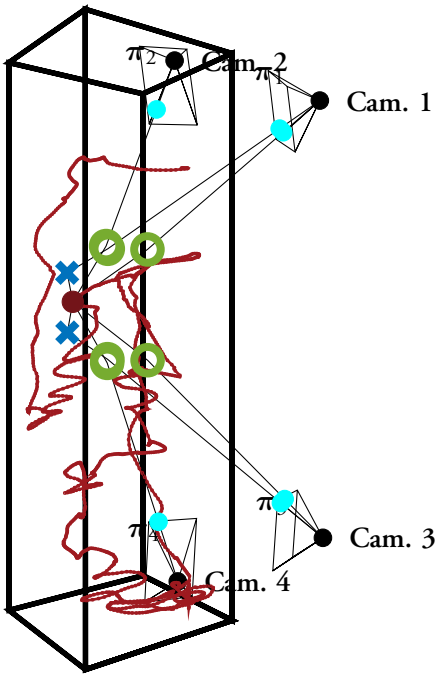
Figure 37: A frame from the zebrafish videos. Pose estimation results for half of the larvae are superimposed in blue.

The methods outlined in section 4.1 were used to create a background image. The geometrical modeling and tracking of the larvae is performed as presented in section 5.5. Kinematic features are extracted by the methods outlined in section 6.3. Lastly, the behavior of the larvae are analyzed as proposed in paper VI.

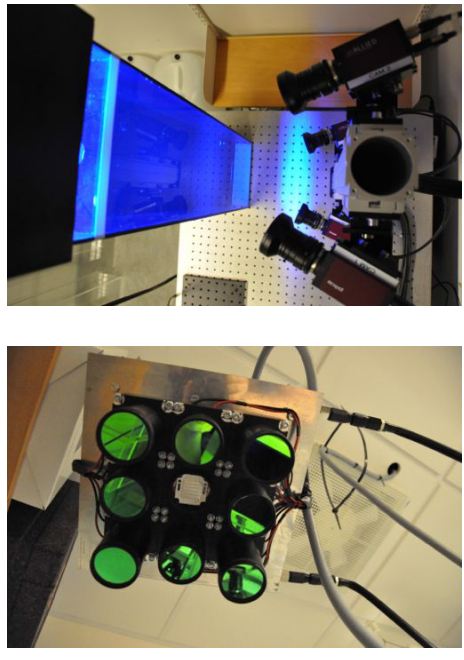
15 UV-light and predator response in *Daphnia*

The zooplankton *Daphnia* is a small aquatic organism that lives in e.g. lakes in large parts of the world. Their response to multiple threats (UV-radiation and predation) is studied in the presented application (paper IX). This behavioral study is enabled by the computer vision methods proposed in paper VIII.

The experimental set-up consists of an aquarium, an array of UV-light LEDs,



(a) The result of calibration of the four cameras and the triangulated 3D-positions of one *Daphnia*.



(b) Photos of the aquarium and cameras (top) and the LED array used to excite the fluorescent quantum dots and simulate UVR threat (bottom).

Figure 38: The experimental *Daphnia* set-up used for the experiments on *Daphnia*. (a) shows the results of calibration, results of positioning and the ray-paths from one point to the cameras. Here, the cyan dots are image points, the green circles are points of refraction, the blue crosses are points of reflection, and the large red dot is the estimated 3D position of a *Daphnia*.

excitation light LEDs for the quantum dots, and four cameras (see Figure 38).

The methods proposed in section 9 and 3.3 are used to calibrate the scene and to estimate the positions of the *Daphnia*. Extraction of feature data and analysis of feature data is conducted as proposed in paper IX.

Overview of the papers

In this chapter is presented an overview of the papers included in this thesis. Note that the order of papers I and II is reversed in this section.

Paper II - A System for Automated Tracking of Motor Components in Neurophysiological Research

This paper presents a novel method for tracking a rat paw and a novel application of the method for a specific neurophysiological test. For the purpose of obtaining useful behavioral data, the rat paw is modeled by a set of hierarchically connected ellipsoids that represent the palm and the phalanges. In this model, the pose of the paw is described by the position of the wrist, angle of the palm of the paw, and the angles (along the anatomically possible axes of rotation) of each of the phalanges. An iterative and generative method is proposed to estimate the paw poses. Furthermore, methods for analyzing the correlations between fine motor behavior and neural activity are proposed. Lastly, examples of correlations found in the recorded data using the tracked paw poses are presented.

Paper I - Rat Paw Tracking for Detailed Motion Analysis

This paper provides more in-depth descriptions of the computer vision algorithms employed in paper II. Some updates of the algorithms are also presented. The used dataset is identical to that of paper II.

Paper III - Automated Tracking of Motor Behavior as a Means to Assess Severity of Symptoms in the 6-OHDA Marmoset Model of Parkinsons Disease

This paper propose a method for tracking marmosets in two separate experimental testing set-ups - *box* and *tower*. In the box set-up, the marmoset is placed in

a cubic testing chamber and recorded by two cameras. The animal is modeled by an ellipsoid of variable axis lengths, position and orientation (i.e. $3 + 3 + 2$ parameters). In the tower set-up, the animal is recorded in a high and narrow setup using one camera. The marmoset is modeled by an ellipsoid of variable axis lengths, position and orientation. Movements of the marmoset is constrained to the (x, y)-plane since it is observed by a single camera. Therefore it is described by only $2 + 2 + 1$ parameters (instead of $3 + 3 + 2$, that is the unconstrained case).

Generative and iterative tracking methods are proposed for estimating the pose parameters in both set-ups. Furthermore, methods for robustly obtaining robust and relevant behavioral measurements from the tracking data are proposed. Lastly, a preliminary behavioral analysis of recorded data from a set of experiments on the 6-OHDA model of Parkinson's disease in marmosets is presented.

Paper IV - Characterization of Long-Term Motor Deficits in the 6-OHDA Model of Parkinson's Disease in the Common Marmoset

The paper presents an application of the methods proposed in paper III for the purpose of characterizing the long-term motor deficits in the the 6-OHDA common marmoset model of Parkinson's disease. A larger data set than presented in paper III is analyzed and additional measures for analyzing the motor deficits are proposed.

Paper V - Visual Analysis of Zebrafish Behavior

This paper propose a high-throughput system for analyzing the behavior of zebrafish larvae. 48 spatially separated larvae are recorded in videos of 640×480 pixels resolution in which they appear as curves of $\sim$ 2 pixels width and $\sim$ 8 pixels length. The larvae are modeled as curves consisting of eight points, starting at the head position. For each frame and larvae, the parameters of the eight-point-skeleton model are estimated. Thus is provided 48 time series containing information of the position, direction and bending of the larvae over time.

Each time series is separated into discrete *swim bouts* by applying a threshold to the estimated speed function. Methods for classification of swim bouts into K groups is proposed. The distributions over time of bout classes in disjoint time bins are analyzed for experiments on amphetamine and apomorphine, and are compared to those of a control experiment.

This paper provides more in-depth descriptions of the computer vision algorithms employed in paper VI. A subset of the data and methods of analysis proposed in paper VI is used and described.

Paper VI - Action Sequencing in the Spontaneous Behavior of Zebrafish Larvae with Implications for Drug Development

This paper presents an application of the methods proposed in paper V. A larger dataset is used and additional methods for analysis are proposed. Drugs with known effects are studied and previously known effects are reproduced.

The swim bout classifier is used to convert the tracking data time series to time series of classification results. Thus the behavior of each larva is described by a sequence of numbers. These new time series are then modeled as n-th order Markov chains, and the transitional probabilities of which are estimated. A number of measures (speed, duration, transitional probabilities, etc.) are used to create a 435-dimensional feature vector for each larva.

The behavior of zebrafish larvae in a control experiment is first characterized. Then the behavior of larvae given different doses of amphetamine, apomorphine and MK-801 is analyzed using the 435-dimensional feature vector in disjoint time bins.

In conclusion, it is shown that the tracking and behavioral analysis methods proposed in paper V are of practical relevance. Furthermore, it is shown that the sequencing of motor behavior of zebrafish-larvae are of importance for drug development.

Paper VII - Tracking the Motion of Box Jellyfish

This paper presents a simple framework for tracking the motion of box jellyfish. Box jellyfish are fastened such that they can make swimming motions but not move. The bell of the jellyfish is translucent and difficult to measure the movement of, but the *rhopalia* (clusters of eyes) are visible and relatively simple to detect. The movement of the rhopalia provides a coarse measurement of the movement of the bell, thus also the swimming behavior of the jellyfish. For this purpose, a method for detecting and tracking the rhopalia is proposed.

Paper VIII - Calibration, Positioning and Tracking in a Refractive and Reflective Scene

This paper proposes a method for tracking objects within a scene where objects are observed through refraction and reflections.

The trigonometric equation describing refractions (Snell's law) is reformulated on vector form using a property of the cross product. Reflections are managed by reflecting points instead of ray-directions, thus creating *virtual points*. These are shown to be a useful tool. A bundle adjustment framework is developed for this purpose, allowing for optimization of scene points (i.e. triangulation), image points (i.e. projection) and refractive and reflective planes (here denoted *scene calibration*). The proposed system for positioning, calibration and tracking is applied on videos from a biological experiment on *Daphnia*.

Paper IX - Daphnia Response to Multiple Threats from UV-A and Predation

This paper presents an application of the method presented in paper VIII. The response of *Daphnia* to the multiple threats introduced by UV-A radiation from above and predation from below is investigated.

Paper X - The Misty Three Point Algorithm for Relative Pose

This paper propose a method for estimating the relative motion between two cameras in an underwater setting given three points. The idea behind the paper is to leverage the depth-information caused by a distance dependent attenuation of light that can be observed in underwater images to estimate camera motion.

The exponential attenuation of light is modeled mathematically by the Jaffe-McGlamery equation. The observed colors of three pairs of point-correspondences are used to create a system of equations where the difference in depth (Δz_1, Δz_2 and Δz_3) can be solved for up to an unknown scale. Thus, pseudo depth data are obtained.

By defining z_1, z_2 and z_3 as the distances from the first camera to the scene points, the distances from the second camera to the scene points are $z_1 + \Delta z_1$, $z_2 + \Delta z_2$ and $z_3 + \Delta z_3$ (using the previously estimated depth differences). A set of equations for the relative pose problem are then provided by noting that the Gramian for the scene points at distance z_k from the first camera is constrained to be equal to the Gramian for the scene points at distance $z_k + \Delta z_k$ from the second camera. This results in a system of three polynomial equations in three unknown

(z_1, z_2 and z_3). The action matrix method is employed for solving the system, i.e. for computing z_1, z_2 and z_3. Thereby the scene points can be reconstructed and the pose of the second camera can be computed by the standard resectioning algorithms.

References

[1] Mariam Alaverdashvili and Ian Q Whishaw. "A behavioral method for identifying recovery and compensation: hand use in a preclinical stroke model using the single pellet reaching task". In: *Neuroscience & Biobehavioral Reviews* 37.5 (2013), pp. 950–967.

[2] Yuval Bahat and Michal Irani. "Blind dehazing using internal patch recurrence". In: *Computational Photography (ICCP), 2016 IEEE International Conference on*. IEEE. 2016, pp. 1–9.

[3] J Bednar and T Watt. "Alpha-trimmed means and their relationship to median filters". In: *IEEE Transactions on acoustics, speech, and signal processing* 32.1 (1984), pp. 145–153.

[4] Giuseppe Bianco, Mikael Tobias Ekvall, Johan Bäckman, and Lars-Anders Hansson. "Plankton 3D tracking: the importance of camera calibration in stereo computer vision systems". In: *Limnology and Oceanography-Methods* 11 (2013), pp. 278–286.

[5] Federica Bogo, Michael J. Black, Matthew Loper, and Javier Romero. "Detailed Full-Body Reconstructions of Moving People From Monocular RGB-D Sequences". In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[6] Jean-Yves Bouguet. "Camera calibration toolbox for matlab". In: (2004).

[7] Kristin Branson and Serge Belongie. "Tracking multiple mouse contours (without too many samples)". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1. IEEE. 2005, pp. 1039–1046.

[8] Laura J Brattain, Rogier Landman, Kerry A Johnson, Patrick Chwalek, Julia Hyman, Jitendra Sharma, Charles Jennings, Robert Desimone, Guoping Feng, and Thomas F Quatieri. "A multimodal sensor system for automated marmoset behavioral analysis". In: *Wearable and Implantable Body Sensor Networks (BSN), 2016 IEEE 13th International Conference on*. IEEE. 2016, pp. 254–259.

[9] Martin Bujnak, Zuzana Kukelova, and Tomas Pajdla. "Making minimal solvers fast". In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 1506–1513.

[10] Martin Byröd, Klas Josephson, and Kalle Åström. “Fast and stable polynomial equation solving and its application to computer vision”. In: *International Journal of Computer Vision* 84.3 (2009), pp. 237–256.

[11] Jonathan Cachat, Adam Stewart, Eli Utterback, Peter Hart, Siddharth Gaikwad, Keith Wong, Evan Kyzar, Nadine Wu, and Allan V Kalueff. “Three-dimensional neurophenotyping of adult zebrafish behavior”. In: *PloS one* 6.3 (2011), e17597.

[12] John Canny. “A computational approach to edge detection”. In: *IEEE Transactions on pattern analysis and machine intelligence* 6 (1986), pp. 679–698.

[13] Cristian Sminchisescu Catalin Ionescu Fuxin Li. “Latent Structured Models for Human Pose Estimation”. In: *International Conference on Computer Vision*. 2011.

[14] M Chasles. “Question 296”. In: *Nouv. Ann. Math* 14.50 (1855), p. 1855.

[15] Nathan G Clack, Daniel H O’Connor, Daniel Huber, Leopoldo Petreanu, Andrew Hires, Simon Peron, Karel Svoboda, and Eugene W Myers. “Automated tracking of whiskers in videos of head fixed rodents”. In: *PLoS Comput Biol* 8.7 (2012), e1002591.

[16] Kathleen M Conlee, Erika H Hoffeld, and Martin L Stephens. “A demographic analysis of primate research in the United States.” In: *Alternatives to laboratory animals: ATLA* 32 (2004), pp. 315–322.

[17] David Cox, John Little, and Donal O’shea. *Ideals, varieties, and algorithms*. Vol. 3. Springer, 1992.

[18] Robbert Creton. “Automated analysis of behavior in zebrafish larvae”. In: *Behavioural brain research* 203.1 (2009), pp. 127–136.

[19] Franklin C Crow. “Summed-area tables for texture mapping”. In: *ACM SIGGRAPH computer graphics* 18.3 (1984), pp. 207–212.

[20] Christoph M Decker and Fred A Hamprecht. “Detecting individual body parts improves mouse behavior classification”. In: *Proc. of the Workshop on Visual Observation and Analysis of Vertebrate and Insect Behavior, Stockholm, Sweden*. 2014.

[21] Jonathan Deutscher and Ian Reid. “Articulated body motion capture by stochastic search”. In: *International Journal of Computer Vision* 61.2 (2005), pp. 185–205.

[22] Jeroen R Dijkstra, Marcel F Meek, Peter H Robinson, and Albert Gramsbergen. “Methods to evaluate functional nerve recovery in adult rats: walking track analysis, video analysis and the withdrawal reflex”. In: *Journal of neuroscience methods* 96.2 (2000), pp. 89–96.

[23] Mikael T Ekvall, Giuseppe Bianco, Sara Linse, Heiner Linke, Johan Bäckman, and Lars-Anders Hansson. “Three-dimensional tracking of small aquatic organisms using fluorescent nanoparticles”. In: *PloS one* 8.11 (2013), e78498.

[24] Mikael T. Ekvall, Tobias Palmér, Giuseppe Bianco, Jan Heuschele, Johan Bäckman, Kalle Åström, and Lars-Anders Hansson. “Daphnia Response to Multiple Threats from UV-A and Predation”.

[25] Andrew Fitzgibbon, Maurizio Pilu, and Robert B Fisher. “Direct least square fitting of ellipses”. In: *IEEE Transactions on pattern analysis and machine intelligence* 21.5 (1999), pp. 476–480.

[26] Ebraheem Fontaine, David Lentink, Sander Kranenbarg, Ulrike K Müller, Johan L van Leeuwen, Alan H Barr, and Joel W Burdick. “Automated visual tracking for studying the ontogeny of zebrafish swimming”. In: *Journal of Experimental Biology* 211.8 (2008), pp. 1305–1316.

[27] Luca Giancardo, Diego Sona, Huiping Huang, Sara Sannino, Francesca Managò, Diego Scheggia, Francesco Papaleo, and Vittorio Murino. “Automatic visual tracking and social behaviour analysis with multiple mice”. In: *PloS one* 8.9 (2013), e74557.

[28] G Gyory, V Rankov, G Gordon, I Perkon, B Mitchinson, R Grant, and T Prescott. “An algorithm for automatic tracking of rat whiskers”. In: *Proc Int Workshop on Visual observation and Analysis of Animal and Insect Behavior (VAIB), Istanbul, in conjunction with ICPR*. Vol. 2010. 2010, pp. 1–4.

[29] Frank PT Hamers, Alex J Lankhorst, Teus Jan van Laar, Wouter B Veldhuis, and Willem Hendrik Gispen. “Automated quantitative gait analysis during overground locomotion in the rat: its

application to spinal cord contusion and transection injuries". In: *Journal of neurotrauma* 18.2 (2001), pp. 187–201.

[30] Sebastian Haner and Kalle Åström. "Absolute Pose for Cameras Under Flat Refractive Interfaces". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1428–1436.

[31] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[32] Kaiming He, Jian Sun, and Xiaoou Tang. "Single image haze removal using dark channel prior". In: *IEEE transactions on pattern analysis and machine intelligence* 33.12 (2011), pp. 2341–2353.

[33] Janne Heikkila and Olli Silvén. "A four-step camera calibration procedure with implicit image correction". In: *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*. IEEE. 1997, pp. 1106–1112.

[34] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. "Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.7 (July 2014), pp. 1325–1339.

[35] Jules S Jaffe. "Computer modeling and the design of optimal underwater imaging systems". In: *Oceanic Engineering, IEEE Journal of* 15.2 (1990), pp. 101–111.

[36] Alejandro Jaimes and Nicu Sebe. "Multimodal human–computer interaction: A survey". In: *Computer vision and image understanding* 108.1 (2007), pp. 116–134.

[37] Anne Jordt. "Underwater 3D Reconstruction Based on Physical Models for Refraction and Underwater Light Propagation". PhD thesis. Universitätsbibliothek Kiel, 2013.

[38] Anne Jordt-Sedlazeck and Reinhard Koch. "Refractive structure-from-motion on underwater images". In: *Proceedings of the IEEE international Conference on Computer Vision*. 2013, pp. 57–64.

[39] Tobias Kjellberg, Tobias Palmér, Magnus Oskarsson, and Kalle Åström. "Tracking the Motion of Box Jellyfish". In: *Visual observation and Analysis of Vertebrate and Insect Behavior 2014*. 2014.

[40] Laurent Kneip, Roland Siegwart, and Marc Pollefeys. “Finding the exact rotation between two images independently of the translation”. In: *European conference on computer vision*. Springer. 2012, pp. 696–709.

[41] E. Kruppa. “Zur Ermittlung eines Objektes aus Zwei Perspektiven mit innerer Orientierung”. In: *Sitz-Ber. Akad. Wiss., Wien, math. naturw. Kl. Abt* IIa.122 (1913), pp. 1939–1948.

[42] Yubin Kuang and Kalle Åström. “Numerically stable optimization of polynomial solvers for minimal problems”. In: *European Conference on Computer Vision*. Springer. 2012, pp. 100–113.

[43] Zuzana Kukelova, Martin Bujnak, Jan Heller, and Tomáš Pajdla. “Singly-bordered block-diagonal form for minimal problem solvers”. In: *Asian Conference on Computer Vision*. Springer. 2014, pp. 488–502.

[44] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. “Automatic generator of minimal problem solvers”. In: *European Conference on Computer Vision*. Springer. 2008, pp. 302–315.

[45] Jan Kunze, Sarah Hartmann, Klaudia Witte, and Klaus-Dieter Kuhnert. “Daphnia magna as biosensor for Ag-nanoparticles in water systems: Development of a computer vision system for the detection of behavioral changes”. In: (2016).

[46] Po-Lun Lai, D Michele Basso, Lesley C Fisher, and Alison Sheets. “3D Tracking of mouse locomotion using shape-from-silhouette techniques”. In: *The 2012 International Conference on Image Processing, Computer Vision, and Pattern Recognition (Las Vegas, Nevada)*. 2011.

[47] JMF Landsmeer. “The coordination of finger-joint motions”. In: *J Bone Joint Surg Am* 45.8 (1963), pp. 1654–1662.

[48] Mercy Lard, Johan Bäckman, Maria Yakovleva, Bengt Danielsson, and Lars-Anders Hansson. “Tracking the small with the smallest–using nanotechnology in tracking zooplankton”. In: *PloS one* 5.10 (2010), e13516.

[49] John P Lewis. “Fast template matching”. In: *Vision interface*. Vol. 95. 120123. 1995, pp. 15–19.

[50] H Christopher Longuet-Higgins. “A computer algorithm for reconstructing a scene from two projections”. In: *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms, MA Fischler and O. Firschein, eds* (1987), pp. 61–62.

[51] Elena I Miklyaeva, Eddie Castañeda, and Ian Q Whishaw. “Skilled reaching deficits in unilateral dopamine-depleted rats: impairments in movement and posture and compensatory adjustments”. In: *The Journal of neuroscience* 14.11 (1994), pp. 7148–7158.

[52] Elena I Miklyaeva, David J Martens, and Ian Q Whishaw. “Impairments and compensatory adjustments in spontaneous movement after unilateral dopamine depletion in rats”. In: *Brain research* 681.1 (1995), pp. 23–40.

[53] August Ferdinand Möbius. *Der barycentrische calcul*. 1827.

[54] H Michael Möller and Hans J Stetter. “Multivariate polynomial equations with multiple zeros solved by matrix eigenproblems”. In: *Numerische Mathematik* 70.3 (1995), pp. 311–329.

[55] David Nistér. “An efficient solution to the five-point relative pose problem”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.6 (2004), pp. 756–770.

[56] Tobias Palmér, Kalle Åström, Olof Enqvist, Nela Ivica, and Per Petersson. “Rat Paw Tracking for Detailed Motion Analysis”. In: *Visual observation and analysis of Vertebrate And Insect Behavior 2014*. 2014.

[57] Tobias Palmér, Kalle Åström, Olof Enqvist, and Per Petersson. “Visual Analysis of Zebrafish Behavior”. In: *Visual Observation and Analysis of Vertebrate and Insect Behavior 2016*. 2016.

[58] Tobias Palmér, Kalle Åström, and Jan-Michael Frahm. “The Misty Three Point Algorithm for Relative Pose”.

[59] Tobias Palmér, Giuseppe Bianco, Mikael T. Ekvall, Lars-Anders Hansson, and Kalle Åström. “Calibration, Positioning and Tracking in a Refractive and Reflective Scene”. In: *International Conference on Pattern Recognition*. 2016.

[60] Tobias Palmér, Fredrik Ek, Olof Enqvist, Roger Olsson, Kalle Åström, and Per Petersson. “Action Sequencing in the Spontaneous Behavior of Zebrafish Larvae with Implications for Drug Development”.

[61] Tobias Palmér, Maxwell B. Santana, Romulo A. Fuentes, and Per Petersson. “Automated Tracking of Motor Behavior as a Means to Assess Severity of Symptoms in the 6-OHDA Marmoset Model of Parkinsons Disease”. In: *Visual observation and analysis of Vertebrate And Insect Behavior 2012*. 2012.

[62] Tobias Palmér, Martin Tamtè, Pär Halje, Olof Enqvist, and Per Petersson. “A System for Automated Tracking of Motor Components in Neurophysiological Research”. In: *Journal of neuroscience methods* 205.2 (2012), pp. 334–344.

[63] Jacquelin Perry and Judith M Burnfield. “Gait analysis: normal and pathological function”. In: (1992).

[64] George Maxwell Peterson. *Mechanisms of handedness in the rat*. Johns Hopkins Press, 1934.

[65] Geo Peterson, La Charles Fracarol, et al. “The relative influence of the locus and mass of destruction upon the control of handedness by the cerebral cortex”. In: *Journal of Comparative Neurology* 68.2 (1938), pp. 173–190.

[66] Jose Pinheiro, Douglas Bates, Saikat DebRoy, Deepayan Sarkar, and R Core Team. *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-128. 2016. URL: `http://CRAN.R-project.org/package=nlme`.

[67] Siddharth S Rautaray and Anupam Agrawal. “Vision based hand gesture recognition for human computer interaction: a survey”. In: *Artificial Intelligence Review* 43.1 (2015), pp. 1–54.

[68] Karl Rohr. “Towards model-based recognition of human movements in image sequences”. In: *CVGIP: Image understanding* 59.1 (1994), pp. 94–115.

[69] Lori-Ann R Sacrey, Mariam Alaverdashvili, and Ian Q Whishaw. “Similar hand shaping in reaching-for-food (skilled reaching) in rats and humans provides evidence of homology in release, collection, and manipulation movements”. In: *Behavioural brain research* 204.1 (2009), pp. 153–161.

[70] Maxwell B. Santana, Tobias Palmér, Hougelle Simplício, Romulo A. Fuentes, and Per Petersson. “Characterization of Long-Term Motor Deficits in the 6-OHDA Model of Parkinson’s Disease in the Common Marmoset”. In: *Behavioural brain research* 290 (2015), pp. 90–101.

[71] Benjamin Sapp, Alexander Toshev, and Ben Taskar. “Cascaded models for articulated pose estimation”. In: *European conference on computer vision*. Springer. 2010, pp. 406–420.

[72] Yoav Y Schechner and Nir Karpel. “Clear underwater vision”. In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 1. IEEE. 2004, pp. I–536.

[73] Yoav Y Schechner and Nir Karpel. “Recovery of underwater visibility and structure by polarization analysis”. In: *IEEE Journal of Oceanic Engineering* 30.3 (2005), pp. 570–587.

[74] Yoav Y Schechner, Srinivasa G Narasimhan, and Shree K Nayar. “Instant dehazing of images using polarization”. In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE. 2001, pp. I–I.

[75] Leonid Sigal, Alexandru O Balan, and Michael J Black. “Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion”. In: *International journal of computer vision* 87.1-2 (2010), pp. 4–27.

[76] Leonid Sigal and Michael J Black. “Measure locally, reason globally: Occlusion-sensitive articulated pose estimation”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*. Vol. 2. IEEE. 2006, pp. 2041–2048.

[77] P Stefanovic. “Relative orientation–a new approach”. In: *ITC Journal* 3 (1973), pp. 417–448.

[78] Bjoern Stenger, Paulo RS Mendonça, and Roberto Cipolla. “Model-based 3D tracking of an articulated hand”. In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2001, pp. II–310.

[79] H. Stewénius, D. Nistér, M. Oskarsson, and K. Åström. "Solutions to Minimal Generalized Relative Pose Problems". In: *Workshop on Omnidirectional Vision*. Beijing China, Oct. 2005.

[80] Henrik Stewénius. "Gröbner Basis Methods for Minimal Problems in Computer Vision". PhD thesis. Lund University, 2005.

[81] Henrik Stewénius, David Nistér, Magnus Oskarsson, and Kalle Åström. "Solutions to minimal generalized relative pose problems". In: *Workshop on omnidirectional vision*. Vol. 1. 2. 2005, p. 3.

[82] Peter Sturm. "A historical survey of geometric computer vision". In: *Computer Analysis of Images and Patterns*. Springer. 2011, pp. 1–8.

[83] Matan Sulami, Itamar Glatzer, Raanan Fattal, and Mike Werman. "Automatic recovery of the atmospheric light in hazy images". In: *Computational Photography (ICCP), 2014 IEEE International Conference on*. IEEE. 2014, pp. 1–11.

[84] Ivan E Sutherland. "Three-dimensional data input by tablet". In: *Proceedings of the IEEE* 62.4 (1974), pp. 453–461.

[85] R Core Team. *R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 2013*. 2014. URL: `URL:%20http://www.R-project.org/`.

[86] Jing Tian, Amit Satpathy, Ee Sin Ng, Soh Guat Ong, Wei Cheng, Jean-Marc Burgunder, and Walter Hunziker. "Motion analytics of zebrafish using fine motor kinematics and multi-view trajectory". In: *Multimedia Systems* (2014), pp. 1–11.

[87] Hjalmar K Turesson, Thamiris Botelho Ribeiro Conceicao, and Sidarta Ribeiro. "Automatic Head Tracking of The Common Marmoset." In: *bioRxiv* (2016), p. 079566.

[88] Paul Viola and Michael Jones. "Robust real-time object detection". In: *International Journal of Computer Vision* 4 (2001).

[89] Jakob Voigts, Bert Sakmann, and Tansu Celikel. "Unsupervised whisker tracking in unrestrained behaving animals". In: *Journal of neurophysiology* 100.1 (2008), pp. 504–515.

[90] IAN Q WHISHAW, WILLIAM T O'CONNOR, and STEPHEN B DUNNETT. "The contributions of motor cortex, nigrostriatal dopamine and caudate-putamen to skilled forelimb use in the rat". In: *Brain* 109.5 (1986), pp. 805–843.

[91] Ian Q Whishaw, Sergio M Pellis, and Boguslaw P Gorny. "Skilled reaching in rats and humans: evidence for parallel development or homology". In: *Behavioural brain research* 47.1 (1992), pp. 59–70.

[92] Michael W Whittle. *Gait analysis: an introduction*. Butterworth-Heinemann, 2014.

[93] Tai-Hsien Ou-Yang, Meng-Li Tsai, Chen-Tung Yen, and Ta-Te Lin. "An infrared range camera-based approach for three-dimensional locomotion tracking and pose reconstruction in a rodent". In: *Journal of neuroscience methods* 201.1 (2011), pp. 116–123.

[94] Gale Young and Alston S Householder. "Discussion of a set of points in terms of their mutual distances". In: *Psychometrika* 3.1 (1938), pp. 19–22.

[95] Fisher Yu and Vladlen Koltun. "Multi-Scale Context Aggregation by Dilated Convolutions". In: *ICLR*. 2016.

[96] Zhengyou Zhang. "A flexible new technique for camera calibration". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22.11 (2000), pp. 1330–1334.

Paper I

Rat Paw Tracking for Detailed Motion Analysis

TOBIAS PALMÉR[1,2], ÅSTRÖM[1], OLOF ENQVIST[3], NELA IVÍCA[2] AND PER PETERSSON[2]

[1] *Centre for Mathematical Sciences, Lund University, Sweden*
[2] *Integrative Neurophysiology and Neurotechnology, NRC, Department of Experimental Medical Sciences, Lund University, Sweden*
[3] *Department of Signals and Systems, Chalmers University of Technology, Sweden*

Abstract: This work is part of a research project studying the learning of fine motor skills in rats. A system for tracking a rat paw using a system of high-speed cameras is presented along with preliminary analysis of the correlation between paw movement and neural data. The tracking method is generative, modeling the rat paw as a set of linked ellipsoids. To find the most probable paw pose a number of hypothetical parameter values are explored. For each set of values the paw model is projected in the different cameras and compared to the actual measured images. The pose that is most consistent with the intensity and edge information in the different views is chosen. By efficient utilization of integral images, this evaluation can be performed very quickly, creating a tractable and completely automatic method with good performance.

1 Introduction

1.1 Background

The central nervous system fundamentally deals with the control of actions. Consequently behavioral studies have often been a natural starting point for investigations aimed at understanding its functions. The goal of this work is to study neuronal activity of the central nervous system during learning of new behaviours. As shown in [4], we have developed a system for analysis of motor behaviour of rats during skilled reaching experiments in order to learn about the changes occurring in the central nervous system.

There are several ways to improve the system from [4]. Firstly, evaluation of the quality function has been improved from quadratic time to linear time, with respect to the length of the shortest side of the videos. This has made available more advanced methods for quality function optimization, i.e. both tracking quality and computational efficiency has been improved. Secondly, the depth of each projected quadric is used in order to detect occlusions from the experimental

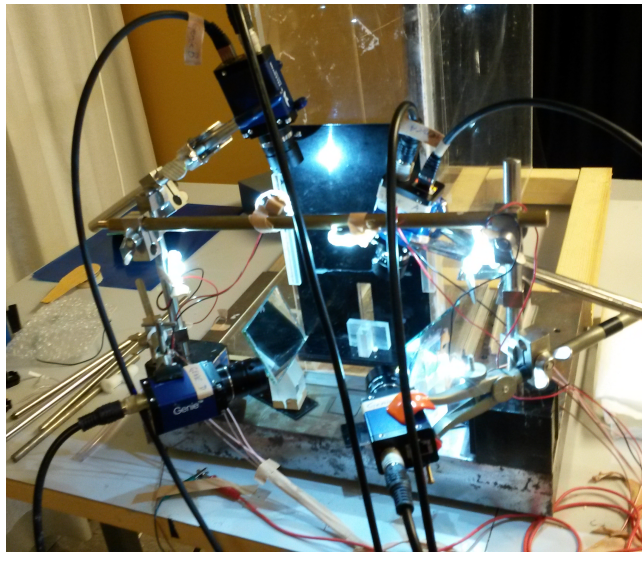


Figure I.1: The experimental setup with four cameras, lights and a calibration object.

setup. This makes the similarity measures used for quality evaluation more appropriate. As this is work-in-progress, there are not yet any behavioural recordings available using the latest experimental setup, thus there are no results presented using the new setup.

1.2 The behavioural task

In the behavioral experiment the rat was placed in the reaching apparatus from where it could obtain 45 mg food pellet rewards, positioned in an indentation on the reward shelf, via controlled reaches through the aperture of the wall. At the start of the experiment, the rats are slow and inefficient in this task, but after approximately two weeks of almost daily training, they have fully learned the reaching behaviour. This is interesting from a neural point of view as we then can study what happens in the brain as the rats learn this new behaviour.

1.3 The videos

An improvement from the setup in [4], is the use of four front-view cameras, as seen in Figure I.1. The previous version of the setup featured two combined with

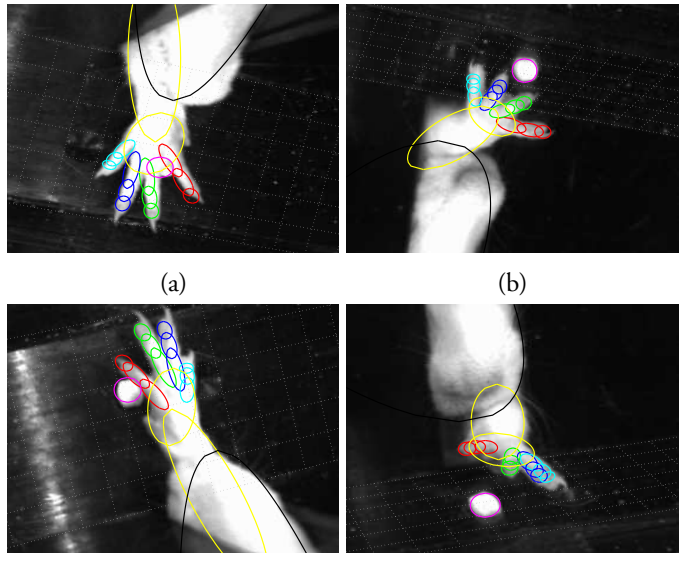


Figure I.2: Video inputs from four views in subfigures (a),(b),(c) and (d) with superimposed projections of pose estimation results. In this particular frame, the paw of the rat is approaching the food pellet through a slit in the wall. The nose can also be seen in this frame. The digits are illustrated in red, green, blue and cyan, where red corresponds to the human index finger, green corresponds to the middle finger, etc. Note that the rat paw only has four long digits; the thumb is considered too short and is not used in this model. The palm of the paw and the forearm are yellow, the nose is black and the food pellet is magenta.

mirrors, to generate six views. The use of mirrors introduced problems with the focus, as the distance for light to travel from the paw through one of the mirrors to the camera could be as much as twice as far as the closest distance to the camera. Additionally, recording speed was increased from 200 frames per second to 300 frames per second, to enable better tracking of the very quick movements of the rats. As no behavioural videos are yet recorded using the new camera setup, the old ones are presented in Figure I.2.

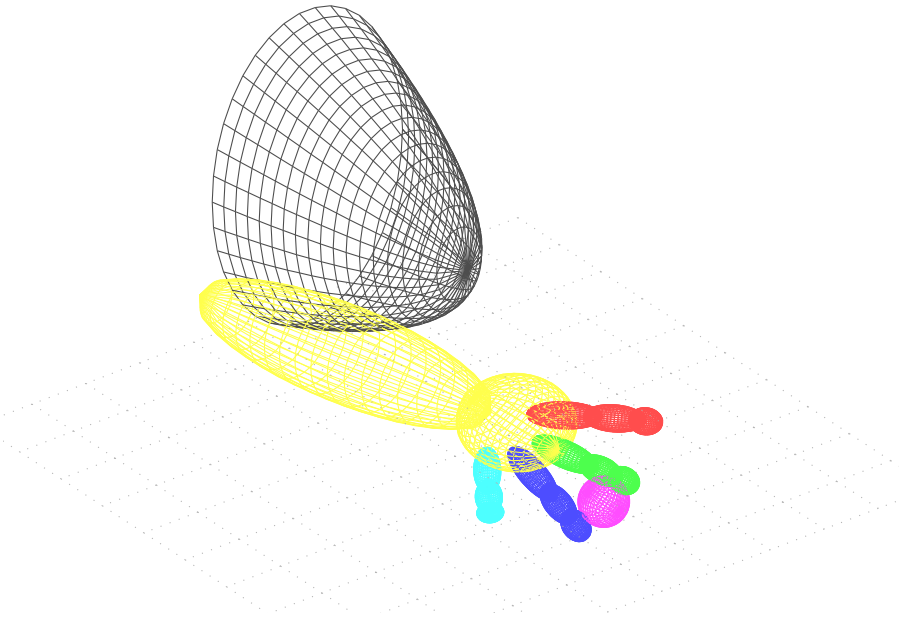


Figure I.3: The model for the paw, nose and pellet, with colors as explained in Fig I.2.

1.4 Calibration

Calibration was performed by placing an object with known geometry at a predefined location and then manually selecting the object corners in each of the videos. Given these selected image points, the camera matrix was computed using direct linear transform (DLT) [2].

2 Tracking

2.1 Related work

The subject of hand pose estimation and tracking has gained a lot of attention the last 5-10 years as the computational power available is approaching what is needed for real-time applications [1]. In human applications it is possible to use for example colored gloves [6],[7], or using a combination of normal cameras and

depth cameras [3]. However, for the presented application this is not possible as the movements are so fast and the paws are so small (less than 1cm in width), and the rats are not as collaborative as humans can be.

2.2 Overview of the method

The tracking method is generative, i.e. the paw is modelled as explained in section 2.3, below, with 22 variable parameters and these parameters are subsequently estimated as the argument of the maximum of the quality function given in section 2.7. Evaluation of the quality function is derived starting at the projection of quadrics in section 2.4, followed by the evaluation of the similarity of measured images and estimated poses in sections 2.5, 2.6 and 2.7. A brief presentation of the algorithm used for maximizing the quality measure in each frame and over time is presented in section 2.8.

2.3 Rat model

The rat paw consists of four long *digits* ("fingers") and each digit consists of three *phalanges* (bones). The paw is modelled by 13 quadrics, of which 12 represent ellipsoids for the phalanges and one represents an ellipsoid for the palm of the paw. Furthermore, the forearm is modelled as an ellipsoid and the nose is modelled as an elliptic paraboloid. In total, the rat pose is modelled using 15 quadrics, as seen in Figure I.3. Due to anatomical constraints, the kinematics of each digit can be modelled using only four degrees of freedom - one for adduction/abduction at the proximal joint and three for flexion at each one of the joints. Consequently, the paw can be modelled using 16 parameters for the digits, four constant vectors representing the metacarpal bones and 6 parameters for position and rotation of the palm of the paw. Furthermore, the forearm is assumed to be fixated at the wrist and can rotate along all three axes in space. This amounts to a total of 22 parameters.

2.4 Projection of quadrics

A quadric surface in 3D can be described as the solutions $\mathbf{X} = (X, Y, Z)$ to the quadratic equation

$$\begin{bmatrix} X & Y & Z & 1 \end{bmatrix} C \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = 0 \tag{I.1}$$

The projection of such a surface onto an image plane is a conic section which can be described as the solutions $\mathbf{x} = (x, y)$ to the quadratic equation

$$\begin{bmatrix} x & y & 1 \end{bmatrix} C \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0 \tag{I.2}$$

As described by [5], the conic matrix C can be easily computed along with the depth of any projected point on the quadric.

2.5 The quality function

The quality function used here is a measure of how well an estimated pose fits the intensity and edges of an image. The area quality function for one view is defined as

$$q_A = \frac{\sum_{x,y} \min(P(x,y), F(x,y))}{\sum_{x,y} \max(P(x,y), F(x,y))} \tag{I.3}$$

where P is the projection of the estimated paw pose and F is the measured foreground image. Here P is binary while F is normally not. Note that for binary functions, the quality function is identical to the Jaccard index, which is used in [4]. The measured foreground image F is defined as the image minus the background, i.e. $F = I - B$. The projection of the paw pose, P, is defined as the binary image with 1's at each pixel covered by a projected quadric of the estimated pose, and 0's elsewhere. Thus, we are interested in evaluating how much of the projection of a quadric covers the foreground of an image, i.e. to compute the sum of the elements in an image that are covered by the filled conic. The naive way to evaluate which pixels are covered by a conic is to evaluate (I.2) for each

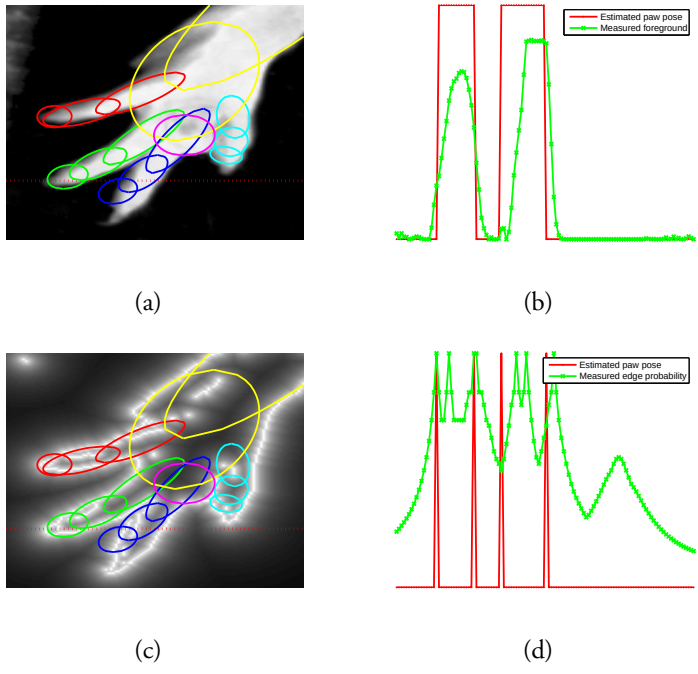


Figure I.4: An image (a) with superimposed projections of pose estimation results and a horizontal dotted line depicting the row of choice for (b). (b) is an illustration of the row shown in (a), where the red line shows where the image is covered by the estimated paw pose and the green line is the grayscale values of the row. (c) is an illustration of the edge-probability used in (I.7) and (d) is, analogously to (b), an illustration of the particular row of choise in (c).

pixel, which would require MN evaluations of (I.2). This is the method used in [4] and is very time consuming. This can be improved by noting that for each fixed $x_0 \in [1, M]$,

$$\begin{bmatrix} x_0 & y & 1 \end{bmatrix} C \begin{bmatrix} x_0 \\ y \\ 1 \end{bmatrix} = 0, \tag{I.4}$$

is a quadratic equation in y. Then the line segment $\{y_0\} \times [y_0, y_1]$, where y_0 and y_1 are solutions to (I.4), covers the conic. Each such slice can be computed by finding the real roots of the quadratic equation. If there are no real roots, the

conic does not intersect that particular row, x_0. As the estimated paw pose consists of 16 quadrics, for each row x_0 there is a (potentially empty) set of intervals that cover parts of the row. After merging overlapping intervals, we get a set of disjoint intervals for each row. Computing the contribution to (I.3) from one such interval, can be done very efficiently using integral images. Let $I_F(x, y)$ be the integral foreground image

$$I_F(x, y) = \sum_{j=0}^{y} F(x, j) \tag{I.5}$$

Then the contribution to (I.3) from an interval $[y_0, y_1]$ is given by

$$I_F(x, y_1) - I_F(x, y_0) \tag{I.6}$$

Thus the cover of a projected quadric can be computed in linear time (with respect to the smallest side of an image). For each row, 16 equations equations need to be solved, up to 16 intervals merged into disjoint intervals and grayscale values of the integral images at the endpoints of each interval needs to be evaluated. This is considerably faster than naive evaluation of (I.3).

2.6 Edge quality function

Let D be a matrix where, for each coordinate (x, y), $D(x, y)$ is the shortest distance from (x, y) to an edge in an image I (can be computed efficiently using distance transforms). Then for each point (x, y) on the edge of the projected paw pose, $\frac{1}{D(x,y)+1}$ is related to the probability of (x, y) corresponding to an edge. The edge quality function is defined as

$$q_E = \frac{1}{|\partial P|} \sum_{(x,y)\in\partial P} \frac{1}{D(x, y) + 1}. \tag{I.7}$$

2.7 The quality function

For a predicted paw pose, a point in time and each camera, the quality function is defined as (I.3). The quality in each of the views are multiplied to give the quality function for a predicted pose and a point in time:

$$q = \prod_{k=1}^{N} \left(q_A^{(k)} \lambda + (1 - \lambda) q_E^{(k)} \right) \tag{I.8}$$

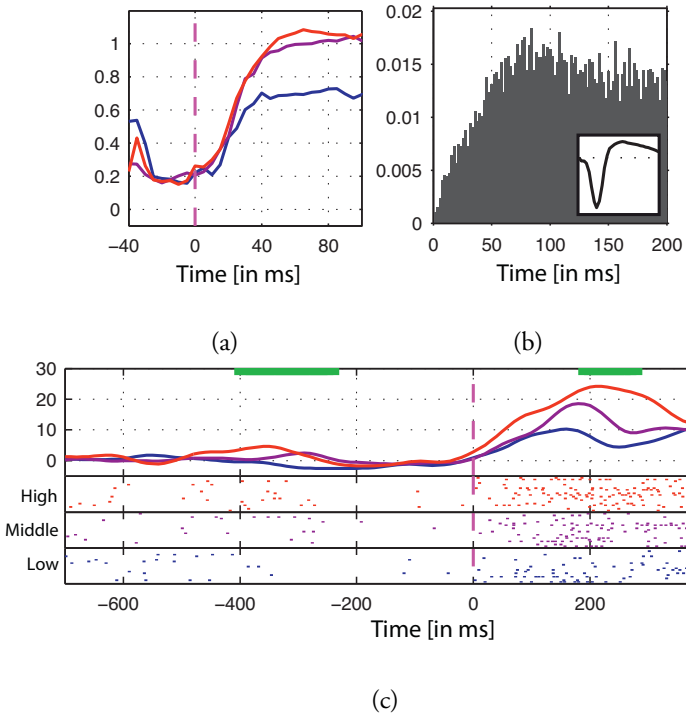


Figure I.5: Neural correlation to motor components. Examples of neurons correlated to individual motor components. Reach attempts are sorted depending on the peak value for a motor component during the attempt, as shown in (a). (c) is sorted for grasp so that the lowest degree of fist closure are pooled into the blue group, the intermediate closures into the purple group and the highest into the red group. (c) show standardized peri-event firing rates and raster plots of individual trials for each group, aligned to time point of maximum paw extension and averaged over all attempts in each group. A green line indicates a significant difference in firing rates between any of the three groups ($p < 0.05$). (b) shows the autocorrelogram and waveform for each presented neuron. Modified from [4].

2.8 Maximizing the quality function

As the movement of the paw was so quick with respect to the recording speed, using temporal consistency to improve tracking proved to be difficult. Thus the method presented here is only based on frame-by-frame pose estimation. For each frame, a number of previously encountered poses together with the pose of

the previous frame was evaluated, to find an appropriate starting point. Then a gradient-descent like optimization method was applied to find a local maxima, which given that the more coarse global optimization is close to the global maxima, is also the global maxima.

3 Alignment of neural recordings to kinematic data

Nerve cells in the brain communicate by briefly altering their membrane potential, these impulses are referred to as action potentials. Already in the first analyses of the changes in neuronal activity patterns related to different aspects of the motor output, we could identify single nerve cells that appeared to show specific modulations in the frequency of emitted action potentials in relation to specific parts of the motor sequence. In Figure I.5, the recorded neuronal activity of a nerve cell, in multiple reaching trials, was aligned to the part of the reaching movement where grasping is initiated. It was noted that the average number of action potentials detected scaled with the degree of fist closure suggesting that this nerve cell may have a role in controlling this part of the reaching and grasping compound movement.

4 Discussion

The presented system uses a calibration procedure (DLT) that is easy to code but is numerically unstable and is a potential bottleneck for the rest of the system. Even though calibration quality has been improved by using views with higher resolution, we aim to improve this part by using a new calibration method that will also be easier to perform during the experiments.

References

[1] Ali Erol, George Bebis, Mircea Nicolescu, Richard D Boyle, and Xander Twombly. “Vision-based hand pose estimation: A review”. In: *Computer Vision and Image Understanding* 108.1 (2007), pp. 52–73.

[2] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second Edition. Cambridge University Press, 2004.

[3] Meng-Fen Ho, Chuan-Yu Tseng, Cheng-Chang Lien, and Chung-Lin Huang. “A multi-view vision-based hand motion capturing system”. In: *Pattern Recognition* 44.2 (2011), pp. 443–453.

[4] Tobias Palmér, Martin Tamtè, Pär Halje, Olof Enqvist, and Per Petersson. “A System for Automated Tracking of Motor Components in Neurophysiological Research”. In: *Journal of neuroscience methods* 205.2 (2012), pp. 334–344.

[5] Bjoern Stenger, Paulo RS Mendonça, and Roberto Cipolla. “Model-based 3D tracking of an articulated hand”. In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2001, pp. II–310.

[6] Jorge Usabiaga, Ali Erol, George Bebis, Richard Boyle, and Xander Twombly. “Global hand pose estimation by multiple camera ellipse tracking”. In: *Machine Vision and Applications* 21.1 (2009), pp. 1–15.

[7] Robert Y Wang and Jovan Popović. “Real-time hand-tracking with a color glove”. In: *ACM Transactions on Graphics (TOG)*. Vol. 28. 3. ACM. 2009, p. 63.

Paper II

Journal of Neuroscience Methods 205 (2012) 334–344

Contents lists available at SciVerse ScienceDirect

Journal of Neuroscience Methods

journal homepage: www.elsevier.com/locate/jneumeth

Basic Neuroscience

A system for automated tracking of motor components in neurophysiological research

Tobias Palmér[a,1], Martin Tamtè[a,*,1], Pär Halje[a], Olof Enqvist[b], Per Petersson[a]

[a] Neuronano Research Center, Department of Experimental Medical Science, BMC F10, Lund University, 221 84 Lund, Sweden
[b] Center for Mathematical Sciences, Lund University, 22100 Lund, Sweden

ARTICLE INFO

Article history:
Received 30 June 2011
Received in revised form 11 January 2012
Accepted 12 January 2012

Keywords:
Reaching
3D-model
Hand pose estimation
Corticostriatal
Motor control

ABSTRACT

In the study of motor systems it is often necessary to track the movements of an experimental animal in great detail to allow for interpretation of recorded brain signals corresponding to different control signals. This task becomes increasingly difficult when analyzing complex compound movements in freely moving animals. One example of a complex motor behavior that can be studied in rodents is the skilled reaching test where animals are trained to use their forepaws to grasp small food objects, in many ways similar to human hand use. To fully exploit this model in neurophysiological research it is desirable to describe the kinematics at the level of movements around individual joints in 3D space since this permits analyses of how neuronal control signals relate to complex movement patterns. To this end, we have developed an automated system that estimates the paw pose using an anatomical paw model and recorded video images from six different image planes in rats chronically implanted with recording electrodes in neuronal circuits involved in selection and execution of forelimb movements. The kinematic description provided by the system allowed for a decomposition of reaching movements into a subset of motor components. Interestingly, firing rates of individual neurons were found to be modulated in relation to the actuation of these motor components suggesting that sets of motor primitives may constitute building blocks for the encoding of movement commands in motor circuits. The designed system will, thus, enable a more detailed analytical approach in neurophysiological studies of motor systems.

1. Introduction

The central nervous system fundamentally deals with the control of actions. Consequently behavioral studies have often been a natural starting point for investigations aimed at understanding its functions. For the same reason, the search for new therapies for neurological and psychiatric diseases largely depend on animal models designed to mimic certain aspects of the disease that cause observable changes in the behavior of the subject. With the more recent development of techniques allowing for simultaneous recording of neuronal activity in many parts of the central nervous system in freely behaving animals, the electrophysiological processes underlying such changes in behavior – or even the generation of specific components of observed actions – have the potential to be investigated in much greater detail (Nicolelis, 2008). The access to neuronal data with sub-millisecond temporal precision in turn further increases the need for more detailed documentation of movement patterns displayed by freely behaving animals. However, because natural behavior typically involves chains of movement sequences incorporating many partially overlapping motor components, an extra challenge in this respect is to reliably identify and isolate the execution of these individual motor elements. The most common approach for behavioral recording in neurophysiological research is probably the use of digital video techniques, where image sequences are obtained from different camera angles and specific behaviors are either manually identified off-line (Cenci and Lundblad, 2007; Whishaw et al., 1999) or, when clearly visible in any of the cameras, automatically identified and quantified from this viewing angle (Peikon et al., 2009; Vorhees et al., 1992). In situations where movements involve several joints the problem of tracking motor components involving for example small angle changes in distally located joints becomes increasingly complex. A well-studied and functionally very important example of movement sequences involving parallel movements in multiple joints in humans is the skilled arm and hand movements involved in reaching for and grasping of different objects (de Bruin et al., 2008; Gentilucci et al., 1997; Jeannerod, 1984). Perhaps surprisingly, rodents can after extensive training also perform reaching and grasping movements using their forepaw in many ways similar

* Corresponding author. Tel.: +46 46 222 14 64; fax: +46 46 222 05 78.
E-mail addresses: Tobias.palmer@med.lu.se (T. Palmér), Martin.tamte@med.lu.se (M. Tamtè), Par.Halje@med.lu.se (P. Halje), Olofe@maths.lth.se (O. Enqvist), Per.petersson@med.lu.se (P. Petersson).
[1] These authors contributed equally to this work.

0165-0270/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.jneumeth.2012.01.008

to a human hand, hence making this behavior particularly well suited for studies on skilled motor control in translational research (Peterson, 1934; Sacrey et al., 2009).

To be able to track forelimb movements with high fidelity in the skilled reaching task we designed a system that uses a three-dimensional (3D) model of the paw for which movements are reconstructed based on image sequences recorded from multiple viewing angles. Each paw pose is estimated by an optimization procedure that maximizes a matching quality measure in order to retrieve the best approximation of that pose. The matching quality is measured as the discrepancy between projections of the 3D model onto the image planes and the actual images, using edges and silhouettes as cues. We here describe how this system allows us to correlate single unit activity of neurons in corticostriatal circuits in rats to different motor components in the reaching-grasping sequence, opening up for significantly more detailed analyses of skilled movement control.

2. Materials and methods

2.1. Animals

One adult female Sprague–Dawley rat (230 g; Taconic Inc.) was used in the study. The animal was kept on 12:12 h light cycle and received food and water ad libitum except for a 22 h-period prior to each testing session during which no food was provided. After each testing session the animal had free access to food for 1 h. All experiments were approved in advance by the Malmö/Lund ethical committee of animal experiments.Training protocol

The rat was trained during a three week period prior to the implantation of recording electrodes according to the protocol described by Whishaw et al. (2008). Training entailed habituation of the rat to the apparatus, habituation to the food reward (pellet) and establishment of paw dominance. The training continued until performance no longer improved between sessions, reflecting a fully learned behavior (Hermer-Vazquez and Moshtagh, 2009). See Appendix A for details.

2.3. Electrodes

Formvar-insulated 33 μm tungsten wires (CFW Inc.) were arranged into four separate 4×5 arrays with 250 μm spacing between adjacent wires. Each array consisted of 16 recording channels and two reference channels, as well as two blind channels. The wires of each array were cut to the appropriate length for the corresponding recording site (cortical or striatal). Reference wires were cut ∼1 mm shorter than the recording wires and deinsulated ∼300 μm at the tip, positioning them dorsally to the recording site (at the cortical surface and within the corpus callosum, for cortical and striatal arrays, respectively). The wires were attached to board-to-board-connectors (Kyocera 5602) with conducting epoxy (Epotek EE 129-4), and linked to the acquisition device via a board-to-Omnetics connector adapter (Kyocera 5602; Omnetics). A 200 μm silver wire was used as animal ground via direct connection to four screws inserted into the cranium.

2.4. Surgery

Implantations were performed under Fentalyl/Medetomidine anesthesia (0.3/0.3 mg/kg, i.p.). Electrodes were implanted in the forelimb area of the primary motor cortex (center coordinates: AP: +1.5, ML: ±2.8, DV: −1.0 from cortical surface, Donoghue and Wise, 1982) and of the striatum, (center coordinates: AP: +0.2, ML: ±3.8, DV: −3.5 from cortical surface, West et al., 1990) in both hemispheres. The implant was fixated with dental acrylic attaching to screws in the skull. After surgery the anesthesia was reversed by Atipamezole hydrochloride (5 mg/kg, i.p.). Buprenorphine (0.5 mg/kg, s.c.) was administered as postoperative analgesic. The animal was allowed to recover for a week after implantation before testing commenced.

2.5. Experimental set-up

The testing apparatus for the reaching task consisted of a 450 mm × 140 mm × 350 mm (l/w/h) transparent Plexiglas cage with a 13 mm wide aperture at the middle of one of the short sides (vertical position of aperture: 40–150 mm above the ground). Outside the aperture a 30 mm deep shelf was positioned. To facilitate the placement of food pellets, three separate hemispherical indentations (5 mm in diameter) were made in the shelf 15 mm from the outer edge of the slit. The middle pocket was positioned right in front of the aperture with the other two pockets centered 6.5 mm more lateral on each side. This configuration prevented the rat from using its tongue to acquire the pellet. Furthermore, it permitted the experimenter to decide which paw the rat had to use, as this geometry allows only reaches with the paw contralateral to the side pockets (for further details, see Whishaw and Pellis, 1990). At the center of the cage was a 40 mm high solid obstacle that enforced a forelimb stepping movement similar to the actual reaching and grasping movement, for comparison of similar movements with different purposes.

2.6. Reaching task

In the behavioral task the rat was placed in the reaching apparatus from where it could obtain 45 mg food pellet rewards, positioned in the indentation on the reward shelf, via controlled reaches through the aperture of the wall. A trial ended either if the rat acquired the pellet after one or several reach attempts, or if the pellet at any time was moved from its original position, in which case the food pellet was manually removed by the experimenter. In order to produce discrete reaching trials, the rat was trained to return to the end of the cage opposite to the reaching slit before it was presented with another food pellet, requiring the animal to reposition before every trial. Moreover, by semi-randomly withholding food the rat was prompted to identify the presence of a pellet before each reach attempt, yielding maximal accuracy of each skilled reach (Appendix A).

2.7. Acquisition of neurophysiological signals

Extracellular neuronal recordings were acquired using a multichannel recording system with Cheetah software (Neuralynx Inc.) and digitized at 32 kHz per channel. Local field potentials were bandpass filtered between 0.1 and 300 Hz (not used in this study) and action potential waveforms between 600 and 9000 Hz.

2.8. Video acquisition systems

The details of the paw movement during the reach and grasp behavior were captured by two front-view cameras (CMOS, 640 × 480 pixels; Dalsa Inc.) positioned close to the aperture. Additionally, three mirrors positioned along the edges of the reward shelf gave two extra viewing planes for each front-view camera. Thus, the front-view cameras and mirrors were mounted such that six complementary viewing planes covered the region of interest where the rat forelimb was moving. Extra care was taken to avoid uneven light conditions or reflexes from surfaces. To ensure that inadvertent variations in the camera positions would not influence the 3D reconstruction a calibration procedure with an object of known measures was performed at the start of each recording.

This allowed for off-line determination of the mathematical function that best described the projections of 3D objects onto the image planes. The apparatus also included one side camera recording the movements of the rat within the box (Stingray, 640 × 480 pixels; Allied Vision Technologies).

The two front-view cameras were triggered by an external pulse generator (Master 8, A.M.P.I) acquiring images at 200 Hz during a time period manually controlled by a switch during the ongoing experiment, while the side-view camera continuously acquired images at 50 Hz. A set of images acquired from this set of cameras at the same moment in time is in the following referred to as a *multiframe*. In order to ensure perfect temporal alignment between the acquired images and the neuronal recordings, the timestamp for each multiframe was also stored in the Neuralynx multichannel recording system. Uncompressed image data was sampled from the three cameras on three separate computers, using Common Vision Blox software (Stemmer Imaging, Gmbh) and FireView (Allied Vision Technologies), for the front-view and side-view cameras, respectively. Post-acquisition compression of stored data was made for reduction of data size using standard DivX-codecs.

2.9. Software implementation

The analysis software was implemented using an object-oriented approach in Matlab (MathWorks) along with mex-functions (C code compiled in Matlab) that handled a few frequently used low-level functions [*the program can be obtained from the authors on request*]. For derivation of equations used in projective geometry see Appendix B.

2.10. Paw model

The rat paw was modeled as a set of 13 elements of which one element represents the palm of the paw and the other 12 represent the phalanges (three for each of the four long digits – the most radial digit, corresponding to the human thumb, was not included in the model due to its particularly reduced size in rodents). Each element was described as an ellipsoid joined to the adjacent elements at defined points, corresponding to anatomical bones and joints. The size and position of the paw elements was manually adjusted to fit a high resolution 3D-image of the paw. To allow for individual differences in paw size, overall scaling of the model paw was performed for each animal (system robustness to choice of parameter values are presented in Appendix C). A few logical constraints based on physiological limitations were introduced to narrow down searches for possible paw poses. For example, the most distal phalanx of the paw is especially small which makes the tracking of movements in the third (distal interphalangeal; DIP) joint of the digit less reliable. This joint was therefore assumed to have a flexion/extension parameter determined by the physiological influence of the second (proximal interphalangeal; PIP) joint; in detail, the DIP-joint was set to have the same angle as the PIP-joint, with a flexion limit of 60 degrees, measured from its fully extended state (Landsmeer, 1963).

After decreasing the degrees of freedom, the included parameters were the following:

(i) Three parameters for wrist position in space
(ii) Three parameters for wrist movements
(iii) One parameter for opposition of the palm due to movement of the metacarpal bones
(iv) One parameter for adduction/abduction for each digit
(v) Two parameters for flexion/extension of the first two joints of each digit (the metacarpophalangeal; MCP-joint, and the PIP-joint).

These parameters add up to a total of 19 degrees of freedom that together define the movements of the model paw. The most likely paw pose in 3D space could then be estimated by adjusting the orientation of the ellipsoid elements around their respective movement axes and projecting the composite pose onto calculated image planes corresponding to the camera and mirror positions (Fig. 1).

2.11. Paw pose estimation

For each multiframe, an iterative optimization procedure was employed to estimate the paw pose in terms of the parameter set described in Section 2.10. In order to find the initial set of parameters, we employed a database containing the parameter sets corresponding to the most commonly encountered poses (see Section 2.11.3). The search algorithm then generated a number of hypothetical poses and chose the one that best matched the captured video in each iterative step. The following section describes how matching quality was measured and Section 2.11.2 how hypothetical poses were generated in the search algorithm.

2.11.1. Matching quality

To measure the matching quality of a given pose with respect to the video images we used a combination of two matching quality measures – first, the silhouette matching quality, measuring how well the video image silhouette is explained by the given pose, and second, the edge matching quality, measuring how well the video image edges are explained by the pose (Oikonomidis et al., 2011).

Prior to each reaching session, an image of the static background was captured. This image was then used to detect foreground objects by so called background subtraction. Pixels deviating significantly from the background image are likely to belong to foreground objects such as the forelimb, the food pellet or the snout. Hence, pixels with a deviation over a predefined threshold are classified as foreground. This information was stored in a binary silhouette image $\boldsymbol{S}$, where $S_{ij} = 1$ means foreground and $S_{ij} = 0$ means background for the pixel at (i, j).

If we then consider a given pose, defined by the values for all the 19 parameters denoted in Section 2.10, and project the 3D model of the pose onto a given image plane, we obtain a predicted silhouette image $\hat{S}$. If the model parameters are correct, matrices S and $\hat{S}$ should be similar. To quantify this we used the Jaccard index, being the number of pixels that are equal to 1 in both of the matrices S and $\hat{S}$, divided by the number of pixels equal to 1 in at least one of the matrices $\boldsymbol{S}$ or $\hat{S}$. Thus, the silhouette matching quality was defined as

$$q_S = \frac{\sum_{i,j} S_{ij}\hat{S}_{ij}}{\sum_{i,j} S_{ij} \| \hat{S}_{ij}} \tag{1}$$

where || is the logical OR-operator.

To measure the edge matching quality, an edge image $\boldsymbol{E}$ was computed from $\boldsymbol{S}$ in the following way. First, the 8-neighborhood (horizontal, vertical and diagonal) of each foreground pixel is analyzed. If at least one of the pixels in the neighborhood is background, the pixel is considered to be an edge pixel. The edge orientation in that point is approximated as the mean angle from the center pixel to all background pixels in the neighborhood. Depending on this angle, the corresponding element in the edge image $\boldsymbol{E}$ is set to either -1 (for angles from 0 to π) or +1 (for angles from π to 0). If there are no background pixels in the neighborhood the corresponding element is set to 0. Finally, the measured edge image $\boldsymbol{E}$ is filtered with a Gaussian filter to increase robustness.

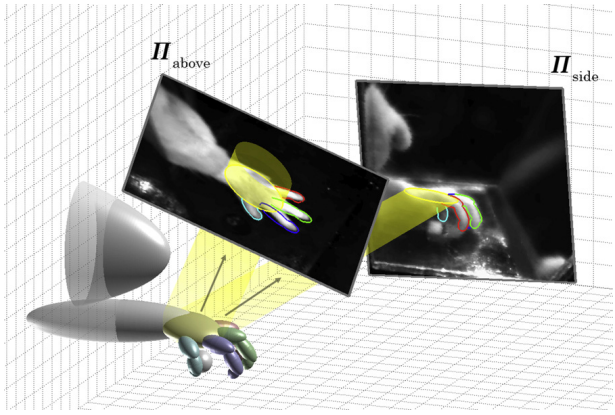


Fig. 1. Computational principle for estimation of paw poses. The pose of a 3D forepaw model is inferred by fitting of the calculated 2D projections onto the different image planes. An estimated 3D paw pose and two of the six image planes (Π_{above} and Π_{side}) are shown. Colored silhouettes outlined on the video images denote the calculated projections.

Analogously, a predicted edge image $\hat{E}$ was constructed from $\hat{S}$, but without the Gaussian filtering. Edge matching quality was then defined as:

$$q_E = \frac{\sqrt{\max\left\{0, \sum_{i,j} E_{ij}\hat{E}_{ij}\right\}}}{\sum_{i,j}|\hat{E}_{ij}|} \quad (2)$$

Both q_S and q_E yield values between 0 and 1, where 1 corresponds to a perfect match. The combined matching quality for each view was then obtained as

$$q = q_S q_E \quad (3)$$

Finally, the total matching quality was defined as the average of the combined matching quality over all viewing planes.

2.11.2. Generating hypotheses

The 19 parameters describing paw poses were divided into five subsets which were processed consecutively, as an exhaustive search over all combinations of the 19 parameters would be computationally unfeasible. At each step of the search algorithm, optimization was performed over one of these subsets. To illustrate how this optimization is carried out we here consider two of the subsets. The others follow the same principles.

2.11.2.1. Search algorithm I – flexion/extension optimization. The algorithm searches for the best combination of flexion/extension parameter values for a pose with their current values as the starting point.

1. Create n_1 hypotheses for the flexion at the first joint (MCP) of digit 2 in an interval centered at the current flexion value.
2. Similarly, create n_2 hypotheses for the second joint (PIP).
3. For each of the $n_1 n_2$ pairs of hypotheses, compute the fraction of the major axes of the three ellipsoids that overlaps with areas classified as foreground when projected onto all the different camera image planes. If below some threshold, discard the hypothesis.
4. For the remaining hypotheses, generate silhouette and edge images for digit 2 from the 3D model and evaluate how much of the silhouette and edges that correspond to the observed counterparts. Keep the m best hypotheses.
5. Repeat steps 1–4 for digits 3, 4, and 5.
6. For all m^4 combinations of flexion parameters, evaluate the total matching quality and assume the best combination as the result of optimization.

2.11.2.2. Search algorithm II – wrist position optimization. The algorithm searches for the best position of the wrist with the current position as the starting point.

1. Sample n values for each of the three spatial dimensions in intervals centered on current position values. The possible combinations of these values lead to n^3 hypothetical poses.
2. For each of the n^3 hypothetical poses, compute the fraction of the major axes of the ellipsoids of the digits that overlaps with areas classified as foreground when projected onto all the different camera image planes. If below some threshold, discard the hypothesis.
3. For all remaining hypotheses, evaluate the total matching quality and assume the best combination as the result of the optimization.

Optimization was iterated over the five subspaces until no further improvement of matching quality was found for any of the different search spaces. Control experiment confirmed that differences in matching quality of final pose estimates depending on the specific search order used were small (Appendix C).

2.11.3. Pose estimation initialization

The database employed to initialize the optimization procedure for paw pose estimation contained the parameter sets corresponding to the most commonly encountered paw poses. In addition, poses that had previously proven difficult to estimate were also included (such poses were identified by poor matching quality in earlier tracking results and were semi-automatically re-estimated and manually verified to have high matching quality before being added to the database). As a starting point, each search was initialized through selection of a set of parameters from the database based on the optimal parameter set found in the previous step. The

entire database was used if no such parameter set existed. Heuristic functions were employed to decrease the number of parameter sets to a more limited number and the set with best matching quality according to Eq. (3) was chosen for initialization. This procedure contributed to an increase in robustness and speed of the subsequent search algorithms.

2.12. Analysis of single unit activity

Spiking activity in each channel of the extracellular recordings was separated into single unit (SU) or multi unit (MU) activity through manual spike feature based sorting techniques (Offline Sorter, Plexon Inc.). To isolate action potentials from a single unit, waveforms plotted in principal component feature space (PC1–PC3) were required to form a cluster well separated from both noise and other units. A refractory period of 1.6 ms was assumed and was used as a control criterion for isolated SUs. SUs encountered in the same channel across days were assumed to be the same if they had similar waveform and firing dynamics. The video tracking gave us an extensive parametrical description of the whole reaching and grasping movement, allowing flexibility in how to align and compare different reaching trials. We chose to use the time of maximal forelimb extension as our temporal reference since that permitted us to include and compare also those attempts that occurred after a failed first attempt. This was important for the subsequent neurophysiological analysis, where a main goal was to study how variations in the reaching and grasping movements were reflected in neural activity, and these higher-order attempts added substantial variation to our data set. Once the times of maximal extension were determined standardized firing rates were estimated by convoluting the spike trains with a Gaussian kernel ($\sigma = 30$ ms, step size = 10 ms, "*psth*" function in the Chronux toolbox, Mitra and Bokil, 2008) and normalizing them to the standard deviation of the baseline (-1500 ms to -500 ms). Paired, two-sided Wilcoxon signed rank tests were used to test for significant deviations from baseline activity ($p < 0.05$, "*signrank*" function in Matlab 2010b).

3. Results

3.1. Acquisition of experimental data

Behavioral and neuronal data were acquired in eight sessions on different days, where each session continued as long as the rat showed interest for the task. Each of the eight sessions contained on average 52 ± 37 (mean $\pm$ SD) reaching trials, which in unsuccessful trials often included several additional reach attempts.

3.2. Reliability of pose estimation

It was initially confirmed through visual inspection of a large number of reaching sequences that the calculated paw poses corresponded very well to the subjective estimates of the poses as judged by the good fitting of the pose projection onto the video images from the different cameras. A typical image sequence from two camera views and the projection of the calculated pose of the 3D model onto the corresponding image planes are shown in Fig. 2A. Obtaining a more quantitative measure of tracking performance is however a more challenging task since it essentially requires *ground truth* information on the exact paw pose; that is, direct measurements on the physical paw (Erol et al., 2007; Ho et al., 2011) which is not possible in freely moving animals. We therefore instead evaluated the performance of the system quantitatively using a paraformaldehyde fixated rat forepaw with static joint angles that was manually positioned in many different spatial locations typical of normal reaching movements in the set-up. The 'true' joint angles of the fixated paw were determined in a separate procedure from a large number of photos taken from different angles that were used to create a visual hull of the paw surface to which the 3D model could be manually fitted with high precision (Fig. 2B and C).

The joint angles of the pose estimates generated by the system for the fixated paw (one for each paw positioning) was shown to deviate on average only $2.4 \pm 2.4\%$ of the full movement range for each joint compared to the reference pose obtained by manual fitting of the 3D model to the visual hull (in Cartesian coordinates this error corresponds to a spatial shift of the joint positions of 0.29 ± 0.22 mm [being in the same range as the camera resolution for an individual image plane: $\sim$5 pixels/mm]). In the histogram in Fig. 2D, the mean error of all the estimated poses compared to the reference pose are summarized and an example of a relatively poor estimate (worse than 95% of the estimates) is shown to illustrate the degree of resemblance in this situation (Fig. 2E). Thus, it appeared that at least in this simplified testing paradigm the system provided accurate pose estimates for all the positions tested. It was also confirmed that the matching scores generated by the quality function used to assess the goodness of pose fitting (Section 2.11.1) had a strong inverse correlation with the calculated distance from the reference pose [Pearson's correlation coefficient: $\rho = -0.85$, $p \ll 0.001$].

Furthermore, computer simulations were employed to evaluate the robustness of the search algorithm to variations of free parameter values (such as foreground/background thresholds or size scaling of the model paw) and sensitivity to input noise (for example light reflections or variations in illumination). Such noise will typically result in segmentation errors. In the simulations performed the system overall displayed a stable performance in situations resembling actual recording conditions and using parameter values within the normally used range. Outside a given interval, system performance would then gradually deteriorate. These data are summarized in Appendix C.

3.3. Identification of motor components

In most types of natural behavior several, partially overlapping, movement components are combined into compound motor actions. In previous studies using manual classification of movements, a limited number of heuristically defined motor components have been proposed to make up the reaching and grasping movements of rats performing this task (Gholamrezaei and Whishaw, 2009; Whishaw and Pellis, 1990). As a starting point for identification of motor components we therefore divided a subset of recorded reaching trials into movement sequences incorporating four of the previously suggested phases: (1) *Advancement*, (2) *Arpeggio*, (3) *Grasping* and (4) *Retraction*. Due to variations in reaching speed, the number of multiframes in each phase often varied slightly between different trials. However, the start and end points of each phase were generally easily identifiable. For all the 19 degrees of freedom, the parameter values obtained from the multiframe corresponding to the start point of each phase were subtracted from that corresponding to the end point. Averaging these parameter differences thus generated four 19-dimensional movement vectors representing the joint movements of each phase. By considering Retraction as backward Advancement, the Retraction vector could be replaced by the negative Advancement vector. The remaining three vectors were subsequently used as base vectors of the motor components, spanning a 3D motor component space. To eliminate tracking noise and to further separate the different motor components, any non-zero vector element corresponding to movement in a joint that was judged not to be part of a given motor component was set to zero – for example wrist movements were not considered to be part of grasping. Examples

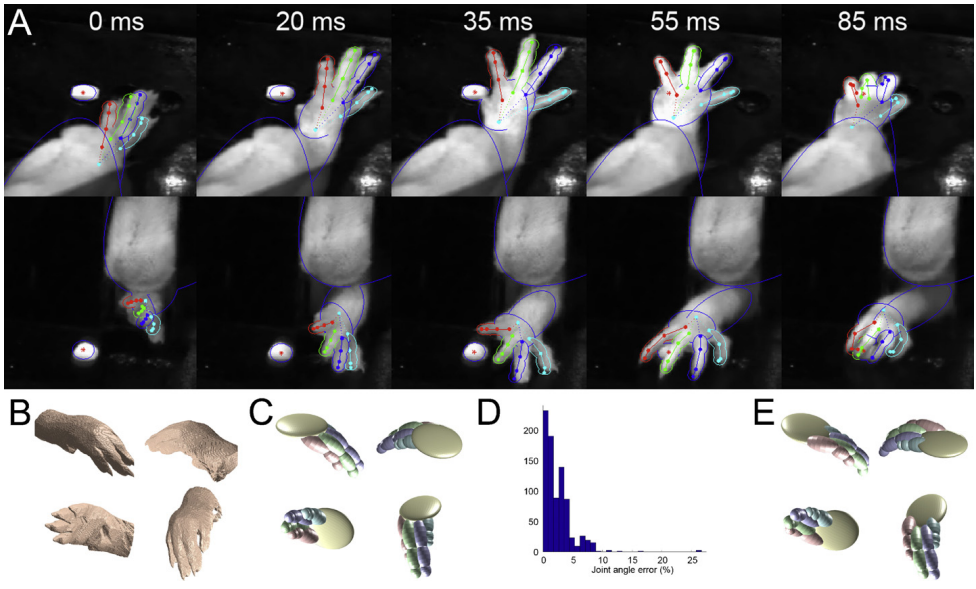


Fig. 2. Tracking of skilled reaching movement sequences. (A) The tracking of the paw in a typical reaching and grasping movement for two of the six viewing planes. The silhouette outline of the estimated paw pose is superimposed on the images for illustration of tracking performance (colored dots/lines indicate joints/bones of the model). (B) Visual hull of the fixated paw shown from four different viewing angles. (C) Reference pose obtained by manual fitting of the 3D model to the visual hull. (D) Histogram of differences in joint angles between estimates provided by the system and the manually fitted reference pose (errors are denoted in percentage of total movement range; $n = 76$ poses). (E) Example of a less accurate pose estimate provided by the system (mean angle error = ~3.3%). Note that for 95% of the estimates the mean error is less than 3.3% and that even for less accurate pose estimates great resemblances to the manually fitted reference pose (C) is evident.

of calculated motor components are shown in Fig. 3. Note that even though each motor component was extracted from movements of a certain part of the compound action sequence, they were often flexibly combined with other motor components during the reaching and grasping movement.

3.4. Reaching movements described in motor component space

By reducing the 19 degrees of freedom to this much smaller set of motor components, each reach attempt could be relatively well described as a trajectory in 3D motor component space. The reduction was achieved through a least-squares approximation of each 19-dimensional vector to a linear combination of the three base vectors, which results in a unique point on the trajectory in the 3D motor component space. Interestingly, in spite of the reduced dimensionality this compact 3D representation of the compound movements proved sufficient to separate different reaching and grasping patterns in most reaching trials. In fact, in many reaching trials the three motor components were executed in close to sequential order giving rise to line segments of movement trajectories running almost parallel with the axes in 3D motor component space. Ten individual reaching trials recorded during a single session are shown in Fig. 4. Note that although no single trial is identical to the others, trajectories from successful trials (green) are spatially separated from failures (red) in this representation (shadow images on the sides of the box denote one standard deviation from the mean trajectory of each of the two groups).

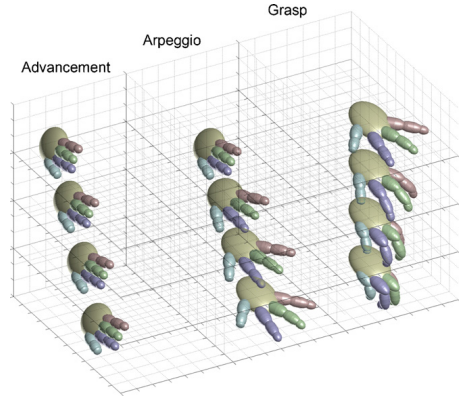


Fig. 3. Decomposition of skilled reaching into motor components. Examples of heuristically defined motor components of compound reaching and grasping movements. The Advancement motor component (left) captures translation along the axis ranging from the aperture of the box to the food pellet. Arpeggio (middle) captures the pronation of the forelimb and adduction of the digits preceding grasping and Grasp (right) is associated with changes in the angles of the joints of the paw and digits.

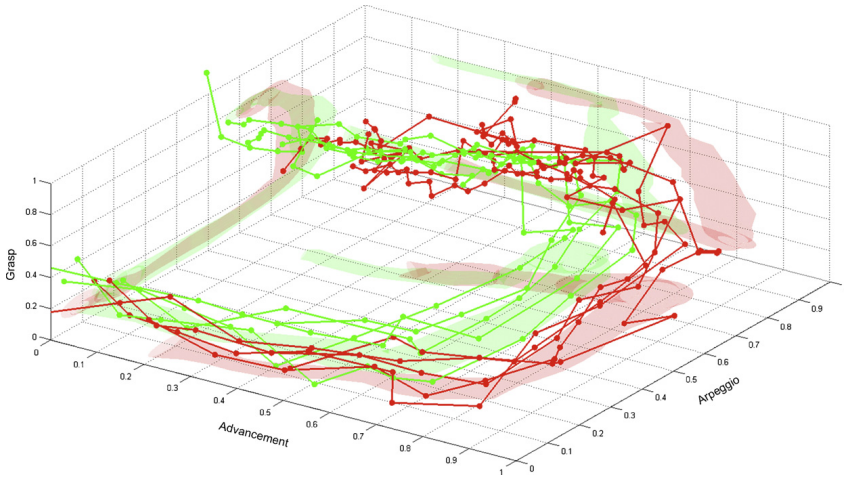


Fig. 4. Movement trajectories in motor component space. Example of reaching trials represented in motor component space. Five individual successful reaching trials (green) are shown together with five unsuccessful trials (red). Dots indicate the time points of paw pose estimations (5 ms intervals) and colored shadows on box walls indicate one SD from the mean trajectory for each group. For ease of comparison, the plot only contains first reach attempts.

3.5. Modulation of neuronal firing correlated with actuation of motor components

The neural recordings included single- and multiunit activity from the primary motor cortex and dorsolateral striatum. As expected, several units displayed a wide range of different task related firing rate modulations. This included phasic increase in activity before, during or after the task, but also partial or total suppression of activity during the task. Using a comparatively long time window of −500 ms to 200 ms (relative to the time of maximal forelimb extension) when comparing firing rates to baseline (−1500 ms to −500 ms) we found that the overall fraction of task related SU neurons was 35% (29/83). The corresponding number for MUs was 13% (5/38). To search for neuronal modulation specifically linked to the actuation of the identified motor components, the reaching attempts were sorted with respect to the maximum value of each component in each attempt and data were divided into three equally sized groups (low/middle/high). That meant that each reach attempt was assigned to either low, middle or high three times (once for each component) so that a specific reach attempt could be, for example, low in the advancement, middle in the arpeggio and high in the grasp component. Group dependent rate modulations were subsequently tested for by comparing differences between the medians of the three groups (Kruskal–Wallis test, 2 d.f.) in 10 ms time steps (see Section 2.12 for details). Using the distinction that a cell was considered significantly modulated in relation to a given motor component if the null hypothesis that the medians were identical could be rejected at the $p = 0.05$ level in at least eight consecutive time steps, it was found that 11%, 13% and 10% of the SUs were significantly modulated in relation to Advancement, Arpeggio and Grasp, respectively. For MUs the corresponding numbers were 21%, 5% and 21%. Examples of three cells [SUs recorded in striatum (top/middle) and motor cortex (bottom)] showing specific modulations in relation to actuation of the three different motor components are presented in Fig. 5. Thus, the system proved to allow for the correlation of neuronal recordings to kinematic data representing sub-components of compound reaching movement thereby opening up for future studies on how the central nervous system encodes motor commands.

4. Discussion

To perform detailed motion tracking of movements around multiple joints in freely moving animals using traditional semi-manual movement tracking is a daunting task requiring many hours of analyses by well-trained observers (Hermer-Vazquez et al., 2004; Hyland and Jordan, 1997; Whishaw and Pellis, 1990). For some motion tracking applications, fully automatic systems have been developed to facilitate this procedure, typically by the use of reflective markers on concerned body parts (Peikon et al., 2009; Zakotnik et al., 2004). However, in the case of hand pose estimation in humans, improved motion tracking algorithms not requiring body markers have been developed recently (Erol et al., 2007). To track fine movements in subjects not suitable for tracking with markers such as freely behaving rodents, this latter solution is clearly preferable. Consequently, a primary objective of the current study was to develop a marker-less motion tracking system and evaluate the system using reaching and grasping movements performed by rats. A further requirement was that the system must provide kinematic data on forepaw movements in this behavioral task at a level of detail that allow us to study minor differences between reaching trials to correlate this variability to parallel neuronal recordings in motor control circuits.

In the data presented here, it is shown that high-resolution tracking of fine movements involving multiple joints can be performed automatically by the system from recorded video sequences of a rat grasping food pellets, with reliable outcome. Although the accuracy of the pose estimates generated by the system could not be directly evaluated in the real reaching situation the control experiments on a fixated paw indicated that mean errors were typically only a few tenths of a millimeter. This precision was judged to be sufficient for the intended purposes and compares well to systems designed for human hand tracking (Oikonomidis et al., 2010; Nirei et al., 1996). The system also proved to be relatively robust to

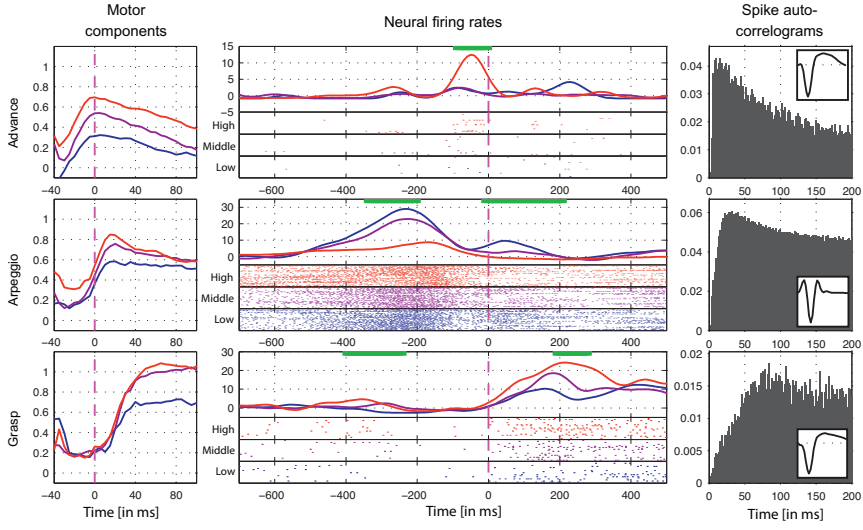


Fig. 5. Neural correlation to motor components. Examples of neurons correlated to individual motor components. Reach attempts are sorted depending on the peak value for a motor component during the attempt, as shown in the left column. The top row is sorted for advance so that the shortest extensions are pooled into the blue group, the intermediate extensions into the purple group and the longest into the red group. In the same manner, the middle row is sorted for arpeggio and the bottom row is sorted for grasp. The middle column show standardized peri-event firing rates and raster plots of individual trials for each group, aligned to time point of maximum paw extension and averaged over all attempts in each group. A green line indicates a significant difference in firing rates between any of the three groups ($p < 0.05$). The right column shows the autocorrelogram and waveform for each presented neuron.

addition of artificial noise or manipulation of free parameters over a wide range. Thus, under the variations of normal experimental conditions the pose estimates provided by the system appears to be reliable enough for the desired applications. It is also illustrated how this more detailed kinematic description can be used as a basis for new analytical approaches in neurophysiological investigation of motor signals. In fact, several examples of specific firing rate modulations in single cells relating to the actuation of the individual motor components were found. Although a more comprehensive analysis of motor coding in corticostriatal circuits would require a significantly larger data set the current results nevertheless clearly indicate that the obtained kinematic data opens up for very detailed neurophysiological studies of motor control circuits. These findings are encouraging since the demand for detailed tracking of movement kinematics with high temporal precision is steadily increasing in neurophysiological research to permit interpretation of the rapidly growing neuronal data sets of freely behaving animals (Hoffman and McNaughton, 2002; Kruger et al., 2010; Nicolelis, 2001). Hence, automated methods like the one described in this paper could potentially become an important tool in basic and applied neuroscience. However the described method could likely also improve the quality of behavioral experiments in many other fields and partly reduce the need for the elaborate and time consuming process of manual scoring that, in addition, often requires extensive training to ensure reliable and reproducible identification of movement patterns (Cenci and Lundblad, 2007; Eshkol, 1958; Whishaw and Pellis, 1990; von Laban, 1980).

In the current study previously identified motor components (advancement/retraction, arpeggio and grasping) was used as a starting point for analysis. Nevertheless, the decomposition of any movement into motor components can naturally be done in many different ways once the detailed movements over individual joints have been extracted by the system. This flexibility in the choice of bases functions and spatial coordinate system for the description of movements may prove particularly important in investigations of supraspinal motor control systems given the ongoing controversy regarding the primary coding strategy used by these systems (see for example; Georgopoulos et al., 1986; Loeb et al., 1996; Todorov and Jordan, 2002).

For experiments requiring an even higher level of detail in motion tracking a few improvements to the current system could be made. In our current model design, the distal interphalangeal joint of the digits is assumed to have the same angle as the proximal interphalangeal joint due to the small size of the 3rd phalanx (up to a maximum distal joint flexion angle of 60 degrees). While this is a reasonable approximation when considering the anatomical constraints of the muscles and tendons of the digits (Landsmeer, 1963) and was deemed sufficient for the current purposes, it may be desirable in other applications to have independent tracking of movements around this joint. Similarly, additional parameters such as separate metacarpal bones, a shoulder and an elbow joint could be beneficial to add to the model. Indeed, in a recently published study using X-ray video techniques to track skeletal bones during reaching movements in rats, complex coordination of the proximal parts of the limb and trunk were found (Alaverdashvili et al., 2008). In relation to our data it is worth noting that the X-ray images highly resemble the projections of our 3D paw model onto these viewing planes, although in our study images of the paw were also obtained from an additional viewing plane. Finally, as neural data is sampled in magnitudes of tens of kilohertz it would in some experiments be

appropriate to have a comparable rate of acquisition of images for the behavioral tracking. In the current set-up an image acquisition rate of 200 Hz appears to meet basic criteria, but a higher acquisition rate would likely remove some of the tracking noise since consecutive poses would be more alike.

As previously pointed out, the reaching and grasping behavior in rodents bears many similarities to the corresponding behavior in primates and other larger animals indicating that this model lends itself well to translational research, although the very rapid execution of movements in small animals obviously constitutes an extra challenge. This difference between primate and rodent behavior must also be taken into account when analyzing neuronal correlates of different movement components since the neuronal activity related to specific sub-components of the movement as well as the potential use of sensory feedback for corrections of ongoing movements is more limited in the latter group (typical minimum latencies of tactile cortically evoked potentials in rat/man are ~10/20 ms whereas typical reaching times in this task are ~250/500 ms, see; Allison and Hume, 1981; Sacrey et al., 2009). It is also worth pointing out that while rats primarily depend on odor cues to locate the position of the food pellets humans rely on visual guidance in a similar reaching task when the experimental conditions allow for it (de Bruin et al., 2008; Whishaw and Tomie, 1989). However, in spite of the overall slower actuation of reaching and grasping movements in humans, many of the individual motor components involved are undoubtedly executed very rapidly – thus the automated image analysis approach developed here may also open up for new studies of reaching in humans. The technique could potentially provide a new tool for more detailed movement tracking both in the lab (for example in studies of motor learning and control) and in the clinic – diagnosing diseases which affect the motor system at an early time point, giving guidance in selection of optimal treatment strategies (Doan et al., 2010; Jenkins et al., 2010) or helping to assess the effectiveness of rehabilitation programs after for example trauma or ischemic events affecting motor systems (Massie et al., 2009; McCrea et al., 2005).

Acknowledgments

Grants: The Neuronano Research Center is supported by grants from the Swedish Research Council/Linneus grant (project#: 60012701, 80658701) and from the Knut and Alice Wallenberg Foundation (project#: KAW 2004.0119). This project was also sponsored by grants from the Olle Engkvist, Jeanssons, Magnus Bergvall, Kock, and Segerfalk Foundation and a project grant from the Swedish Research Council (project#: K2010-62X-21400-01-3).

The authors are thankful to Lars Clementz for building the experimental set-up and to Dr. Ulrike Richter for thoughtful comments on earlier versions of the manuscript.

Appendix A. Training protocol

1. Every day for a few days prior to the experiments, the rats were accustomed to the food reward after 20–24 h food deprivation. After each training session the animal received free supply of their regular food for 1 h.
2. The teaching of the reaching behavior began by making the rats aware that there will be food placed on the shelf. This was done by placing multiple pellets on the shelf nearby the aperture for the rats to reach for in any way they can.
3. The acquisition of food was subsequently made harder by moving the pellets further away from the opening slit, forcing the rats to use their forepaws to reach for the pellets.
4. As the rats developed a preference of paw, only one pellet was placed in the socket contralateral to their paw of choice to further promote the use of this paw.
5. In the next training step the rat was taught to move to the back of the cage after each reaching trial, this was achieved by simply not placing any food rewards on the shelf until the rat had reached the opposite end of the cage. This was needed to properly separate different trials.
6. To verify that the displayed behavior was indeed goal directed reaching for food; no food pellet was placed on the shelf at certain occasions. If the rat made a reach attempt anyhow, no pellet was placed on the shelf the consecutive trial either, thereby forcing the rat to identify the presence of food before attempting to reach for it.

Appendix B. Projective geometry

B.1. The camera model

A camera matrix describes the transformation of 3D points to 2D points in a pinhole camera: $\boldsymbol{X} \rightarrow \boldsymbol{PX}$, where $\boldsymbol{P}$ is a 3×4 camera matrix and $\boldsymbol{X} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$ a 3D point represented in homogeneous coordinates. This transformation is expressed in the *camera equation*,

$$\lambda \boldsymbol{x} = P\boldsymbol{X}, \tag{B.1}$$

where $\boldsymbol{x} = \begin{bmatrix} X & Y & 1 \end{bmatrix}^T$ is the 2D image point $\begin{bmatrix} x & y \end{bmatrix}^T$ represented in homogeneous coordinates, and $\lambda > 0$ is the *depth*, i.e., the distance from the 3D object point $\boldsymbol{X}$ to the image plane (Hartley and Zisserman, 2000). The estimation of the camera matrix $\boldsymbol{P}$ is performed using Direct Linear Transformation (DLT) (Abdel-Aziz and Karara, 1971). Using an object whose 3D coordinates are known and manually entering the image coordinates of some points of the object gives the user a set of 3D coordinates and their corresponding image coordinates which is used for DLT.

B.2. Quadric surfaces

A quadric surface is a surface in a 3D space defined by a second order implicit equation,

$$\boldsymbol{X}^T \boldsymbol{Q} \boldsymbol{X} = 0$$

where $\boldsymbol{X} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$ is a homogenous coordinate-vector and $\boldsymbol{Q}$ a 4×4 symmetric matrix. Quadrics can be used to describe a number of geometrical shapes, where the ellipsoid and the elliptical paraboloid are the ones used in the present study. In detail, each of the phalanges, the palm of the paw, the forearm, and the food pellet are modeled as ellipsoids, while the snout is modeled as an elliptical paraboloid. Through defining all geometrical shapes of the tracked objects as quadrics, analytical solutions and efficient implementations are possible for the majority of the employed mathematical operations. For instance, the projection of an ellipsoid onto an image plane as well as the silhouette outline of the projection can be computed analytically, and the quadrics $\mathbf{Q}_0$ and $\mathbf{Q}_1$ occupying the same space can be detected by analyzing the eigenvalues of $\mathbf{Q}_0^{-1}\mathbf{Q}_1$. This is described further by (Stenger, 2001).

Appendix C. Robustness to choice of free parameter values and measurement noise

C.1. Assessment of algorithm robustness to variation in free parameter values

In the first stage of the algorithm the paw is separated from the background based on the light intensity in individual pixels

Table C.1
Robustness to light intensity thresholding.

Threshold value	Accuracy mean error (%)	Precision SD of errors (%)
0.10	2.33	0.76
0.20	2.40	1.25
0.30	2.05	0.70
0.40	2.01	0.61
0.50	2.27	0.85
0.60	2.49	0.94
0.70	2.44	0.97

(Section 2.11.1). Proper foreground/background segmentation is therefore sensitive to choice of threshold value. In (Table C.1) the precision (SD of the distribution of pose estimate errors) and accuracy (mean pose estimate error) of pose estimates compared to the fixated reference pose (Fig. 2D) are presented for a range of threshold values (10–70% of dynamic range) showing a relatively stable performance. The error in each joint angle is normalized to the observed full range of motion for that joint angle and the average value over all joint angles is presented.

Because the size of the image of the paw may vary slightly depending on interindividual differences or changes in the camera set-up, the paw model is manually scaled to the video images for each animal. In (Table C.2) the precision and accuracy of pose estimates compared to the fixated reference pose are similarly presented for a range of scaling values (80–120% of actual value). Note the comparatively stable performance for sizes deviating <20% from original size.

As described in (Section 2.11.2) the search algorithm was based on an iterative process consecutively searching the five parameter subspaces. Because methods based on stepwise local optimization run the risk of entering local minima we evaluated to what extent different search orders generated different final pose estimates. The video sequence of a reaching trial containing 25 multiframes was searched iteratively using ten different randomly chosen orderings of the five subspaces. Comparing the variation in final pose estimate accuracy for the different search orders showed that choice of search order had a measurable but in practice negligible effect on final pose estimate quality [SD was on average only ±0.13% of full joint movement range (corresponding to ±0.01 mm) which should be compared to the mean accuracy of 2.94% (or 0.31 mm) for this data set].

C.2. Assessment of algorithm robustness to measurement noise

Measurement noise, such as fluctuations in illumination conditions, focusing errors, light reflexes and shadows will ultimately result in errors in paw segmentation (Section 2.11.1). Hence, to allow for a quantitative evaluation of the robustness of the pose estimates to acquisition noise, computer simulations were performed where errors in segmentation were introduced in a controlled manner through addition of artificial noise to paw images. Synthetic silhouette images were first generated from the geometric paw model by projection onto the image planes and segmentation errors were then modeled by distortion of these silhouette images. To simulate segmentation errors a number of foreground pixels were chosen at random and all foreground pixels at a distance varying from 5 to 10 pixels from these points were labeled as background. Similarly, areas around a number of randomly chosen background pixels were changed to foreground pixels. The effect on pose estimation quality as a function of degree of distortion of the self-generated images is shown in Table C.3. The segmentation error is expressed as the number of erroneously labeled pixels divided by the number of foreground pixels in the original image. The error in each pose parameter is normalized to the observed full range of motion for that parameter and the average value over all parameters is presented.

Table C.2
Robustness to paw size variation.

Scaling factor	Accuracy mean error (%)	Precision SD of errors (%)
0.80	2.83	0.91
0.85	2.51	0.87
0.90	2.28	0.67
0.95	2.34	0.67
1.00	2.44	1.18
1.05	2.29	0.75
1.10	2.52	1.03
1.15	2.35	1.03
1.20	2.85	1.77

Table C.3
Robustness to image distortion.

Noise level (%)	Accuracy mean error (%)	Precision SD of errors (%)
0–5	4.9	0.51
5–10	5.3	0.72
10–15	5.9	0.81
15–20	5.9	1.50
20–25	6.2	1.22
25–30	6.7	1.87
30–35	7.2	0.62
35–40	7.6	1.16
40–45	10.1	3.25

References

Abdel-Aziz YI, Karara HM. Direct linear transformation into object space coordinates in close-range photogrammetry. Proc Symp Close-Range Photogrammetry 1971:1–18.

Alaverdashvili M, Leblond H, Rossignol S, Whishaw IQ. Cineradiographic (video X-ray) analysis of skilled reaching in a single pellet reaching task provides insight into relative contribution of body, head, oral, and forelimb movement in rats. Behav Brain Res 2008;192:232–47.

Allison T, Hume AL. A comparative analysis of short-latency somatosensory evoked potentials in man, monkey, cat, and rat. Exp Neurol 1981;72:592–611.

Cenci MA, Lundblad M. Ratings of L-DOPA-induced dyskinesia in the unilateral 6-OHDA lesion model of Parkinson's disease in rats and mice. Curr Protoc Neurosci 2007 [Chapter 9: Unit 9 25].

de Bruin N, Sacrey LA, Brown LA, Doan J, Whishaw IQ. Visual guidance for hand advance but not hand withdrawal in a reach-to-eat task in adult humans: reaching is a composite movement. J Mot Behav 2008;40:337–46.

Doan JB, Whishaw IQ, Pellis SM, Suchowersky O, de Bruin N, Brown LA. Challenging context affects standing reach kinematics among Parkinson's disease patients. Behav Brain Res 2010;214:135–41.

Donoghue JP, Wise SP. The motor cortex of the rat: cytoarchitecture and microstimulation mapping. J Comp Neurol 1982;212:76–88.

Erol A, Bebis G, Nicolescu M, Boyle RD, Twombly X. Vision-based hand pose estimation: a review. Comput Vis Image Underst 2007;108:52–73.

Eshkol N, Wachman A. Movement notation. London: Weidenfeld and Nicolson; 1958.

Gentilucci M, Toni I, Daprati E, Gangitano M. Tactile input of the hand and the control of reaching to grasp movements. Exp Brain Res 1997;114:130–7.

Georgopoulos AP, Schwartz AB, Kettner RE. Neuronal population coding of movement direction. Science 1986;233:1416–9.

Gholamrezaei G, Whishaw IQ. Individual differences in skilled reaching for food related to increased number of gestures: evidence for goal and habit learning of skilled reaching. Behav Neurosci 2009;123:863–74.

Hartley R, Zisserman A. Multiple view geometry. Cambridge University Press; 2000.

Hermer-Vazquez L, Hermer-Vazquez R, Moxon KA, Kuo KH, Viau V, Zhan Y, Chapin JK. Distinct temporal activity patterns in the rat M1 and red nucleus during skilled versus unskilled limb movement. Behav Brain Res 2004;150:93–107.

Hermer-Vazquez L, Moshtagh N. Rats' learning of a new motor skill: insight into the evolution of motor sequence learning. Behav Processes 2009;81:50–9.

Ho M-F, Tseng C-Y, Lien C-C, Huang C-L. A multi-view vision-based hand motion capturing system. Pattern Recogn 2011;44:443–53.

Hoffman KL, McNaughton BL. Coordinated reactivation of distributed memory traces in primate neocortex. Science 2002;297:2070–3.

Hyland BI, Jordan VMB. Muscle activity during forelimb reaching movements in rats. Behav Brain Res 1997;85:175–86.

Jeannerod M. The timing of natural prehension movements. J Mot Behav 1984;16:235–54.

Jenkins ME, Johnson AM, Holmes JD, Stephenson FF, Spaulding SJ. Predictive validity of the UPDRS postural stability score and the Functional Reach Test, when

compared with ecologically valid reaching tasks. Parkinsonism Relat Disord 2010;16:409–11.
Kruger J, Caruana F, Volta RD, Rizzolatti G. Seven years of recording from monkey cortex with a chronically implanted multiple microelectrode. Front Neuroeng 2010;3:6.
Landsmeer JM. The coordination of finger-joint motions. J Bone Joint Surg Am 1963;45:1654–62.
Loeb GE, Brown IE, Scott SH. Directional motor control. Trends Neurosci 1996;19:137–8.
Massie C, Malcolm MP, Greene D, Thaut M. The effects of constraint-induced therapy on kinematic outcomes and compensatory movement patterns: an exploratory study. Arch Phys Med Rehabil 2009;90:571–9.
McCrea PH, Eng JJ, Hodgson AJ. Saturated muscle activation contributes to compensatory reaching strategies after stroke. J Neurophysiol 2005;94:2999–3008.
Mitra P, Bokil H. Observed brain dynamics. New York: Oxford University Press; 2008.
Nicolelis MA. Actions from thoughts. Nature 2001;409:403–7.
Nicolelis MAL. Methods for neural ensemble recordings. CRC 2008.
Nirei K, Saito H, Mochimaru M, Ozawa S. Human hand tracking from binocular image sequences. In: Industrial Electronics, Control, and Instrumentation, 1996, Proc of the 1996 IEEE IECON 22nd Intl Conf, vol. 1; 1996. p. 297–302.
Oikonomidis I, Kyriazis N, Argyros AA. Markerless and Efficient 26-DOF Hand Pose Recovery, in Proceedings of the 10th Asian Conference on Computer Vision, ACCV'2010, Part III, LNCS 6494, 2010, pp. 744–757, Queenstown, New Zealand, Nov. 8–12.
Oikonomidis I, Kyriazis N, Argyros A. Markerless and efficient 26-DOF hand pose recovery. In: Kimmel R, Klette R, Sugimoto A, editors. Comp Vis. – ACCV 2010. Berlin /Heidelberg: Springer; 2011. p. 744–57.
Peikon ID, Fitzsimmons NA, Lebedev MA, Nicolelis MA. Three-dimensional, automated, real-time video system for tracking limb motion in brain-machine interface studies. J Neurosci Methods 2009;180:224–33.
Peterson G. Mechanisms of handedness in the rat. Comp Psychol Monogr 1934;9:1–67.
Sacrey LA, Alaverdashvili M, Whishaw IQ. Similar hand shaping in reaching-for-food (skilled reaching) in rats and humans provides evidence of homology in release, collection, and manipulation movements. Behav Brain Res 2009;204:153–61.
Stenger B.Mendon PRS, Cipolla R, editors. Model-based 3D tracking of an articulated hand; 2001. p. 310.
Todorov E, Jordan MI. Optimal feedback control as a theory of motor coordination. Nat Neurosci 2002;5:1226–35.
West MO, Carelli RM, Pomerantz M, Cohen SM, Gardner JP, Chapin JK, Woodward DJ. A region in the dorsolateral striatum of the rat exhibiting single-unit correlations with specific locomotor limb movements. J Neurophysiol 1990;64:1233–46.
Whishaw IQ, Haun F, Kolb B. Analysis of behavior in laboratory rodents. Modern techniques in neuroscience. Berlin, Germany: Springer; 1999, 1243-1275.
Whishaw IQ, Pellis SM. The structure of skilled forelimb reaching in the rat: a proximally driven movement with a single distal rotatory component. Behav Brain Res 1990;41:49–59.
Whishaw IQ, Tomie JA. Olfaction directs skilled forelimb reaching in the rat. Behav Brain Res 1989;32:11–21.
Whishaw IQ, Whishaw P, Gorny B. The structure of skilled forelimb reaching in the rat: a movement rating scale. J Vis Exp 2008;18, e816, DOI: 10.3791/816.
von Laban R, editor. The mastery of movement. London: Macdonald & Evans; 1980.
Vorhees CV, Acuff-Smith KD, Minck DR, Butcher RE. A method for measuring locomotor behavior in rodents: contrast-sensitive computer-controlled video tracking activity assessment in rats. Neurotoxicol Teratol 1992;14:43–9.
Zakotnik J, Matheson T, Dürr V. A posture optimization algorithm for model-based motion capture of movement sequences. J Neurosci Methods 2004;135:43–54.

Paper III

Automated Tracking of Motor Behavior as a Means to Assess Severity of Symptoms in the 6-OHDA Marmoset Model of Parkinson's Disease

TOBIAS PALMÉR[1,3], MAXWELL SANTANA[2], ROMULO FUENTES[2] AND PER PETERSSON[3]

[1] *Centre for Mathematical Sciences, Lund University*

[2] *Edmond and Lily Safra International Institute of Neuroscience of Natal (ELS-IINN)*

[3] *Integrative Neurophysiology and Neurotechnology, NRC, Department of Experimental Medical Sciences, Lund University, Sweden*

Abstract: We present a method for analyzing locomotive behavior of marmosets in two experimental setups used for the analysis of severity of symptoms in a primate model of Parkinson's disease. A characterization of changes in motor behavior following experimental manipulations based on automatic tracking procedures is here presented.

1 Introduction

To be able to develop new therapies for the treatment of Parkinson's disease (PD) researchers critically depend on valid animal models of the disease that allows for repeated testing of motor disabilities over extended time periods. Motor symptoms in PD typically involve rigidity, tremor and an overall reduction and slowing of movements. However, because of the technical challenges involved in tracking detailed motor behavior in freely moving animals the sensitivity and robustness of the assessment methods used in characterization of motor deficits often turn out to be a key limiting factor in this field of research. To this aim we have developed methods for tracking of freely moving animals and present a first characterization of a range of motor symptoms tested over several months in a primate model of PD-marmoset monkeys (Callithrix Jacchus) that have been exposed to the neurotoxin 6-OHDA causing lesions to dopaminergic cell groups in the midbrain. To validate the developed methods, scores obtained by automatic methods have also been verified using manual scoring protocols based on the rating scales applied by neurologists to evaluate severity of symptoms in PD patients that were here adapted to suit the motor behavior of marmoset monkeys (that is so called UPDRS scores). Importantly, through the use of automated quantitative motion tracking

procedures we are now able to assess the severity of motor deficits in different testing set-ups of spontaneous motor behavior in freely moving animals. In addition, the new screening methods will allow us to investigate more subtle deficits that until now have been hard to evaluate, such as the gradual slowing of certain movements or a reduced ability to combine a sequence of motor components into compound actions.

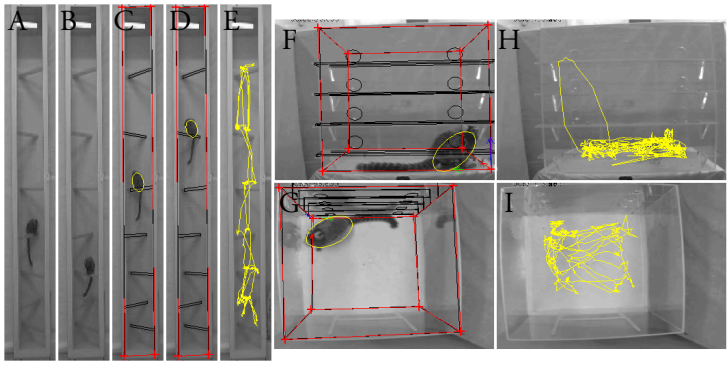


Figure III.1: Video data in grayscale from the tower experiment (A-B). Tracking data for two images (yellow) and tower calibration (red and black) drawn on video data (C-D). Movement trace for the whole experiment drawn on the background image (E). Tracking data (yellow) and box calibration (red and black) drawn on video data for the two tower cameras (F-G). Movement trace for the whole experiment drawn on the background images for the two cameras (H-I).

Analyses of the first set of experiments suggest that: first, the current methodological approach indeed seems to generate reliable tracking data from freely moving primates suitable for long-term analysis of motor behavior. Second, robust motor symptoms lasting for several months could be induced when using a two-stage neurotoxic lesioning procedure involving one hemisphere of the brain at a time, indicating that this non-human primate model of PD may be well suited for long-term evaluation of novel therapies for treatment of PD.

Here we present the methods developed for tracking of spontaneous motor behavior and outline ongoing and future work aimed at obtaining more detailed and reliable characterizations of the kinematics of compound motor acts.

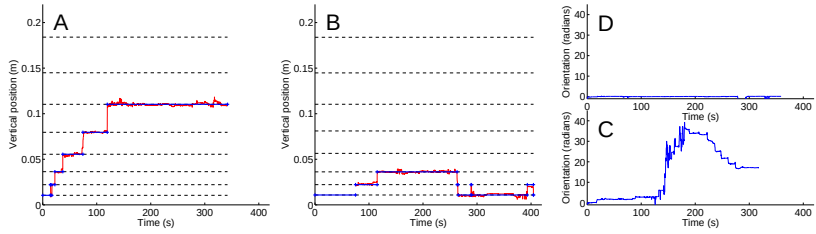


Figure III.2: Characterization of motor behavior in a hemi-parkinsonian animal. Baseline experiments (A,C) and after pharmacological blockade of dopamine synthesis (B,D). In (A,B), the tracked y-coordinate is presented for the tower experiment. In (C,D) the rotation (given in radians) of the subject is plotted over time. Notice that the dopamine depletion affects how high the animal reaches in the tower and how many jumps it makes. In the box experiment the animal turns both left and right in the baseline recording, but after treatment it does not rotate as much and only to the left.

2 Methods

The experiments in this study are comprised of two different experimental setups to be able to observe a range of different behaviors and impairments.

The first setup, the tower experiment (Fig. III.1A-E), is designed to capture the natural locomotive behavior of the common marmoset, including jumps. The setup is a 2.3 meter high tower with 7 bars (levels) that are further separated at the top of the tower. Experiments are filmed by a 640x480 pixel color camera generating video as in Fig. III.1A-E.

The second setup, the box experiment (Fig. III.1F-I), is designed to capture gross aspects of forelimb reaching movements, horizontal locomotion and rotational behavior (which is sometimes seen in lesioned animals). The setup is a transparent box with dimensions 45x45x45 cm. One of the walls has four shelves attached to it, where food rewards can be placed (Fig. III.1F, H). This allows for detailed studies of how test subjects reach out and grasp food objects when retrieving the rewards (for details see [3]). The experiments are filmed by two cameras: one placed above the box, filming straight down (top view) and one filming head on the shelves (front view).

UPDRS scores include 16 different aspects of motor behavior that are each scored from 0 (normal) to 3 (severely parkinsonian). Summed scores are presented in this paper giving a total scale ranging from zero to 84.

2.1 Extraction of kinematic parameters

Motion tracking in the two setups was performed using similar methods. Constant light conditions during each recording sessions as well as a high contrast between the subjects and the background eliminated the need for advanced background models. Hence, a simplified mixture of Gaussians model [4] was employed with two components. That is, each pixel is modeled to belong to either one of two Gaussian distributions. After an initiation procedure, the background is estimated by randomly selecting a number of frames, and then for each pixel in each of the frames decide which component it most likely belongs to. The expectation value of the most probable component is updated using that pixel value. Consequently, an estimate of the video without the subject is contained in the brighter pixels, as the animals in this set of experiments were always darker than the experimental setup. Thus, a binary foreground image can be estimated for each frame in the video by, for each pixel checking whether it is likely to belong to the background component or not. The binary foreground images are used in the shape analysis.

By assuming that the two-dimensional image of the monkey in each camera plane is approximately elliptically shaped, the position and orientation of the animal can be estimated from the position and orientation of the ellipse that best fits the binary foreground images. Given a binary foreground image F and a hypothetical foreground image M, generated by an ellipse (x, y, θ), matching quality is defined as the Jaccard index

$q = \frac{|M \cap F|}{|M \cup F|}$.

The definition means that the quality measure is given by the number of pixels with coinciding foreground classifications divided by the total number of pixels classified as foreground. Notice that the foreground image M is not actually computed, but instead the quality measure is computed using the conic matrix for the ellipse.

Movement tracking in time is then carried out by using the last known location to initiate estimation for a given frame, followed by step-wise improvement of the matching quality by gradual adjustment of the parameters of the estimated ellipse. In practice, a set of hypothetical ellipses are generated randomly, and the matching quality is computed for each of them. The ellipse yielding the highest matching quality is chosen as the result of estimation. Due to the sometimes relatively high movement speed of the marmoset, the empirically set values for maximum change in position is 50 pixels in any direction and change in rotation

is 45 degrees. These calculations are performed for every frame in the video, resulting in the vectors (x, y, θ, q), each of length N, where N is the number of frames in the video and q is the quality measure given above.

Software tools were is developed in MATLAB and includes a few mex implementations (MATLAB compiled c-code).

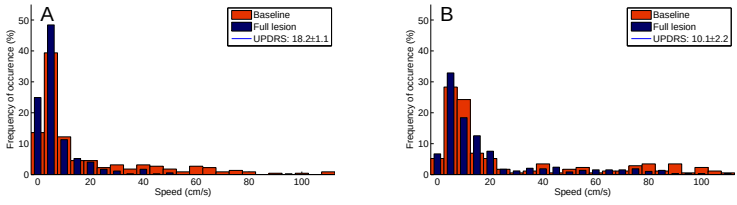


Figure III.3: Evidence of bradykinesia (slowing of movements) in spontaneous motor behavior was observed in two bilaterally lesioned animals. Data was provided by the automatic tracking in the box testing set-up. A threshold (20 cm/s) was used to classify movement bouts into two groups; fast and slow. The group-size quota was modeled as a binary distribution, and a two-proportion z-test of the data showed that there is a 0.001% significant similarity between the distributions for test subject A, and 12% significance in subject B. The number of recording sessions pooled for each animal and condition is denoted over each panel. In the rightmost panels the mean $\pm$ SD of UPDRS scores after lesioning are inserted (as assessed during the same days as the box tests were performed). Note that even though these two animals were only moderately parkinsonian a slowing of movements could still be detected.

3 Results

In the first experiment, a hemi-parkinsonian animal was experimentally dopamine depleted through pharmacological blockade of dopamine synthesis. As expected, the subjects movements are greatly impaired by the drug (Fig. III.2). Firstly, it is unable to reach higher than the second level and secondly, it does not make as many jumps as in the baseline experiment. Two animals with bilateral lesions were then tested in the box experiment over a prolonged period involving multiple recording sessions over many months. In these subjects, a slowing of movements (bradykinesia) could be detected even in moderately parkinsonian animals (Fig. III.3). It should be noted that moderate bradykinesia is particularly challenging to detect using manual qualitative scoring procedures. In the two animals analyzed bradykinesia was however still correlated to the total UPDRS score.

3.1 Tracking evaluation

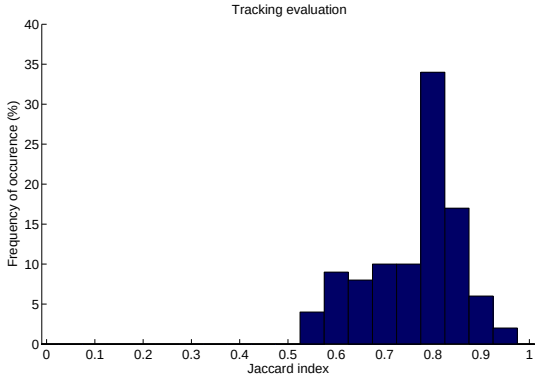


Figure III.4: A comparison of tracking data to ground truth for 100 random frames.

To estimate how well the system performs, ground truth-ellipses were manually entered for 100 random frames. For each of the frames, tracking data was compared to ground truth using the Jaccard index on the corresponding ellipses. The results are presented in figure III.4. Furthermore, a subjective evaluation of tracking quality was done by scoring tracking quality as 1 (good), 0.5 (decent), 0 (bad), for each of the 100 frames. The resulting total score out of the 100 frames was 85.

4 Future work

The system works well enough in practice, although more advanced techniques could be applied, for example condensation [1], Kalman filters, etc. Furthermore, modelling the subjects as ellipses has proved to be a source for errors in certain poses of the subjects.

Along with each experiment there is also a Unified Parkinson's Disease Rating Scale (UPDRS) score, which allows for further analysis. The primary goal is not to replace the UPDRS score, but rather to create a method to complement it (indeed, UPDRS assessment gives some data that cannot be measured by our video system - for example presence of tremors). In future work the strengths and weaknesses of the two assessment methods will be further evaluated.

To our knowledge, this is the first published description of an automated system for tracking of spontaneous motor behavior in marmosets in the used testing set-ups. However in previous studies, 3D tracking of the movement trajectories of freely moving rhesus monkeys performing outdoor spatial navigation tasks has been performed using related techniques [2]. Attempts have also been made to perform more detailed tracking of head movements in marmoset monkeys performing specific motor tasks but in this case manual tracking was chosen [5].

5 Acknowledgements

We are thankful to Dr. Miguel Nicolelis for providing the resources and facilities needed at the IINN-ELS research institute in Brazil where the experiments have been performed.

References

[1] A. Isard M. och Blake. “Condensation–conditional density propagation for visual tracking”. In: *International journal of computer vision* 29.1 (1998), pp. 5–28.

[2] Z. Khan, R.A. Herman, K. Wallen, and T. Balch. “An outdoor 3-D visual tracking system for the study of spatial navigation and memory in rhesus monkeys”. In: *Behavior research methods* 37.3 (2005), pp. 453–463.

[3] Tobias Palmér, Martin Tamtè, Pär Halje, Olof Enqvist, and Per Petersson. “A System for Automated Tracking of Motor Components in Neurophysiological Research”. In: *Journal of neuroscience methods* 205.2 (2012), pp. 334–344.

[4] C. Stauffer and W.E.L. Grimson. “Adaptive background mixture models for real-time tracking”. In: *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* Vol. 2. IEEE. 1999.

[5] B. Voelkl and L. Huber. “Imitation as faithful copying of a novel technique in marmoset monkeys”. In: *PLoS one* 2.7 (2007), e611.

Paper IV

Behavioural Brain Research 290 (2015) 90–101

Contents lists available at ScienceDirect

Behavioural Brain Research

journal homepage: www.elsevier.com/locate/bbr

Research report

Characterization of long-term motor deficits in the 6-OHDA model of Parkinson's disease in the common marmoset

M. Santana[a,d,g], T. Palmér[b,c], H. Simplício[a,e], R. Fuentes[a,f], P. Petersson[b,*]

[a] Edmond and Lily Safra International Institute of Neuroscience, Santos Dumont Institute, Macaíba/RN 59280-000, Brazil
[b] Integrative Neurophysiology and Neurotechnology, Neuronano Research Center, Department of Experimental Medical Sciences, Lund University, Medicon Village, Building 404 A2 Scheelevägen 8, 223 81 Lund, Sweden
[c] Mathematics LTH Centre for Mathematical Sciences, Faculty of Engineering, Lund University, S-22184 Lund, Sweden
[d] Psychobiology Program, Federal Univ. of Rio Grande do Norte, Natal, 59078-970, Brazil
[e] State Univ. of Rio Grande do Norte, Mossoró/RN, 59610-210, Brazil
[f] Programa de Fisiologa y Biofísica, Instituto de Ciencias Biomédicas, Facultad de Medicina, Universidad de Chile, Independencia 1027, Santiago 8389100, Chile
[g] Bacharelado em Ciências Biológicas, Instituto de Ciências e Tecnologia das Águas, Universidade Federal do Oeste do Pará (UFOPA), Santarém, Pará, Brazil

HIGHLIGHTS

- Two-stage bilateral 6-OHDA lesions produced long-lasting Parkinsonism in marmosets.
- An adapted motor disability scale was developed for the Parkinsonian marmoset.
- Motor behavior was quantified through mathematical image analysis techniques.

ARTICLE INFO

Article history:
Received 10 March 2015
Received in revised form 18 April 2015
Accepted 22 April 2015
Available online 28 April 2015

Keywords:
Animal model
Parkinson's disease
Motor behavior

ABSTRACT

Research aimed at developing new therapies for Parkinson's disease (PD) critically depend on valid animal models of the disease that allows for repeated testing of motor disabilities over extended time periods. We here present an extensive characterization of a wide range of motor symptoms in the 6-OHDA marmoset model of PD when tested over several months. The severity of motor deficits was quantified in two ways: (i) through manual scoring protocols appropriately adapted to include species specific motor behavior and (ii) using automated quantitative motion tracking based on image processing of the digital video recordings. We show that the automated methods allow for rapid and reliable characterization of motor dysfunctions, thus complementing the manual scoring procedures, and that robust motor symptoms lasting for several months could be induced when using a two-stage neurotoxic lesioning procedure involving one hemisphere at a time. This non-human primate model of PD should therefore be well suited for long-term evaluation of novel therapies for treatment of PD.

1. Introduction

When investigating new therapeutic approaches for PD, researchers crucially depend on valid animal models of the disease and reliable methods to assess the symptoms displayed. Although several different animal models of PD exist, the preferred choices by many labs today are the 6-hydroxydopamine (6-OHDA) lesioned rat or the MPTP-lesioned non-human primate [1,2], since these models have proven to capture several important features of the disease. However, MPTP is a severe safety hazard to the personnel handling the animals and strict procedures and appropriate laboratory safety equipment are an absolute requirement [3]. Consequently, there have also been a number of studies aimed at developing a primate model of PD based on intracerebral 6-OHDA lesions which would minimize the risk of inadvertent toxic exposure for researchers and animal care taking personnel that is associated with systemic MPTP treatment [4–7]. In parallel with the ongoing efforts to improve the reliability and validity of PD animal models, more sophisticated and diverse methods to assess severity of PD symptoms in animals has also been a key objective in the methodological development for several labs [8,9]. Given that the relevance of preclinical research ultimately is dictated not only by the validity of the model, but also to a great extent by the reliability and sensitivity of the testing

* Corresponding author at: Medicon Village, Building 404 A2 Scheelevägen 8 223 81 Lund, Sweden. Tel.: +46 46 222 05 78; fax: +46462756011.
E-mail address: Per.Petersson@med.lu.se (P. Petersson).

http://dx.doi.org/10.1016/j.bbr.2015.04.037
0166-4328/

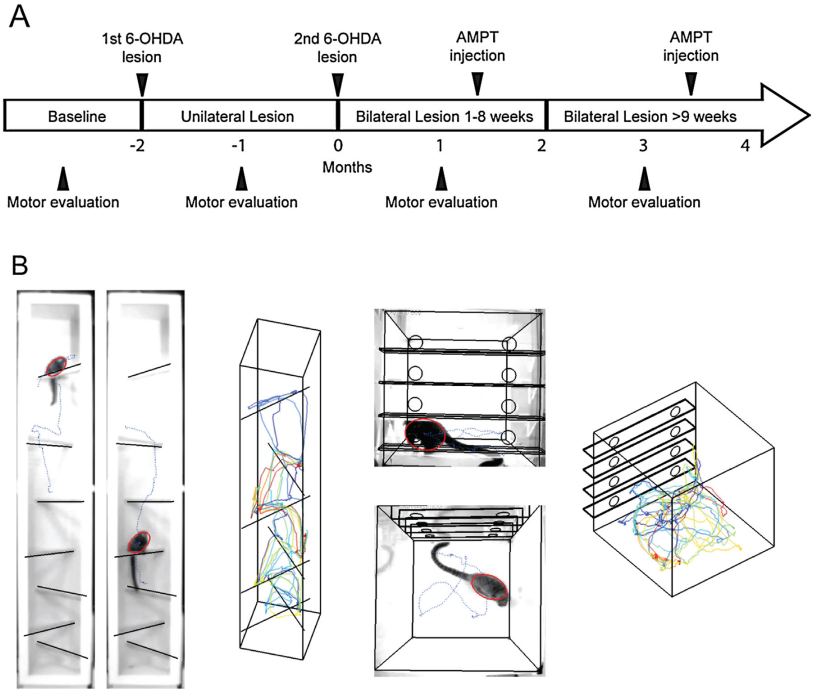


Fig. 1. Description of experimental procedures. (A) Timeline of experimental procedures. The two-stage 6-OHDA bilateral lesion procedure allowed for repetitive assessment of motor symptoms at gradually more severe stages of Parkinsonism over extended time periods. (B) Spontaneous locomotion was evaluated in two testing chambers designed to capture different types of locomotive behavior, including both horizontal and vertical locomotion in both set-ups. Left: Tower – two examples of typical movement bouts between bars locates at different heights are illustrated, blue lines denote the tracked movement traces and the red ellipses the position of the thorax as estimated by the image system. 3D image shows the total amount of locomotion displayed during a typical 5 min recording period in a healthy individual (Time represented in color code ranging from dark blue ($t = 0$) to red ($t = 5$ min), Right: Box - side and top view, respectively (color codes as in Tower).

methods used, this work aims towards further improvement of the procedures used to assess symptoms in animal models of PD. In particular, when evaluating new potential therapies, for example, neuromodulatory approaches like deep brain stimulation [10,11], or spinal cord stimulation [12,13], robust testing procedures are needed to allow researchers to repetitively assess the severity of the symptoms displayed over long time periods in response to changes in therapeutic interventions.

To this end, we have here developed new methods for behavioral assessment of PD symptoms in the 6-OHDA lesioned common marmoset (*Callithrix jacchus*). These procedures include manual scoring of PD symptoms according to an adapted PD motor rating scale, and automated movement tracking procedures based on digital video recordings. Using these methods, a thorough characterization of changes in motor behavior in nine 6-OHDA lesioned marmoset monkeys were conducted over a time period of several months. By testing the animals in four different symptomatic stages in a step-wise lesioning procedure, different levels of motor symptom severity could be characterized. The stages evaluated were: (1) intact state prior to lesion; (2) after unilateral lesion; (3) after bilateral lesion; and (4) after bilateral lesion plus treatment with the dopamine synthesis blocker alpha-methyl-p-tyrosine (AMPT). The order of successive lesions and testing procedures is shown in Fig. 1A.

2. Material and methods

2.1. Animals and housing conditions

Nine adult male common marmosets (*Callithrix jacchus*) 300–550 g were used in the study. The animals were housed in pairs in cages ($1.0 \times 1.0 \times 2.3\ m^3$) in a vivarium with natural light cycle (~12/12 h). Each cage has cover for rain and direct sun light, and the vivarium has a mobile roof that can be opened or closed according to weather changes such as heavy rain. Common marmosets are endemic to Northeast Brazil where the vivarium is located; thus ensuring suitable temperature, humidity and light conditions. To enrich the housing environment, cages are supplemented with elements such as sticks, tubes, ropes and ladders. Each cage has a small wooden box used as nest for protection and sleeping. Animals are offered two meals a day consisting of primate chow, local fruits, vegetables, mealworm larvae, gum arabic, dairy products, grains, eggs, and meat under the supervision of a veterinarian.

All animal procedures were carried out according to approved protocols by AASDAP Ethics Committee and strictly in accordance with the National Institute of Health Guide for the Care and Use of Laboratory Animals (NIH Publications No. 80–23). This project was approved by SISBIO/Brazilian Institute of Environment and Natural Resources (IBAMA) (No. 20795-2).

2.2. Procedure for 6-OHDA injections

The animals were initially sedated with ketamine (10–20 mg/kg, i.m.) and atropine (0.05 mg/kg i.m.) followed by deep anesthesia with isoflurane inhaled though a nose cone, to be finally intubated with an endotracheal tube and ventilated with artificial ventilator to be maintained with isoflurane 1–5% in oxygen at 1–1.5 L/min during the surgery. One mL of 6-OHDA hydrochloride (4 mg/mL dissolved in 0.05% ascorbate/saline solution) was freshly prepared and stored protected from light on ice before use. Five injections (2 μL each) were made with a 32 gauge Hamilton syringe at 0.5 μL per minute into the medial forebrain bundle (MFB) in the following locations (Anteroposterior/Mediolateral/Dorsoventral from the interaural midpoint): 6.5/1.2/6.0; 6.5/1.2/7.0; 6.5/2.2/6.5; 6.5/2.2/7.5; 6.5/3.2/8.0, which corresponds to a slightly modified version of the protocol used by Annett et al., 1992 [5]. Anteroposterior coordinates were corrected according to the dimensions of the skull of each animal based on the anatomy atlas by Stephan et al., 1980 [14]. After each infusion, the needle was left in place for another 3 min to allow the spread of the solution through the cerebral tissue at the exact area of interest.

During the 3–5 days following the surgery, the animals received non-steroid anti-inflammatory analgesic, flunixin meglumine (1 mg/kg, s.c.) and dexamethasone (0.5–1 mg/kg, i.m.), and a supplementary high-energy liquid diet. After at least eight weeks, the same procedures were repeated to cause a second 6-OHDA lesion in the other hemisphere, as previously described by Mitchell et al., 1995 [4]. The second 6-OHDA lesion was made in the contralateral hemisphere to the preferred limb.

2.3. Pharmacological dopamine depletion

Nine hours prior to behavioral testing the animals received a first injection of alpha-methyl-para-tyrosine (AMPT; 240 mg/kg, s.c. dissolved in physiological saline). This injection was repeated three hours later, resulting in a total administered dose of 480 mg/kg.

2.4. Adaptation to box and tower behavioral testing set-ups

Animals were accustomed to the behavioral testing procedures in a step-wise manner. First, while in their home cage, three times a week for two weeks, the animals were habituated to the food rewards used: $2 \times 5 \times 10\,mm^3$ (~60 mg) marshmallow pieces or mealworm (*Tenebrio molitor)* depending on the preference of each marmoset. Second, animals were accustomed to the transportation box (animals were allowed to explore the transportation box containing food baits while being free to return to their home cage at any time). Once showing interest in the transportation box, the animals were accustomed to a sound signaling entrance and another sound signaling exit from the transportation box. Animals were then trained to exit from the transportation box and explore the two different behavioral testing set-ups used in the study–a transparent cubic acrylic box ($0.45 \times 0.45 \times 0.45\,m^3$) and a vertical tower (width × depth × height: $0.36 \times 0.37 \times 2.20\,m^3$) with seven horizontal bars located at different distances above ground (0.1, 0.2, 0.4, 0.6, 0.9, 1.25, and 1.75 m; Fig. 1B) [15]. In this training, pieces of marshmallows were placed on the floor of the box or on the bars in the tower testing set-up to encourage the animal to explore the environment. A disposable white sheet of ethylene-vinyl acetate foam covered the floor to preventing the animal from slipping. This training scheme was performed twice a week for two weeks (in a parallel study animals were trained to reach and grasp food rewards through holes in one of the walls - this behavior was not evaluated in the current study and the shelves were not baited). All the procedures were performed either between 10:30 and 12:00 or between 14:00 and 15:30, corresponding to the natural peak of motor activity (cf. Fig. 3C). The food rewards obtained during training of the task replaced the juice portion that the animals would normally receive in their home cages.

2.5. Automated assessment of motor activity in home cage

Spontaneous motor activity of two animals were collected using actimeters (Actiwatch Mini, CamNtech) worn in custom made vests inside theirs home cage. The actimeters collected data every two seconds for three consecutive days (72 h) during the baseline, unilateral and bilateral periods. For the panel in Fig. 3B, the average raw motor activity of two consecutive 4am-6pm periods of the 72 h-recording session is represented in relation to the date of the second 6-OHDA lesion (except for the unilateral lesion period of Monkey 6 where only one 4 am–6 pm period was used, since the data from the second period was not available). For the graphics of Fig. 3 C, each recording was smoothed with a one-h (1800 samples) moving average window sliding at every sample, divided in two full 24-h periods, and the periods finally averaged.

2.6. Manual PD scoring

To evaluate the motor disability of the parkinsonian animals, we adapted the Unified Parkinson's Disease Rating Scale developed by Fahn and colleagues for the clinical setting [18] to fit aspects of non-human primate behavior based on previously developed procedures [16], [17].

The adapted scale consists of 16 categories scored from zero to three, which corresponds to absence of altered state to more intense symptomatology, respectively. Some categories involve symptoms that were evaluated for each body part individually (i.e., limbs, trunk, head), each receiving a maximum score of 3, thus, these categories could reach a maximum of 15 ("Tremor at rest" and "Tremor in motion") or 9 points ("Fine motor skills"), and were therefore subsequently normalized to 3 in order to facilitate the direct comparison of different categories of symptoms. Hence, the maximum total score of the scale is $16 \times 3 = 48$ points (Table 1).

The motor examination was performed in the animal's home cage. Assessments occurred at two times of the day: in the morning (~9 am) or afternoon (~5 pm). All tests were done before meals.

The quantified categories are the following:

(i) Tremor at rest
- [0]: Absent
- [1]: Occasional or detected rarely
- [2]: Frequent or easily detected
- [3]: Continuous and intense

(ii) Tremor in motion
- [0]: Absent
- [1]: Rarely detected, present during action
- [2]: Moderate amplitude, present during action
- [3]: Moderate amplitude, can interfere with feeding

(iii) Freezing
- [0]: Unhindered to move the body and show normal use of the limbs, e.g., in finding and grasping marshmallows in the reaching task
- [1]: Difficulties in starting to walk, or in the initiation of particular movements. For example, when reaching for a marshmallow, the start of the reaching movement is delayed. In these cases the freezing episodes are short
- [2]: Same as in [1], but the freezing episodes have a longer duration – between 5 and 10 s
- [3]: Same as in [1], but freezing episodes last over 10 s

Table 1
Summary of assessments performed in the nine male marmosets included in the study. Animals exposed to bilateral lesions were generally also assessed following the first lesion providing additional data to the hemilesioned group. When multiple tests were performed in the same animal in the Tower/Box set-up and through manual scoring, assessments were made during the same day to facilitate direct comparisons.

Animal	Lesion	Manual PD assessment	Tower test	Box test	Activity in home cage
1 - Beto	Bilateral	X	X		
2 - Dedé	Bilateral	X		X	
3 - Max	Bilateral	X			
4 - Tom	Bilateral	X	X		
5 - Kaká	Bilateral		X	X	
6 - Pele	Bilateral	X	X	X	X
7 - Zeca	Bilateral	X	X	X	X
8 - Deco	Unilateral	X	X		
9 - Kadu	Unilateral	X	X		

(iv) Gait and locomotion
- [0]: Walks normally according to pre-lesion locomotion patterns, with symmetrical limb use
- [1]: Shows reduced walking activity and walks with mild asymmetry
- [2]: Walks slowly, with asymmetry, and occasionally drags a limb (usually a hindlimb)
- [3]: Unable to walk

(v) Fine motor skills (scored for each arm independently)
- [0]: Normal ability to grasp marshmallows
- [1]: Grasps with difficulty
- [2]: Grasps with difficulty and requires one arm to support the stance while using the other to grab the marshmallow
- [3]: Totally unable to grasp marshmallows

(vi) Bradykinesia (scored independently for limbs and trunk)
- [0]: No difficulty in initiating or performing rapid and precise movements.
- [1]: Difficulties in initiating movements and displays smoother and slower movements when reaching for marshmallows or moving around spontaneously
- [2]: Clear delay in initiating movements and shows a marked slowing of movements in reaching and in spontaneous motor activity
- [3]: Totally immobile

(vii) Hypokinesia
- [0]: Moves freely and is alert and responsive
- [1]: Reduced activity, moves with less speed
- [2]: Low spontaneous activity, moves when provoked
- [3]: Totally immobile

(viii) Rigidity
- [0]: Moves freely; coordinated actions, absence of rigidity
- [1]: Mild rigidity or rigidity apparent only when other body parts are moving
- [2]: Striking stiffness, yet some complete movements are performed easily
- [3]: Severe rigidity, no movements are performed or movements appear incomplete

(ix) Body balance (Spontaneous behavior)
- [0]: Normal stance and coordination
- [1]: Compromised coordination, but is able to change from quadrupedalism to bipedalism without falling
- [2]: Compromised coordination, unstable locomotion with occasional falls
- [3]: Face down or lying in supine position unable to maintain any kind of stance

(x) Body balance (Induced behavior elicited by food offering)
- [0]: Normal stance and coordination
- [1]: Compromised coordination but changes from quadrupedalism to bipedalism, without falling
- [2]: Compromised coordination, unstable locomotion with occasional falls
- [3]: Face down or lying in supine position unable to maintain any kind of stance

(xi) Posture
- [0]: Normal posture
- [1]: Somewhat altered posture when standing, such as wider positioning of limbs. Resting with limbs and tail in abnormal body position
- [2]: Hunched posture, abnormal trunk position; abnormal head posture (neck flexed or inclined to one side)
- [3]: Unable to maintain posture, lying in supine or lateral position

(xii) Startle response
- [0]: Immediate, robust threat response
- [1]: Slightly diminished or delayed response, threats with open mouth
- [2]: Minimal or much delayed response, no open mouth threat
- [3]: No response to provocation

(xiii) Climbing
- [0]: Normal
- [1]: Climbs with difficulty. Slow on the branches and home cage mesh. No falling
- [2]: Very compromised. Climbs branches and cage mesh with great effort. Falling may occurs
- [3]: Not able to climb

(xiv) Gross motor skills (scored for each arm independently)
- [0]: Normal limb use when grasping larger objects
- [1]: Reduced ability to grasp larger objects to support body weight
- [2]: Rarely is able to grasp larger objects to support body weight
- [3]: Unable to grasp and hold large objects/structures

(xv) Facial expression
- [0]: Normal
- [1]: Slightly apparent decrease of facial expression (hypomimia)
- [2]: Moderate hypomimia with lips separated during brief moments
- [3]: Fixed face, severe or total loss of facial expression, lips separated in 6 mm or more

(xvi) Vocalization
- [0]: Normal quantity
- [1]: Spontaneous vocalization reduced
- [2]: Induced vocalization reduced
- [3]: Absent

For the categories “Climbing”, “Bradykinesia”, “Fine Motor Skills” and “Body Balance (Induced)”, mealworms or a piece of marshmallow were offered with tweezers to induce the desired motor behavior. For the evaluation of the category “Rigidity”, a blunt forceps was presented to the animal. Since the animal

associates the forceps with food offering, it would normally grab it. Following grasping of the forceps, gentle 'push and pull' movements were made to evaluate the level of stiffness of the forelimb. The procedure was repeated for both forelimbs.

In experiments involving AMPT-treatment, manual PD-scoring was performed six hours after the second injection.

2.7. Automated tracking procedures

Digital video recording were performed in the two testing setups. Two cameras (AVT–Stingray F033C, 80 fps) were used for digital video recordings in the acrylic box, from top and side views whereas tower activity was recorded using a single front view camera (AVT–Stingray F033C, 80 fps) (Fig. 1B). Motion tracking in the two setups was performed using similar methods. Software tools were developed in MATLAB and included mex-implementations (MATLAB compiled c-code; Mathworks Inc.). Constant light conditions during each recording session eliminated the need for advanced background models. Hence, a simple algorithm where each pixel is modeled as belonging to one of two Gaussian distributions was employed. The two distributions were estimated for each pixel by iterating through a sufficient number of frames of the video and updating the estimated parameters of the most probable distribution. In this case the background is contained in the brighter distributions, as the animals' image in these experiments was always darker than the actual background. After subtracting the background, the resulting foreground images were used in the shape analysis. By assuming that the two-dimensional image of the monkey in each camera plane is approximately elliptically shaped, the position and orientation of the animal could be estimated by the position and orientation of the three-dimensional ellipsoid that best fitted the foreground images. Given a measured foreground image F in a given camera and an estimated foreground image M generated by projection of the ellipsoid onto the camera plane, the matching quality is defined as

$$q = \frac{\sum_{i,j} \min(F(i,j), M(i,j))}{\sum_{i,j} \min(F(i,j), M(i,j))}$$

where i and j are the pixel coordinates. When multiple cameras were used, the combined quality measure was created by multiplying the individual q-scores. Note that the foreground image M is not actually computed, but instead the quality measure was computed using the projected quadric matrix for the ellipsoid. Movement tracking in time was then carried out by using the last known location to initiate estimation for a given frame followed by step-wise improvements of the matching quality by gradual adjustments of the parameters of the estimated ellipsoid. These calculations were performed for every frame in the video, resulting in the vectors $(x, y, z, \theta_x, \theta_y, \theta_z)$ describing the position and orientation of the estimated ellipsoid. Each vector will therefore be of the length N, where N is the number of frames in the video. Note that in the tower experiments, the z-coordinate was fixed and not estimated due to the use of only one camera (see http://homepages.inf.ed.ac.uk/rbf/VAIB14PAPERS/palmer.pdf for technical details on tracking procedures).

2.8. Automated extraction of kinematic parameters presented in plots

Relevant metrics summarizing changes in kinematic parameters over the different experimental conditions were constructed from the tracking data. From the (x,y,z) position vectors, speed was estimated as the Euclidean distance between (x_i, y_i, z_i) and $(x_{i+k}, y_{i+k}, z_{i+k})$, divided by k, frame number difference, and multiplied by the time resolution. Locomotion bouts were detected by applying a threshold on the acquired speed vectors. A locomotion bout was defined as the period of time where instantaneous speed was uninterruptedly greater than the chosen threshold. To improve robustness, multiple values of k were used for this detection, and all different estimates of the speed at a time have to be greater than the chosen threshold (approximately corresponding to a speed of 0.04 m/s).

From each locomotion bout, a number of different parameters were obtained: maximum speed, average speed, distance covered, duration and maximal acceleration.

2.9. Tyrosine-hydroxylase staining and quantification

After the period of the experiments, the animals were sacrificed by intracardiac perfusion after deep sedation with ketamine (40 mg/kg i.m.); (xylazine 0.04 mg/kg i.m.) and atropine (0.05 mg/kg i.m.). Intracardiac perfusion was performed with 0.9% saline solution and heparin at 37 °C, followed by 4% paraformaldehyde in phosphate buffer, 0.1 M (pH 7.4), cooled to 4 °C. The brains were removed and postfixed in the same solution for 2 h, washed in 0.1 M phosphate buffer (pH 7.4) at 4 °C for 24 h, cryoprotected in 20% following 30% sucrose solution at 4 °C, and finally rapidly frozen for cryostat embedding in Tissue-Tek medium. The brains were kept in a freezer at −80 °C until sectioned coronally at 50 μm in a cryostat.

Immunohistochemical staining was performed free-floating or on sections mounted directly on electrically charged glass slides. The sections were processed for immunohistochemical detection of tyrosine hydroxylase (TH) in substantia nigra and in striatal regions using modifications of the protocol of Eslamboli et al. (2003) [19].

The sections were washed in 0.1 M phosphate buffer (PB) for 5 minutes. Then, incubated in 1% hydrogen peroxide/methanol solution for 20 minutes to remove endogenous peroxidase activity, and rinsed in 0.05% phosphate buffer-Tween 0.05% (PB-T) for 5 minutes. Thereafter, the sections were confined with the aid of a hydrophobic PAP pen and incubated in 10% goat normal serum diluted in 0,1 PB for 30 minutes. Excess serum was removed and sections were incubated in the primary anti-TH (rabbit polyclonal antibody; 1:500; diluted in normal serum/PB) overnight at room temperature in a humidity chamber to prevent air-drying of the tissue sections. The sections were washed with PB-T (5 min) and incubated in biotinylated goat anti-rabbit secondary antibody (1:200, diluted in PB; Vector Labs) for two hours. After that, the sections were washed again with PB-T (5 min) and incubated in avidin-biotin-peroxidase solution (Vectastain Standard ABC kit, Vector Laboratories) for one hour.

After removal of the ABC solution, the sections were washed in PB (5 min) and placed in a solution containing 0.03% 3,3′-diaminobenzidine tetrahydrochloride hydrate (DAB) (Sigma) and 0.001% hydrogen peroxide in 0.1 M PB. The reaction was monitored in a light microscope. The sections were washed and slides were left to dry overnight. After dehydration through a series of graded alcohols and clearance in xylene, the slides were cover-slipped using Entellan mounting medium.

2.10. Quantification of striatal and nigral tyrosine hydroxylase immunoreactivity

The tissue samples were mounted and photographed using a microscope with the same camera configuration and under identical illumination conditions. TH reactivity in both striatum (caudate and putamen) and in substantia nigra pars compacta (SNc) was assessed by computer densitometry using digital images captured from a camera (CX9000, MBF Bioscience) attached to the microscope (light field Nikon Eclipse 80i - 10x and 20x objectives). TH-reactivity across the striatum was assessed by optical densitometry

using ImageJ software (NIH, http://rsb.info.nih.gov/ij/). Measurements were obtained using a 0.2 mm^2 square window positioned in different regions throughout the striatum (60 samples per striatum and animal). To reduce the effects of within-group variability, a normalized scale based on the reactivity for TH of the internal capsule (white matter) was adopted (average over measurements of 10 different sites using the same window). For each animal, a contrast index was calculated according to the equation: $C = (G - W)/(G + W)$ [20], in which G is the average optical density of striatal tissue, and W is the optical density of the white matter (internal capsule). To count TH-labeled cells, we used at least three sections per animal. For the different positions along the rostral-caudal axis (rostral, central and caudal area), the boundaries of the SNc were defined in each section according to the atlas by Paxinos et al. [21] and the area of the SNc was calculated using the sections from control animals (it was not possible to identify the SNc contours in the lesioned animals because of the substantial loss of dopaminergic neurons from 6-OHDA treatment). Cells labeled with TH within the defined areas were subsequently counted (StereoInvestigator system, MBF Bioscience Inc) and the resulting cell densities were expressed as TH^+ cells/mm^2.

2.11. Statistical analyses

The statistical tests used in the study are specified in the main text and in the figure legends together with the data used for the respective test. Analyses of significance were performed using either Matlab functions or GraphPad Prism 5.01 software.

3. Results

3.1. Acute effects of 6-OHDA lesions

Immediately following the first lesion, animals showed rigidity in the limbs and visuospatial neglect contralateral to the lesioned hemisphere and ipsilateral head position deviation. In addition, animals showed ipsilateral body rotation while trying to ambulate, and difficulty to use the forelimb contralateral to the lesioned hemisphere. In agreement with earlier studies, these symptoms, which were observed already the first days after the lesioning surgery, spontaneously recovered almost completely during the first two weeks [5], [19]. In spite of these evident motor symptoms the animals were still able to feed themselves in their home cages (as indicated by a < 10% weight loss following surgeries). At least eight weeks later, the animals were exposed to a second injection of 6-OHDA in the opposite hemisphere (Fig. 1A). Directly following this second lesion, animals generally showed similar but more severe motor impairments, requiring special care in some cases when animals had difficulties feeding themselves, to ensure weight loss would not exceed 10% of total body weight during the first two weeks following surgery [22]. Animals were allowed to recover for two weeks before assessments of PD symptoms in the box and tower behavioral testing set-ups commenced, both after the first and the second lesion.

3.2. Evaluation of motor symptoms using an adapted PD motor disability rating scale

In the manual assessment of PD-symptoms a total of 16 different categories were evaluated: (1) resting tremor, which was not observed in this model; (2) tremor in motion and sporadic postural tremor; (3) episodes of freezing – brief periods of sudden immobility when initiating quadripedal locomotion or goal directed reaching; (4) uncoordinated gait – inaccurate positioning of the limbs and wobbling of the trunk during locomotion (in the literature referred to as clumsy, poor-balanced gait; Eslamboli, 2003); (5) deficits in fine motor skills – difficulty using arms to grab any food offered (in some animals the weakness was exacerbated by a worsening of gross motor skills, see below); (6) bradykinesia - noticeable slowing of the execution of movements; (7) hypokinesia – a general reduction in motor activity (motility, grooming, climbing); (8) rigidity – particularly noticeable in forelimbs during extension; (9 and 10) body balance – abnormal body positions and difficulty to rest on branches; (11) hunched posture; (12) a slowed startle response – animals would not respond to alarm vocal signals from mates; (13) slowed climbing; (14) loss of gross motor skills – for example, inability to grasp branches; (15) episodes of hypomimia – reduction of the marmoset's typical behavior of maintaining eye contact and impaired inability to display facial expression in response to interaction with care givers; and (16) lack or decrease of vocalizations (marmosets use vocalizations abundantly to communicate between them).

For each of these 16 categories, the severity of motor disability was repeatedly evaluated in every individual in a total of eight animals under different degrees of Parkinsonism. Following the first lesion, stable parkinsonian symptoms were observed in all individuals over a testing period spanning at least 8 weeks after lesioning surgery (average score [mean ± SEM], week 1–8: 6.9 ± 1.0; Fig. 2A, left). After the second lesion, symptoms were on average more severe compared to the first unilateral lesion during the corresponding assessment period (average score week 1–8: 12.6 ± 0.7, Fig. 2A, right). Animals were then monitored for another few months and persistent symptoms were confirmed. However, during these extended testing periods a certain degree of spontaneous recovery was observed resulting in a gradual decline of the total PD score over a 32 week period (Fig. 2A). However, severe Parkinsonism could always be transiently reinstated for ~18 h through systemic treatment with the dopamine synthesis blocker AMPT (average score under AMPT effect for week 1–16: 24.6 ± 1.8; week 17–32: 19.7 ± 1.0). Interestingly, the degree of functional recovery varied substantially between different types of motor symptoms. When analyzing the PD-scores for each category of symptoms divided into 8-week periods following the second lesion it became evident that for example symptoms related to locomotion and body balance during spontaneous behavior showed negligible improvements over time (Fig. 2B). These findings indicate that quantitative assessments of spontaneous locomotor behavior could be particularly useful in experiments where testing periods lasting over several months are required.

3.3. Twenty-four hour recordings of motility in the home cage

As a complement to the detailed manual assessments of dysfunctions in motor behavior, the overall spontaneous motor activity during 72 h periods in the home cage was also recorded in two animals. It was found that the absolute amount of motor activity was clearly decreased following the first and second lesion, with a relative decrease after unilateral lesion corresponding to: -44% and −39%, and after bilateral lesion: −78% and −36% for the two monkeys, respectively (Fig. 3A and B). At the same time, the characteristic variations in the relative amount of motor activity displayed throughout the day-night cycle was comparatively preserved also in the parkinsonian state (Fig. 3C).

3.4. Automatic assessment of locomotive activity in the Tower testing set-up

The spontaneous locomotion of the animal was recorded for 5 min in each session, in a total of 120 testing sessions in

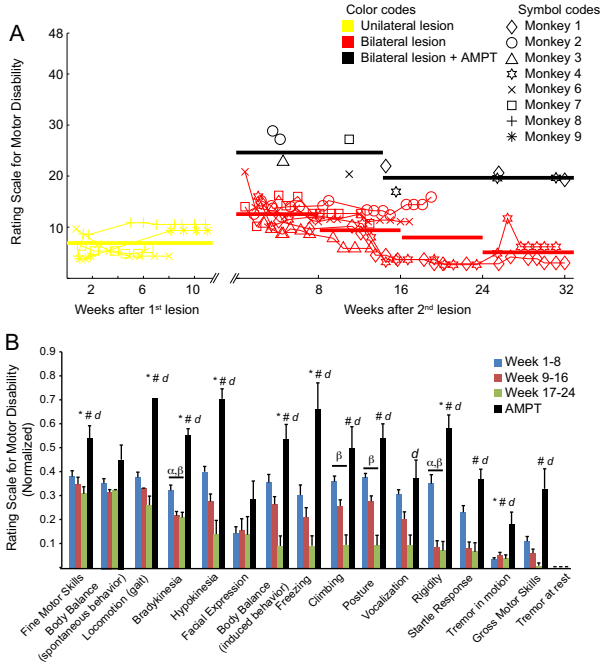


Fig. 2. Manual scoring of motor impairments. (A) Total motor disability score of individual animals during ten weeks following the first lesion ($n = 4$, yellow), 32 weeks following the second lesion ($n = 6$), and under additional treatment with the dopamine synthesis inhibitor AMPT (2×240 mg/kg AMPT administered 6 and 9 h prior to testing; $n = 6$). On the y-axis, 48 points represents the highest possible total score when the partial scores of the 16 categories are added, and zero corresponds to pre-lesion behavior for each individual. The average score during each testing period and condition is represented by the thick horizontal lines. (B) Normalized scores of motor impairment after the second lesion divided by symptom category and testing period, week: 1–8, 9–16, and 17–24 after lesion, and under the additional effect of AMPT (week 1-32 after lesion; mean values shown, error bars represents S.E.M.). Significant differences in average scores were found week [1–8] vs. [9–16] (α, $p < 0.01$), [1–8] vs. [17–24] (β, $p < 0.01$), [1–8] vs. [AMPT] (*, $p < 0.05$), [9–16] vs. [AMPT] (#, $p < 0.01$), and [17–24] vs. [AMPT] (d, $p < 0.05$; ANOVA for repeated measures ($p < 0.05$) with post hoc Bonferroni-corected paired tests).

7 animals. The distance travelled during the testing session was subdivided into vertical and horizontal translation (Fig. 4A). It was evident that intact animals were considerably more active than lesioned animals and that the distance travelled successively declined in the more severe PD models (Fig. 4B). On average the distance traveled (horizontal/vertical) in meters per minutes for animals in the four different stages of Parkinsonism was (mean ± SD), intact: $1.51 \pm 0.52/2.54 \pm 1.41$, hemilesion: $0.81 \pm 0.41/1.16 \pm 0.85$, bilateral: $0.50 \pm 0.28/0.49 \pm 0.29$, bilateral + AMPT: $0.17 \pm 0.06/0.18 \pm 0.08$ (Fig. 4B; showing a statistical group difference, $p < 0.05$, Kruskal-Wallis). These motor deficits were consistent across animals (Fig. A.1). Furthermore, healthy individuals preferred staying on the bars positioned relatively higher up in the tower in contrast to the parkinsonian animals resulting in significant differences in mean expectation values in height over ground for the four groups (Fig. 4C; $p < 0.05$, Kruskal-Wallis). Finally, we also observed that when moving between different heights, healthy individuals often displayed longer uninterrupted movement bouts involving multiple transitions between different levels, whereas the lesioned animals moved more frequently one level at a time (fraction of multi-level transitions for the four groups were: 0.23 ± 0.14, 0.09 ± 0.07, 0.03 ± 0.03, 0.05 ± 0.10; $p < 0.05$, Kruskal-Wallis Fig. 4D).

3.5. Automatic assessment of locomotive activity in the Box testing set-up

Spontaneous locomotion in the transparent cubical box was quantified from ~5 min recordings in a total of 120 testing sessions in 4 animals. Similarly to the tower test, the distance travelled was subdivided into vertical and horizontal translation and in agreement with the behavior in the tower, the distance travelled was clearly reduced in the more severe PD models. On average the distance travelled (horizontal/vertical) per minute for the four groups was (mean ± SD), intact: $2.34 \pm 0.86/0.55 \pm 0.39$, hemilesion: $0.96 \pm 0.40/0.36 \pm 0.41$, bilateral: $0.23 \pm 0.12/0.04 \pm 0.03$, bilateral + AMPT: $0.14 \pm 0.10/0.04 \pm 0.05$ (Fig. 5A; showing a statistical group difference, $p < 0.01$ for both horizontal and vertical distance, Kruskal-Wallis). These motor deficits were consistent across animals (Fig. A.2). A more detailed analysis of the locomotion bouts revealed further differences in the pattern of locomotion. We found that (1) bout duration, (2) bout maximum speed, (3) bout distance, as well as (4) frequency by which bouts of locomotion were displayed were all reduced in parkinsonian animals (Fig. 5B). Finally, in order to verify that the motor deficits observed were stable over extended time periods, the individual experiments were ordered and analyzed with respect to the time of

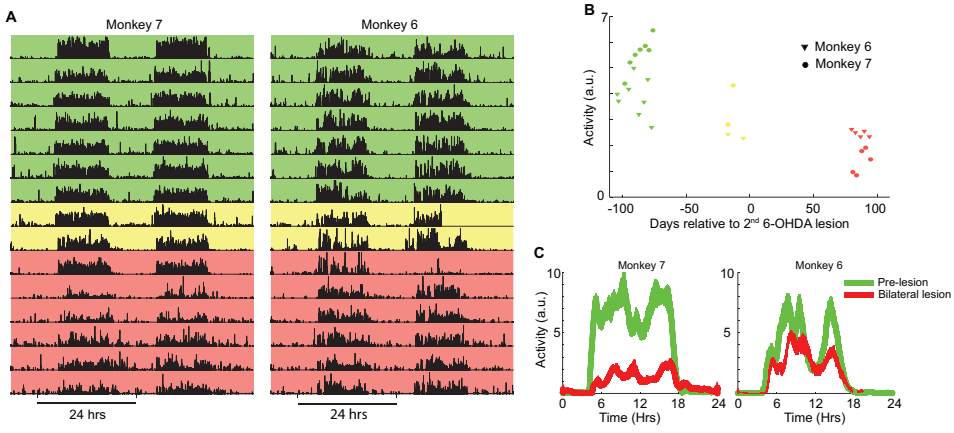


Fig. 3. General activity in home cage before and after 6-OHDA lesions, measured in two animals during 72h-recordings using actimeters. (A) Green panels show data collected during baseline conditions prior to lesioning surgery, yellow and red panels represent the activity displayed after unilateral and bilateral lesions, respectively. (B) Average activity recorded during the active periods of the day (4am–6pm) during the three different conditions (baseline, unilateral and bilateral lesion). On the x-axis, day zero corresponds to the day of the second lesion, days −50 to 0 to unilateral lesion, and earlier than day −50 represents baseline recordings. (C) The average activity displayed during circadian cycle for baseline and bilateral lesion conditions.

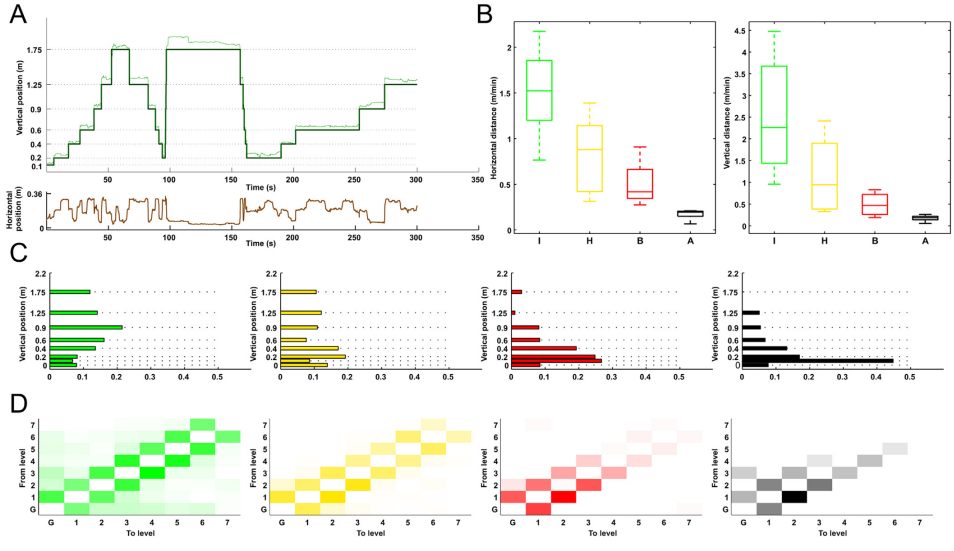


Fig. 4. Behavioral testing in Tower. Quantification of spontaneous horizontal and vertical locomotion in the tower testing set-up reveals clearly different behavior in the four different stages of Parkinsonism. (A) Example of vertical (green trace) and horizontal (brown trace) displacement of the animal during a 5 min recording session (height over ground for the different bars are denoted on the axis to the left, tracking data quantized to the levels are shown in the thick lines and the original tracking data are shown as thinner lines). (B) Summary of the average horizontal and vertical distance travelled in all recordings (median, 25% and 75% percentiles shown in boxes, whiskers denote range). Note the successive decrease in distance travelled in the more severe PD models [green = intact (I), yellow = hemilesion (H), red = bilateral lesion (B), black = bilateral lesion + AMPT (A)]. Kruskal-Wallis, $p < 0.05$, post tests showed significant differences for: [Horizontal: I vs. H, B and A; H vs. A; B vs. A] and [Vertical: I vs. B, A; H vs. A], $p < 0.05$ Wilcoxon rank-sum test). (C) A change in the preference for bars located relatively higher up to bars at lower levels with increasing severity of Parkinsonism. The relative amount of time spent on the respective level is indicated by colored bars. (D) Transition matrices describing the probability that the animal will move from a certain level (row) to another level (column). Levels are denoted from G to 7, where G is ground and 7 is the highest bar, each treatment group is normalized to the total number of transitions observed in that condition. It can be noted that animals move less frequently more than one level at a time and between the higher bars with more severe parkinsonian symptoms.

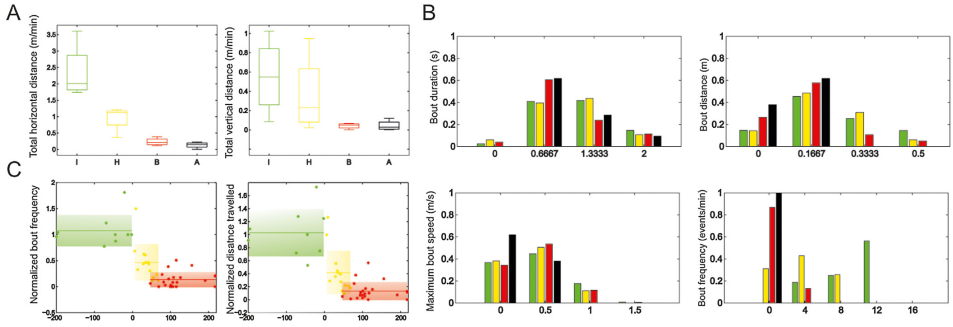


Fig. 5. Behavioral testing in Box. Quantification of spontaneous locomotive behavior in the box set-up reveals marked differences between the different degrees of Parkinsonism. (A) The average horizontal and vertical distance travelled per minute for animals grouped according to severity of Parkinsonism showed a gradual decline. Kruskal-Wallis, $p < 0.05$, post tests showed significant differences for: [Horizontal: I vs. H, B and A; H vs. A] and [Vertical: I vs. B], $p < 0.05$ Wilcoxon rank-sum test). (B) Differences in bout duration, distance, speed and frequency, shown for the different groups in histograms representing the relative frequency of the observed parameter values in four equally sized intervals of the full range for the respective parameters. (C) Bout frequency and total distance travelled shown for all recorded sessions divided according to lesion group (mean and SD indicated by horizontal line and box, respectively). Note the robust reductions following both the first and second lesion which persist throughout each >8 week long testing period. [Color code: green = intact (I), yellow = hemilesion (H), red = bilateral lesion (B), black = bilateral lesion + AMPT (A)].

assessment in relation to the two lesion procedures. To eliminate any inter-individual variability, all the analyzed features of the locomotive behavior were normalized to the motor behavior displayed by each individual during baseline conditions. While slight variations were found between different recording sessions during each of the three conditions, a much greater difference was observed between intact, hemilesioned and bilaterally lesioned animals. Notably, these differences persisted over several months and were found to be particularly evident for bout distance and the frequency by which bouts of locomotion were displayed (data from all recordings in all four animals, normalized to the intact activity level of each individual, are shown in Fig. 5C; see also Fig. A2).

3.6. Immunohistochemical verification of 6-OHDA lesions

Subsequent to these extensive characterizations of behavioral changes during parkinsonian conditions, *post mortem* tissue analyzes were performed. Immunohistochemistry for tyrosine hydroxylase (TH) was used to quantify the extent of the lesions (Fig. 6A and B). A reduction in the number of TH-positive cells of the midbrain dopaminergic neurons projecting to the forebrain in the lesioned hemispheres was confirmed. The cell densities (number of cells/mm^2) were (mean ± SEM), 57.66 ± 6.23 and 139.01 ± 12.13 in bilaterally lesioned and control animals, respectively ($P < 0.0001$, $U = 28$, Mann-Whitney U-test; Fig. 6C, bottom panel). The axonal terminal density of TH positive cells projecting to the caudate-putamen was also quantified. A contrast index was used to quantify the TH-staining in relation to background staining (see Methods for detail) showing a significant reduction of TH-immunoreactivity in lesioned animals vs. controls in both the caudate nucleus (0.155 ± 0.01 vs. 0.254 ± 0.02; $P < 0.05$, $U = 112$, Mann-Whitney U-test) and in putamen (0.135 ± 0.02 vs. 0.213 ± 0.02; $P < 0.05$, $U = 109$, Mann-Whitney U-test; Fig. 6C, top panel). Taken together, the average staining intensity of terminals in the caudate-putamen of lesioned animals was 44%, and the density of stained midbrain cells 41% compared to intact animals.

4. Discussion

Non-human primate models have a key role in PD-research aimed at understanding the underlying pathophysiology of the disease, as well as for the development of new treatment strategies. Whereas experiments in rodents in many cases can provide important insights in the early phase of basic PD-research, results are not always transferable to humans. In particular, the large difference in overall neuroanatomical complexity between the rodent and primate central nervous system can sometimes make findings in rats and mice less clinically relevant. The possibility to perform large scale experiments in the MPTP-treated macaque – which by many researchers is regarded as the most valid model of PD – is on the other hand very limited due to the high costs associated with housing and treating these larger primates and the safety precautions required for safe handling of these animals in order to avoid inadvertent neurotoxic exposure. In this perspective the 6-OHDA marmoset model of PD, which we here thoroughly characterized, may present a valuable complement.

Investigations aimed at developing prospective treatments for PD generally demand long evaluation periods, it was therefore important to systematically evaluate the marmosets with respect to a range of motor deficits over a time period of several months following lesions. To facilitate comparisons with other primate studies using systemic MPTP treatment and to allow for rapid behavioral assessments of bilateral motor symptoms that affect the pattern of locomotion, the majority of the monkeys (7/9) in our study were exposed to bilateral lesions. It should be noted however, that for other experimental paradigms (involving for example tests of asymmetrical limb use, forelimb dexterity etc.) unilateral lesions may be preferable since they induce less severe symptoms and, hence, also demand less post-surgical care efforts by researchers and veterinaries.

While a recovery of certain motor functions was observed after about three months in the detailed manual PD-scoring assessments (including for example, vocalization, startle response, rigidity and posture), other symptoms remained relatively stable (such as fine motor skills, locomotion, body balance in spontaneous locomotion and bradykinesia) also after more than six months following lesions, indicating that this model may indeed be useful for the purpose of evaluating novel PD therapies under chronic disease conditions. Moreover, in experimental situations where severe Parkinsonism is desired, the additional pharmacological treatment with the dopamine synthesis inhibitor AMPT reproducibly induced marked motor disability in all animals tested. In spite of the

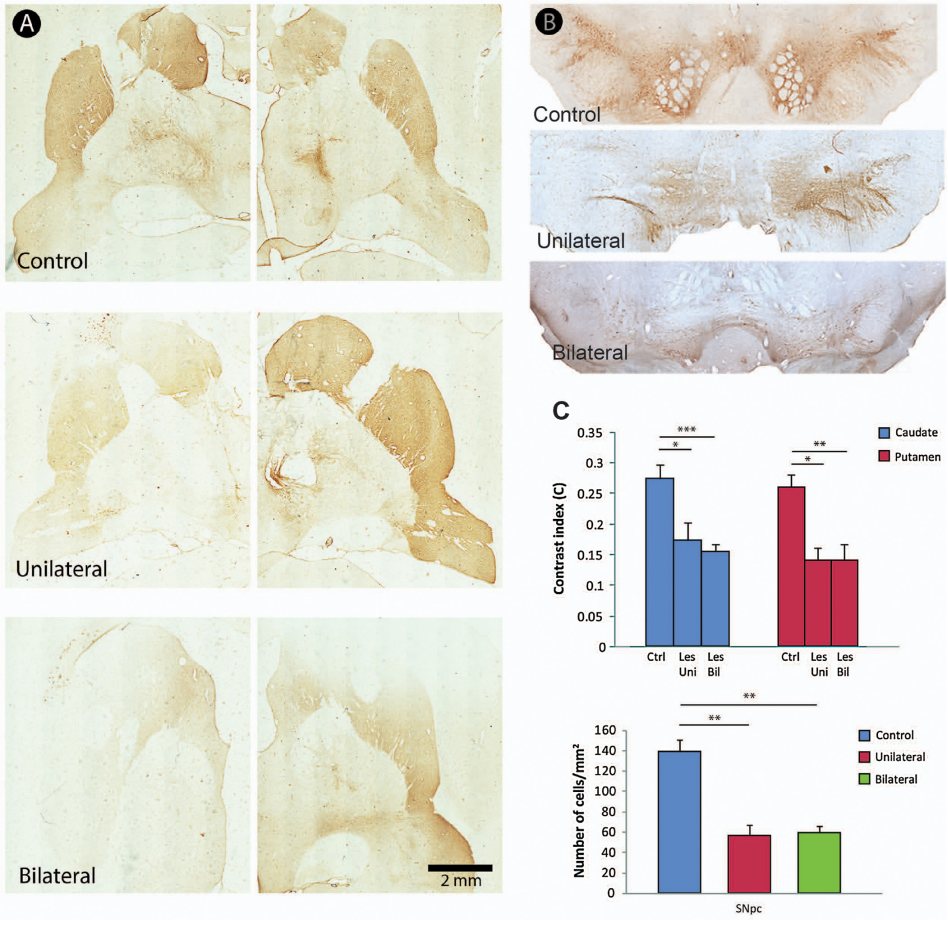


Fig. 6. Histological confirmation of dopaminergic lesions. (A) Examples of tyrosine hydroxylase (TH) immunolabeling in the caudate (Cd), putamen (Put). Immunohistochemistry of TH showed intense labeling of Cd-Put in both hemispheres in control animals (top panel). Lesions performed in the left hemisphere induced a pronounced loss of labeling in Cd/Put on this side (middle panel). In bilaterally lesioned animals, both sides of Cd/Put were strongly affected, showing much weaker TH-staining (bottom panels) compared to controls. (B) Examples of TH immunolabeling in the substantia nigra (SN). In SN, TH-labelling of cell bodies of midbrain dopaminergic neurons in the intact brain is evident but is strongly reduced in lesioned hemispheres. (C) Quantitative summary of TH-immunolabeling of terminals in the caudate-putamen (top) and of cell-bodies in substantia nigra (bottom) confirming extensive dopaminergic lesions.

comparatively severe symptoms that are transiently induced under such conditions, these tests were well tolerated and could be repeated multiple times in all animals.

The use of automated procedures for the analysis of spontaneous locomotive behavior provided important information on motor dysfunctions complementing the outcome of the manual scoring of PD symptoms. Both in 24 h home cage recordings and in the shorter testing sessions in the Tower and Box set-ups, consistent differences between the different parkinsonian states were observed. In fact, even though behavior was only recorded for a few minutes in each testing session, a persistent reduction in locomotor-related kinematic parameters such as bout frequency and total distance travelled could reliably be detected and was confirmed in consecutive testing sessions over very long time periods following the second lesion. In addition, quantitative measurements of bradykinesia (a main symptom in PD) which is very challenging to characterize using manual procedures was reliably obtained in all testing sessions. The speed an ease by which these assessments can be performed is particularly valuable for experimental paradigms where a rapid evaluation of an experimental treatment paradigm is required, such as for neuromodulatory approaches using electrical stimulation of neuronal structures.

To attain a closer link to the assessment procedures used in the clinic, we here also adapted the unified Parkinson's disease rating scale [18] to the marmoset model of PD. Out of the 16 categories of the adapted rating scale, 11 categories were judged to be more reliably assessed using manual scoring than automated measures (i.e., all symptoms that were not directly linked to aspects of locomotion). Consequently, manual PD-scoring was shown to be particularly helpful in providing a broad characterization of motor dysfunctions using methods closely resembling those used in Parkinson patients.

Taken together, because the automated procedures have the advantage of providing rapid and robust measures of certain motor symptoms, while the manual scoring procedures on the other hand, can provide a more complete view of complex patterns of symptoms involving a broader range of motor behavior, it is advisable that the methods used are selected based on the behavioral tasks that are most appropriate in any given study.

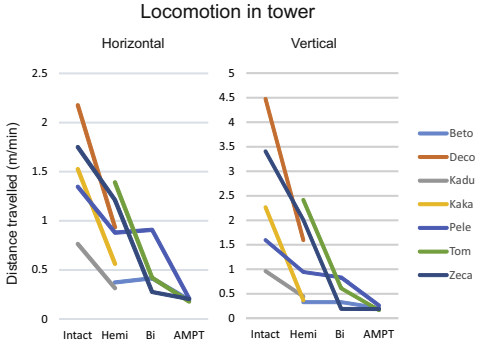
It can be concluded that, using the methods developed herein, the two-stage 6-OHDA marmoset model of PD provides a robust and reliable primate model of PD lasting for periods of months that can potentially have an important role in the future development of novel therapies.

Acknowledgments

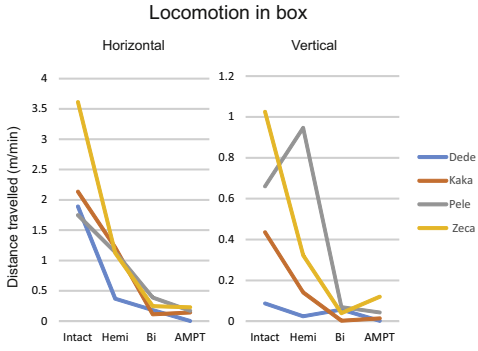
We are thankful to Miguel Nicolelis for creating the research environment needed to conduct this research in Macaíba/RN and for supporting our work. We would also like to thank Marcelo Carvalho for technical support. This research was supported by The Michael J. Fox Foundation for Parkinson's Research, FINEP 01.06.1092.00, and INCEMAQ–Program of National Institutes of Science and Technology of CNPq/MCT; AASDAP - Alberto Santos Dumont Association for Research Support; The Swedish Research Council [#325-2011-6441], the Olle Engkvist, Parkinson Research, Crafoord, Åke Wiberg, Magnus Bergvall, Kockska and Segerfalk Foundation.

Appendix A.

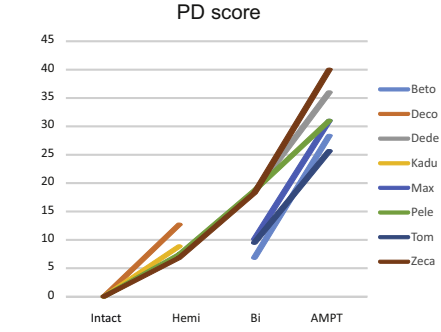
The average amount of locomotion displayed for each of the recorded individuals in the four different stages of Parkinsonism is summarized in Figs A.1 and A.2 illustrating that the results presented Figs. 4 and 5, are representative also for each individual. The total manual PD score is shown in Fig. A.3 for comparison (intact score is zero for all animals).



A.1. Locomotion in tower.



A.2. Locomotion in box.



A.3. PD score.

References

[1] Dauer W, Przedborski S. Parkinson's disease: mechanisms and models. Neuron 2003;39:889–909.
[2] Eslamboli A. Marmoset monkey models of Parkinson's disease: Which model, when and why? Brain Res Bull 2005;68:140–9.
[3] Przedborski S, Jackson-lewis V, Naini AB, Petzinger G, Miller R, Akram M. The parkinsonian toxin (MPTP): a technical review of its utility and safety. J Neurochem 2001:1265–74.
[4] Mitchell CBJ, Hughes IJ, Carroll N. Reversal of parkinsonian symptoms by intrastriatal and systemic manipulations of excitatory amino acid and dopamine transmission in the bilateral 6-OHDA lesioned marmoset. Behav Pharmacol 1995;6:492–507.
[5] Annett LE, Rogers DC, Hernandez TD, Dunnett SB. Behavioural analysis of unilateral monoamine depletion in the marmoset. Brain 1992;115(3):825–56.
[6] Ungerstedt U. 6-Hydroxydopamine-induced degeneration of the nigrostriatal dopamine pathway: the turning syndrome. Pharmacol Ther B 1976;2:37–40.
[7] Klintenberg R, Arts J, Jongsma M, Wikström H, Gunne L, Andrén PE. Motor effects of a dopamine stabilizer (GMC1111) in primate models of Parkinson and hemiparkinsonism. Eur J Pharmacol 2003;459:231–7.
[8] Eslamboli A, Romero-Ramos M, Burger C, Bjorklund T, Muzyczka N, Mandel RJ, et al. Long-term consequences of human alpha-synuclein overexpression in the primate ventral midbrain. Brain 2007;130:799–815.
[9] Okano H, Hikishima K, Iriki A, Sasaki E. The common marmoset as a novel animal model system for biomedical and neuroscience research applications. Semin Fetal Neonatal Med 2012;17:336–40.
[10] Benabid AL, Pollak P, Louveau A, Henry S, de Rougemont J. Combined (thalamotomy and stimulation) stereotactic surgery of the VIM thalamic nucleus for bilateral Parkinson disease. Appl Neurophysiol 1987;50:344–6.
[11] Benabid AL, Pollak P, Hoffmann D, Gervason C, Hommel M, Perret JE, et al. Long-term suppression of tremor by chronic stimulation of the ventral intermediate thalamic nucleus. Lancet 1991;337:403–6.

[12] Fuentes R, Petersson P, Nicolelis MAL. Restoration of locomotive function in Parkinson's disease by spinal cord stimulation: mechanistic approach. Eur J Neurosci 2010;32:1100–8.
[13] Fuentes R, Petersson P, Siesser WB, Caron MG, Nicolelis MAL. Spinal cord stimulation restores locomotion in animal models of Parkinson's disease. Science 2009;323:1578–82.
[14] H. (Heinz) Stephan, G. (Georg) Baron, W.K. (Walter K. Schwerdtfeger, The brain of the common marmoset (Callithrix jacchus): a stereotaxic atlas/H. Stephan, G., Baron, W.K. Schwerdtfeger, (1980).
[15] Verhave PSRa, Vanwersch P, van Helden HPM, Smit AB, Philippens IHCHM. Two new test methods to quantify motor deficits in a marmoset model for Parkinson's disease. Behav Brain Res 2009;200:214–9.
[16] Bankiewicz KS, Sanchez-Pernaute R, Oiwa Y, Kohutnicka M, Cummins A, Eberling J. Preclinical Models of Parkinson's Disease. In: in: Curr. Protoc. Neurosci. John Wiley & Sons, Inc; 2001.
[17] Hansard MJ, Smith LA, Jackson MJ, Cheetham SC, Jenner P. Dopamine, but not norepinephrine or serotonin, reuptake inhibition reverses motor deficits in 1-methyl-4-phenyl-1,2,3,6-tetrahydropyridine-treated primates. J Pharmacol Exp Ther 2002;303:952–8.
[18] Fahn S, Elton RLUD. Committee, others Unified Parkinson's disease rating scale. Recent Dev Park Dis 1987;2:153–63.
[19] Eslamboli A, Baker HF, Ridley RM, Annett LE. Sensorimotor deficits in a unilateral intrastriatal 6-OHDA partial lesion model of Parkinson's disease in marmoset monkeys. Exp Neurol 2003;183:418–29.
[20] Freire MAM, Morya E, Faber J, Santos JR, Guimaraes JS, Lemos NAM, et al. Comprehensive Analysis of Tissue Preservation and Recording Quality from Chronic Multielectrode Implants. PLoS One 2011;6:e27554.
[21] Paxinos G, Watson C, Petrides M, Rosa M, Tokuno H. The Marmoset Brain in Stereotaxic Coordinates. San Diego: Academic Press; 2012.
[22] Mitchell IJ, Carroll CB. Reversal of parkinsonian symptoms in primates by antagonism of excitatory amino acid transmission: potential mechanisms of action. Neurosci Biobehav Rev 1997;21:469–75.

Paper V

Visual Analysis of Zebrafish Behavior

TOBIAS PALMÉR[1], KALLE ÅSTRÖM[1], OLOF ENQVIST[2] AND PER PETERSSON[3]

[1] *Centre for Mathematical Sciences, Lund University*
[2] *Dep. of Signals and Systems, Chalmers University of Technology*
[3] *Dep. of Experimental Medical Sciences, NRC, Lund University*

Abstract: There is significant interest from the medical and neuroscience communities for studying the behavior of zebrafish larvae, particularly in the context of drug testing. In this paper, we propose a method for analyzing the behavior of zebrafish larvae in a high-throughput system using automatic image analysis. Specifically, algorithms for estimating the poses of the larvae in video recordings are presented. The pose estimation results are subsequently used to estimate kinematic parameters, segment the discretized movement into swim bouts (short bursts of movement) and to categorize the swim bouts into a number of classes. The distributions of bout classes are analyzed over time for different doses of amphetamine and apomorphine and compared to control groups. Preliminary results are presented indicating that the proposed method is able to measure the effects of pharmacological manipulations in the zebrafish larvae.

1 Introduction

Automatic image analysis for the purpose of analyzing the behavior of animals is a commonly used tool in for example medicine and biology [1, 2, 3, 4, 6, 7, 8, 9, 10, 11]. When used appropriately, it has the potential of greatly improving upon the quantity and/or quality of data gathering.

Zebrafish is a popular model animal in the field of medicine, but up until recently the commercially available automatic tracking softwares have tracked only one position per fish, consequently greatly limiting the behavioral analysis. However, some papers present more advanced whole-body tracking and analysis of curvature and bending motion of the fish. Fontaine et al. proposes a method for tracking zebrafish using a geometrical model of the fish [5], applied on videos recorded at 1500 fps with 1024×1024 pixels resolution. Tian et al. presents a method for tracking a single zebrafish in two cameras recording up to 250 fps at 640×480 pixels resolution [10].

Here, however, we choose a compromise between quantity and quality of tracking. A high-throughput system for automatic tracking and analysis of zebra-

fish larvae behavior is presented, using a skeleton-based model of the larvae. A single camera recording 300 fps at 640 × 480 pixels resolution is used for recording the behavior of 48 spatially separated larvae. The high frame rate enables the detection of quick movements and a higher temporal consistency than lower frame rates would, while the large number of animals makes the system practically useful for evaluating the effects of various stimuli.

2 Equipment and video data

The behavioral setup consisted of a 300 fps digital camera with a resolution of 640x480 pixels (Genie HM640, Teledyne DALSA, Waterloo, Canada) connected to a computer with video recording software (CamExpert v7.00.00.0912, Teledyne DALSA, Waterloo, Canada; LabVIEWâĎć 2011 v11.0, National Instruments, Austin, TX). In each experiment, 48 zebrafish larvae where placed in separate wells in two 24-well microtiter plates (Cat. No. 303002; Porvair Sciences, Leatherhead, UK) that were milled to a depth of 9 mm to reduce shadow and perspective artifacts. The output of the video recording system was typically as presented in Fig V.1.

3 Pose estimation

In this section, the procedure of estimating the poses of the zebrafish larvae from the input video files is described. First, the possible regions of movement of the larvae and a static background image are estimated. Then the poses of the larva are estimated in each regions using difference images created by subtracting the estimated background from the current frame.

3.1 Calibration

Each zebrafish larva is placed in a well in one of the microtiter plates and is constrained to move within the bounds of the well. Consequently, the tracking problem can be formulated as the independent tracking of 48 larvae. Furthermore, the fact that the wells are identically shaped is used for subsequent analysis of zebrafish larvae behavior, where the center of each of the wells is used to define a local coordinate system for each larvae.

The positions of the wells are estimated as follows. First, geometrical models of the microtiter plates are created using the known radii of the wells and the dis-

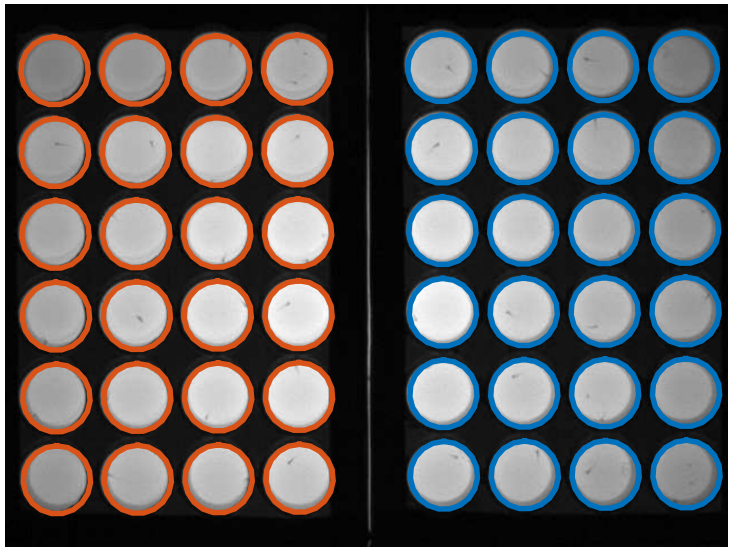


Figure V.1: An image from the video where zebrafish larvae have been placed in two 24 well microtiter plates. Superimposed projections of the geometric model of the microtiter plates are plotted in red and blue.

tances between them. Secondly, a thresholding procedure provides a number of circular blobs corresponding to the wells. Thirdly, affine transformations A_1 and A_2 from the models of the microtiter plates to the center of the blobs are estimated using RANSAC and linear least squares. Lastly, the regions that are subject to tracking is found by projecting the well-regions in the geometrical models of the microtiter plates. An example of calibration results is shown in Fig V.1. Additionally, tracked pixel coordinates of larvae can be transformed to the units in the geometrical model (e.g. millimeters) by applying the inverse of the estimated transformations.

3.2 Background estimation

The static background is estimated by first modeling each pixel (i, j) as belonging to one of two Gaussian distributions (with expected values F_{ij} and B_{ij}) and subsequently estimating the distributions. The distributions are estimated in an iterative procedure defined as follows. Initialization is provided by computing the mean value M_{ij} of each pixel (i, j) over a set of randomly selected frames. The

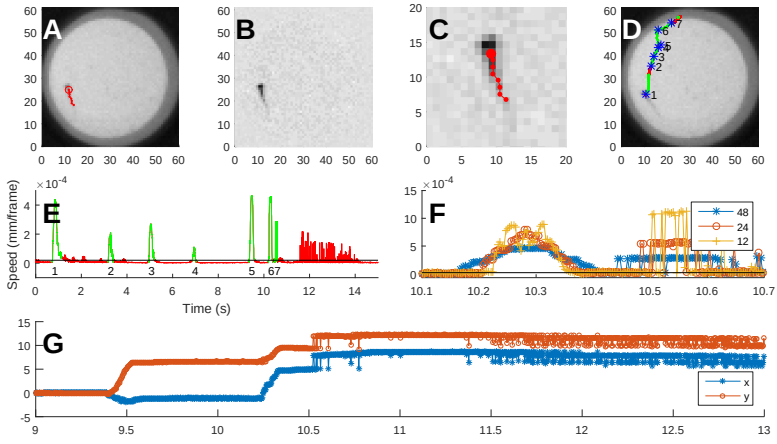


Figure V.2: Tracking and data. (A) shows a subsection of a video frame displaying the well of one larva, with the estimated larva pose superimposed in red. (B) shows the image in (A) after subtracting the background image. (C) shows a subsection of (B) with the larva pose superimposed in red. (D) shows the trajectory of the head position a larva superimposed on an image from the video. The trajectory is colored green at the points belonging to a swim bout, and red elsewhere. The blue asterisks represent the start of a swim bout numbered according to the number next to it. (E) shows the robustified speed function (see equation (V.3)) for the larva in blue and the threshold used for swim bout detection as the straight line in black. (F) shows the speed measure $v_k(t)$ from Eq (V.4) at the time around swim bout number 7 for $k = 12, 24, 48$. (G) shows the tracked x and y-coordinates of the head independently for a section of the time. Note the oscillatory noise that is successfully classified as not being part of an actual movement.

darker distributions are assigned the mean minus ϵ and the brighter distributions are assigned the mean plus ϵ, i.e. $F_{ij}^{(0)} = M_{ij} - \epsilon$ and $B_{ij}^{(0)} = M_{ij} + \epsilon$, for some small $\epsilon > 0$. For a random image I and for each pixel, the distance from the brighter distribution to the image and from the darker distribution to the image is compared, and the closest distribution is updated using moving average. After $N = 1000$ iterations, the estimates have usually converged and the procedure is finished. Since the larvae are dark and the background is bright, the background is defined as the estimated brighter distribution.

3.3 Pose estimation

The pose of a zebrafish larva is estimated by first finding the head position and then tracing the tail by adding points that are incrementally more distant from the head. Each point is estimated in sub-pixel resolution by linear (the tail) or quadratic interpolation (the head).

For this purpose, a Gaussian smoothing is applied on the difference image $F = I - B$ and the position of the maximal value provides an initial estimate of the head position in pixel-resolution. The use of the position of the maximum as the head position works sufficiently well due to the quality of the difference images (cf. Fig V.2(B-C)) and the shape of the larva. The initial estimate is improved upon in the x and y-directions independently by fitting quadratic polynomials to 5x1 and 1x5 neighborhoods around the initial position estimate. The sub-pixel estimate of the head position (x_0, y_0) is defined as the position of maximal value of the fitted polynomials.

The tail of the larva is traced by finding the maximal values in a set of N_{points} circles of increasing radii centered at (x_0, y_0), i.e.

$$(x_k, y_k) = \underset{(x,y)\in\Omega(x_0,y_0;r_k)}{\operatorname{argmax}} F(x, y), \quad k = 1, ..., N_{points} - 1, \tag{V.1}$$

where $F(x, y)$ is evaluated by linear interpolation of F using the four neighboring pixels of the point (x, y), $\Omega(x_0, y_0; r_k)$ is the set containing all points on the circle of radius r_k centered at (x_0, y_0) and $N_{points} - 1$ is the number of sought points on the tail.

Additionally, the foreground pixel intensities ($q_k = F(x_k, y_k)$) at the estimated points are stored and used as a quality measure. This means that for each larva and frame, there are $3N_{points}$ values stored:

$$(x_0, y_0, q_0, ..., x_{N_{points}-1}, y_{N_{points}-1}, q_{N_{points}-1}). \tag{V.2}$$

The pose estimation procedure is repeated for each larva in every frame, creating data on the form $(x_{i,j,k}, y_{i,j,k}, q_{i,j,k})$, where $i \in [0, N_{points} - 1]$ is the point index, $j \in [1, N_{frames}]$ is the frame number and $k \in [1, N_{larvae}]$ is the larva number.

4 Behavioral analysis

The zebrafish larvae move in discrete movements referred to as *swim bouts*. This section describes how the swim bouts are detected and normalized and resampled

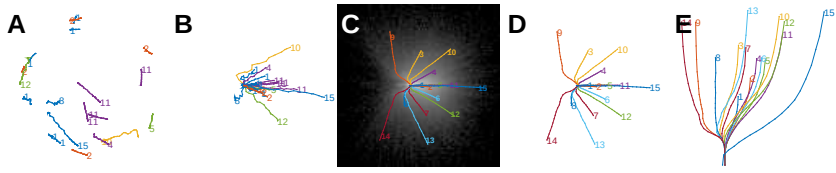


Figure V.3: Swim bouts and classification thereof. (A) shows the trajectories of 20 randomly selected swim bouts, with class numbers printed next to them. (B) shows the trajectories of the swim bouts from (A) after normalization and resampling (see Section 4.2). The element-wise logarithm of the density image of the full set of normalized and resampled trajectories is presented in (C), with the mean trajectories from (D) superimposed. (D) and (E) shows the (x, y) and (x, t) coordinates, respectively, of the mean trajectories (see Section 4.2). Note that the scale of (B), (C) and (D) are equal and the other scales are different.

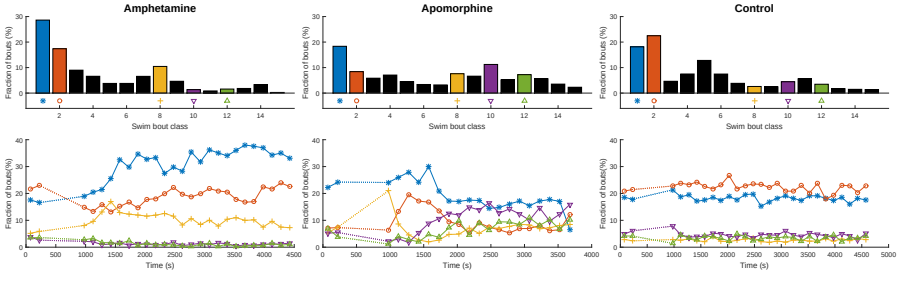


Figure V.4: Distribution of bout class observations. The top row shows the time-average distributions of bout classifications in each experiment (see Section 5). In the bottom row, bout class distributions are plotted in 150s non-overlapping time bins. The bars in the top row that are visualized over time in the bottom row have different colors, while the rest of the bars are black. In addition, the selected classes have a marker below their respective bars that is the same as in the time-dependent plot. The dotted lines denote the time during which drugs are induced. Note that the classes and their numbers are the same as in Fig V.3, but the colors are in general not the same.

to enable meaningful comparisons of different swim bouts. A clustering method is then applied on the data in order to create a set of cluster centers later used for classifying swim bouts. The classification results are used in Section 5 to analyze the behavior of the zebrafish larvae.

4.1 Swim bouts

A swim bout is defined as the time interval where the speed of the larva is greater than some threshold. Due to the presence of oscillatory tracking noise (see Fig V.2(G)), a speed measure that is robust to such noise is necessary. For this purpose, the robustified speed measure $\hat{v}(t)$ is defined as

$$\hat{v}(t) = \min_{k \in S} v_k(t), \tag{V.3}$$

where $v_k(t)$ is the average speed over a time window of width k and centered at t, i.e.

$$v_k(t) = \frac{FPS}{k} \left\| \begin{bmatrix} x(t + k/2) - x(t - k/2) \\ y(t + k/2) - y(t - k/2) \end{bmatrix} \right\|_2 \tag{V.4}$$

where FPS is the frame rate ($FPS = 300$ in this case). In this paper, the set of window sizes $S = 12, 24, 48$ provides a qualitatively good compromise between detecting true swim bouts and rejecting false swim bouts. Visualizations of Eq (V.3) and Eq (V.4) on real data are presented in Fig V.2(E) and Fig V.2(F).

A potential swim bout interval is defined as the time interval where the estimated robustified speed $\hat{v}(x, y, t)$ is larger than a threshold $v_{thr} = 0.2 \cdot 10^{-4}$ mm/frame. An example of the activity function and threshold can be seen in Fig V.2(E). This clearly supports the idea of treating the behavior of the zebrafish larvae as discrete swim bouts.

The potential swim bout intervals are post-processed by a combination of dilation, erosion and removing intervals that have low likelihood values (the q-value that was introduced in Section 3.3). This procedure has the effect that intervals that are close enough are merged and intervals that are too short or where the estimated tracking quality is too low are removed. Some potential swim bout intervals that are removed during post-processing can be seen in Fig V.2(E).

4.2 Swim bout classification

In order to compare different swim bouts, a way of measuring distances between swim bouts with different numbers of frames is needed. To achieve this, the trajectory of the swim bouts were equidistantly (in space) subsampled with K samples and time was added as a dimension. Thus the trajectory of each swim

bout was represented on the form

$$SB_i : X^{(i)} = \begin{bmatrix} x_1^{(i)} & x_2^{(i)} & ... & x_K^{(i)} \\ y_1^{(i)} & y_2^{(i)} & ... & y_K^{(i)} \\ t_1^{(i)} & t_2^{(i)} & ... & t_K^{(i)} \end{bmatrix} \tag{V.5}$$

Thus data for all swim bouts can be represented by column stacking the resampled swim bouts in the matrix $X = [X^{(1)}X^{(2)}...X^{(N_{bouts})}]$.

As indicated by Fig V.3(C), the space of swim bouts appears to be continuous without any apparent clusters. However, it is still meaningful to somehow bin the data. For this purpose, a clustering method (k-means is used in this paper) is applied to separate the data into groups. Before applying the k-means method on X, the data is normalized to remove the effect of the starting position and direction. Therefore, each swim bout is transformed by a rigid transformation with the effect that it starts in the origin and is headed to the right, as shown in Fig V.3. The k-means method subsequently produces $K = 15$ (defined empirically) groups that can be used for classification of swim bouts.

5 Experiments and results

The data set used in this paper contains three experiments on Amphetamine, two on Apomorphine and one control experiment.

Drug experiments were conducted on 10 days post fertilization (dpf) zebrafish larvae which were first placed in the wells and video recorded for 5 minutes, then drugs or placebo were injected in the wells followed by recording of videos for 50 or 60 minutes. In order to study the effects induced by various doses, the 48 larvae where split into 4 equally populated groups: a control group and low, medium and high dose groups.

Similarly, the control experiment was conducted on 10 dpf zebrafish larvae by recording for 60 minutes but without any interference in any subgroup of larvae.

A subset of the data is visualized in Fig V.4 by plotting the distributions of swim bout classifications over time (bottom row) and on average (top row). The time-dependent plots are created by splitting up time into a number of non-overlapping intervals of length 150 seconds. The distribution of swim bout classifications is computed for all swim bouts in each such interval. To increase readability of the figure, a subset of 5 swim bout classes has been selected for plotting. Here, only the data generated by the 48 larvae in the control experiment, the 36

larvae given the highest dose ($10\mu M$) of Amphetamine, and the 24 larvae given the highest dose ($50\mu M$) of Apomorphine, is presented.

6 Discussion

The preliminary data presented in Fig V.4 clearly shows that the proposed method is able to show the difference in behavior induced by Apomorphine and Amphetamine in the zebrafish larvae. For example, the Amphetamine-treated larvae tends to increasingly favor shorter movements (class 1). Furthermore, it is shown that there is a time-dependency in the induced drug effects, for example the sharp rightwards turn (class 8) is not very common in the pre-drug control time interval but is observed a lot more for a period directly after inducing Apomorphine.

References

[1] Cigdem Beyan and Robert B Fisher. “Detection of Abnormal Fish Trajectories Using a Clustering Based Hierarchical Classifier.” In: *BMVC*. 2013.

[2] Giuseppe Bianco, Vincenzo Botte, Laurent Dubroca, Maurizio Ribera d’Alcalà, and Maria Grazia Mazzocchi. “Unexpected regularity in swimming behavior of Clausocalanus furcatus revealed by a telecentric 3D computer vision system”. In: *PloS one* 8.6 (2013), e67640.

[3] Jonathan Cachat, Adam Stewart, Eli Utterback, Peter Hart, Siddharth Gaikwad, Keith Wong, Evan Kyzar, Nadine Wu, and Allan V Kalueff. “Three-dimensional neurophenotyping of adult zebrafish behavior”. In: *PloS one* 6.3 (2011), e17597.

[4] Mikael T Ekvall, Giuseppe Bianco, Sara Linse, Heiner Linke, Johan Bäckman, and Lars-Anders Hansson. “Three-dimensional tracking of small aquatic organisms using fluorescent nanoparticles”. In: *PloS one* 8.11 (2013), e78498.

[5] Ebraheem Fontaine, David Lentink, Sander Kranenbarg, Ulrike K Müller, Johan L van Leeuwen, Alan H Barr, and Joel W Burdick. “Automated visual tracking for studying the ontogeny of zebrafish swimming”. In: *Journal of Experimental Biology* 211.8 (2008), pp. 1305–1316.

[6] Tobias Palmér, Martin Tamtè, Pär Halje, Olof Enqvist, and Per Petersson. “A System for Automated Tracking of Motor Components in Neurophysiological Research”. In: *Journal of neuroscience methods* 205.2 (2012), pp. 334–344.

[7] TH Pinkiewicz, GJ Purser, and RN Williams. “A computer vision system to analyse the swimming behaviour of farmed fish in commercial aquaculture facilities: a case study using cage-held atlantic salmon”. In: *Aquacultural Engineering* 45.1 (2011), pp. 20–27.

[8] Maxwell B. Santana, Tobias Palmér, Hougelle Simplício, Romulo A. Fuentes, and Per Petersson. “Characterization of Long-Term Motor Deficits in the 6-OHDA Model of Parkinson’s Disease in the Common Marmoset”. In: *Behavioural brain research* 290 (2015), pp. 90–101.

[9] Concetto Spampinato, Simone Palazzo, Daniela Giordano, Isaak Kavasidis, Fang-Pang Lin, and Yun-Te Lin. “Covariance based Fish Tracking in Real-life Underwater Environment.” In: *VISAPP (2)*. 2012, pp. 409–414.

[10] Jing Tian, Amit Satpathy, Ee Sin Ng, Soh Guat Ong, Wei Cheng, Jean-Marc Burgunder, and Walter Hunziker. “Motion analytics of zebrafish using fine motor kinematics and multi-view trajectory”. In: *Multimedia Systems* (2014), pp. 1–11.

[11] Carlos E Vargas-Irwin, Gregory Shakhnarovich, Payman Yadollahpour, John MK Mislow, Michael J Black, and John P Donoghue. “Decoding complete reach and grasp actions from local primary motor cortex populations”. In: *The Journal of Neuroscience* 30.29 (2010), pp. 9659–9669.

Paper VI

Action Sequencing in the Spontaneous Swimming Behavior of Zebrafish Larvae - Implications for Drug Development

TOBIAS PALMÉR[+,1,2], FREDRIK EK[+,3], OLOF ENQVIST[4], ROGER OLSSON[*,3], KALLE ÅSTRÖM[*,2] AND PER PETERSSON[*,1,A]

[+] *shared first authorship*
[*] *shared senior authorship*
[a] *corresponding author*
[1] *Integrative Neurophysiology and Neurotechnology, NRC, Department of Experimental Medical Sciences, Lund University, Sweden*
[2] *Mathematics, Centre for Mathematical Sciences, Faculty of Engineering, Lund University, Sweden*
[3] *Chemical Biology & Therapeutics, Department of Experimental Medical Sciences, Lund University, Sweden*
[4] *Department of Signals and Systems, Chalmers University of Technology, Sweden*

Abstract: All motile organisms need to organize their motor output to obtain functional goals. In vertebrates, natural behaviors are generally composed of a relatively large set of motor components which in turn are combined into a rich repertoire of complex actions. It is therefore an experimental challenge to investigate the organizational principles of natural behaviors. Using the relatively simple locomotion pattern of 10 days old zebrafish larvae we have here characterized the basic organizational principles governing the swimming behavior. Our results show that transitions between different behavioral states can be described by a model combining a stochastic component with a control signal. By dividing swimming bouts into a limited number of categories, we show that similar types of swimming behavior as well as stand-stills between bouts were temporally clustered, indicating a basic level of action sequencing. Finally, we show that pharmacological manipulations known to induce alterations in the organization of motor behavior in mammals, mainly through basal ganglia interactions, have related effects in zebrafish larvae. This latter finding may be of specific relevance to the field of drug development given the growing importance of zebrafish larvae in phenotypic screening for novel drug candidates acting on central nervous system targets.

1 Introduction

In natural behavior, motor patterns are arranged into longer action sequences with different compositions depending on factors like the internal brain state of the subject, the behavioral context of the motor act or the functional goal of the behavior. It is not known how such action sequences are constructed and controlled by the nervous system but neuronal circuits of the basal ganglia are thought to play an important role [17, 29, 32]. Support for this notion comes for example from studies of locomotive behavior in fish where supraspinal structures involved in controlling different aspects of swimming behavior have been studied in some detail. In these investigations, a regulatory function of the basal ganglia has been identified upstream of regions in the mesencephalon and the more caudal brainstem, which in turn strongly influence locomotor output through descending pathways to spinal cord locomotor circuits (see e.g. [18, 23]). The basal ganglia are known to be highly evolutionary conserved across vertebrate species, and may in fact even have homologies with brain structures in arthropods. Interestingly, neurochemical similarities have also been established among different vertebrates. For example, the well-known modulatory role of monoamines in basal ganglia circuits in mammals has been shown to have a counterpart in fish [12, 13, 18]. Consequently, many of the basic anatomical, neurophysiological and neurochemical features of the central control of locomotor behavior appear to be very much alike across vertebrate species.

In mammals, natural behaviors typically involve smooth transitions between different motor patterns making it experimentally very challenging to analyze the higher order organization and sequencing of different movement components in any greater detail. In this respect, the locomotor patterns of fish larvae offers a special advantage, as their swimming behavior is highly discretized into isolated relatively brief bouts of locomotive activity. Thus, in order to study different patterns of action sequencing, we have here explored the discretized swimming behavior in zebrafish larvae. For this purpose we have developed an automated image analysis system that lets us analyze the spontaneous swimming behavior in larger groups of larvae in parallel with a level of detail that has previously only been possible in smaller groups using existing motion tracking systems [8, 22, 30, 39, 42]. Using this technology platform, we have here studied the swimming behavior in larvae at post fertilization day 10 (dpf 10) and show that the discretized locomotor behavior in fact has a higher order organization which is characterized by the sequencing of swimming bouts of a certain types. We also show that

pharmacological interventions that are known to alter the sequencing of motor behavior in mammals [26, 27], in part through interference with basal ganglia circuits [24, 31], have an analogous dose-dependent effect in zebrafish larvae.

2 Results

A novel method for automated tracking of spontaneous swimming Swimming behavior was first characterized based on a 65-min recording under dark light-conditions (using IR-illumination which is invisible to the larvae) in 10 days old zebrafish larvae. In the recording session, 48 larvae were placed in two polystyrene 24-well plates and greyscale digital images were captured at 300 frames per second with a spatial resolution of $\sim$ 16 pixels per mm^2. For every image, the curvature and position of the body was estimated through a two-step process. First, the area corresponding to the 2D projection of the body in the horizontal plane was estimated based on an algorithm classifying foreground objects from the stationary background by fitting of a bimodal Gaussian distribution of light intensities for each pixel throughout the recording period. Second, eight coordinate points spanning from head to tail tip, were fitted to the foreground area through an interpolation procedure based on the pixel intensity of concentric circles centered on the head (Figure VI.1A-C; see Materials and methods for details). By tracking of the body coordinate positions over a swimming episode, a $2 \times 8 \times N$ matrix was generated (where N corresponds to the number of frames collected over the duration of the swimming bout) which captures several detailed aspects of the motor behavior. These data were, in the following analyses, used to produce robust descriptions of the whole-body changes observed in horizontal position (Δr) and head angle ($\Delta \psi$; Figure VI.1D during the swimming episode. An example of normal swimming behavior as represented by these measures are shown for a 15s time sequence containing eight discrete swimming bouts in Figure VI.1E-G. A highly discretized locomotor behavior is evident in these representations. It is also apparent that although swimming episodes may give the impression of being quite stereotypic on a gross scale, more detailed analyses of speed and angle changes in individual bouts reveal that a wide range of different bout types appear to be present

2.1 Characterization of individual swimming bouts

High-resolution tracking of swimming behavior in several individuals in parallel makes it possible to quantitatively compare kinematic features in large populations where several thousands of events are analyzed to attain a sensitivity sufficient to reveal minor alterations in motor patterns associated with specific conditions. Building on the developed technology, $\sim$ 35000 swimming bouts recorded in 48 larvae were therefore next analyzed in further detail with respect to their kinematic features. As a first step, distributions of the most elementary kinematic measures, such as bout duration, distance and the cumulative angle change in either turning direction were plotted. These distributions, given the richness of the data sets, could potentially shed some light on aspects of the underlying control mechanisms. For example, if the likelihood of terminating a swimming event is unchanged throughout the execution of a bout (as modelled by a Poisson process where the neuronal network can be thought of as having no memory of when the ongoing bout was initiated) this would result in distributions that are close to exponential. If, on the other hand, the control mechanisms underlying termination of behavior instead resembles a random-walk process (that is, a gradual change in network state as modeled by stepwise movements towards or away from an event threshold denoting the network transition point which causes the termination of the ongoing behavior), distributions will deviate somewhat from strictly exponential distributions and display a relatively higher proportion of long-duration events (typically generated by bouts where the state of the network, as modelled by the random walk, has moved far away from the event threshold). Finally, overlaid on such random processes, motor commands could act to drive the state of the network towards behavioral transition, for example to achieve given behavioral goals. This type of combined process is schematically described in Figure VI.2A and can be mathematically modelled as an Ornstein-Uhlenbeck (OU) process [16, 40]. Indeed, fitting an OU model to the observed distance, duration and cumulative turning angle distributions proved to reproduce the observed behavior much better than a Poisson process or a pure random-walk model (Figure VI.2B-D and quantitative comparison in Supplementary Table VI.1). For duration, distance and angle change, respectively the driver parameters in the OU model were found to be 0.0069, 0.0021 and 0.0007/0.0006 (L/R), suggesting that the duration/distance of the swimming bout has a relatively stronger command component than the extent of turning which more resembles a random-walk process. This relatedness to random-walk processes observed for the control of the dura-

tion of these motor behaviors is intriguing, since it suggests that the swimming behavior of zebrafish larvae in certain aspect resembles the swimming behavior observed on a much larger scale in several species of adult marine predators [34]. In adult fish, this type of random walk or Brownian motion-like swimming behavior is often complemented with occasional very long straight swimming episodes, referred to as Levy flights [15]. If present in zebrafish larvae, Levy flights would however not be detectable in our recording set-up due to the limited size of each well (a necessary restriction to allow for high throughput).

To get a more in depth understanding of the joint translational and rotational behavior during swimming, the bout trajectories were subsequently analyzed in further detail. In agreement with previous reports (e.g. [6, 30]) we could confirm, via manual observation of the high-speed video recordings, that a few easily recognizable bout types were present and that some of these were observed much more frequently than others (e.g. scoots and routine turns compared to escape-like swims). However, larvae clearly displayed a much richer repertoire of movements than what could be summarized by only a few bout types. We therefore performed an automated classification of all the swimming trajectories recorded during a 65 minutes long control experiment in 48 larvae where trajectories were described in a coordinate system that was aligned to the starting position/body angle of the larva at the onset of each bout. Dividing the swimming behavior into 15 separate bout classes based on the shape of the trajectory in three dimensions (the two spatial dimensions of the horizontal plane and time) proved to result in a reasonable trade-off between adequate differentiation of observed differences in motor patterns and granularity of the data (Figure VI.2E-G; Bayesian information criterion suggested a model order selection in the range of 5-20 classes would be suitable, see Supplementary Figure VI.1). It should be noted however that these trajectory classes did not represent isolated behavioral types but rather sub-groups sampled from a rather smooth distribution (cf. Supplementary Figure VI.2; see Material and Methods for details on mathematical procedures). As expected, some bout classes were much more common than others - in particular short bouts with minor turning (Figure VI.2G). It is also worth noting that for the majority of the identified bout classes, a rapid early swimming phase was followed by a gliding phase containing relatively few active motor adjustments, in effect putting rather tight temporal constrains on the time window during which the network state can be actively modulated to control the total duration of the initiated behavior.

2.2 The higher order organization of swimming events

Having analyzed the detailed characteristics of individual bouts and modeled possible state transition mechanisms underlying the termination of motor behavior of each bout we next analyzed the inverse transition – that is, from immobility to locomotion. Also in this case, an OU process proved to model the observed distributions rather accurately (Figure VI.3A and Supplementary Table VI.1). However, compared to transitions from active to passive states the driver component in the OU model that helps controlling the duration of pauses between bouts of activity turned out to be an order of magnitude weaker (0.0001) indicating that neuronal state transitions responsible for initiation of swimming have a relatively stronger stochastic component.

Over more extended time periods, organisms generally cycle through different states of activity/alertness and sometimes display periods of rest [33, 36]. This would result in a change in the duration of stand-stills between bouts extending over a set of bouts. To search for higher order organization of stand-stills between bouts we investigated if larvae had a tendency to switch between either long (top 50% of the total distribution) or short (bottom 50%) waiting-times in a non-random manner. By analyzing the probability of observing waiting-times of the same type (long vs. short) in sequences of waiting-times it became evident that such a mechanism indeed seems to be at play – for example, after observing ten consecutive waiting-times of the same type the probability of switching to the opposite type in the next waiting episode is less than 0.2 (Figure VI.3B). Overall however, larvae frequently switch between long and short waiting times and the probability of repeating the previous type of waiting time for any observed consecutive pair of pauses was found to be close to chance level (50%).

Taken together these data suggest that an OU model with a relatively weak drive component can largely reproduce the observed statistical distributions. At the same time, similar waiting-times (long/short) have a tendency to cluster in time. Thus, for a more complete model of the swimming behavior a higher organization of stand-stills between swimming events will eventually have to be incorporated (for example by assuming that the starting point of each random walk in the OU model is determined by a slower secondary process).

2.3 The higher order organization of bout types

As mentioned in the introduction, a key rationale for the current study was to explore if action sequencing exists also in the comparatively simple locomotor behavior of zebrafish larvae. For example, in a similar way that pauses between swimming events appears to display a higher order organization different types of swimming bouts may also be changing in a state dependent manner. Indeed, Dunn and co-workers recently showed that younger zebrafish larvae (5-9 dpf) often display chains of either left or right turns and suggested that a specific neuronal population in the rhombencephalon may be responsible for this transitory bias towards one of the turning directions [10]. We could here confirm a propensity for repetitive turning pattern also in 10 days old larvae, as shown in Figure VI.4A, where the probability of observing a given degree of turning is plotted as a conditioned probability based on the direction of the previous turn (excluding bout pairs where the first turn is <10 degree [which make up $\sim$ 15% of the total number of bouts]). This turning bias equates to an observed probability of about 0.65 compared to 0.5 at chance. More importantly however, when we analyzed the probability of observing a repetition of anyone of the 15 bout classes defined above (Figure VI.2E-F) we observed an even stronger bias (probability= 0.22 compared to 0.16 at chance). To exemplify the widely differing probabilities of observing action sequences consisting of certain bout types, all permutations of pairs of bouts composed of class 1, 3 and 11 are shown in Figure VI.4B. Notably, all types of pairs made up of the same bout type were observed with a probability above chance level whereas several other unique permutations were observed much less frequently than what would be expected based on the probability of observing each of the two components independently. In fact, the tendency to repeat bouts of the same type is even more pronounced in the analysis of longer action sequences. For example if the same type of bout has been repeated ten times in a row the probability of observing the same type of bout out of the 15 possible classes is >0.8 (Figure VI.4C; chance level is indicated by the dashed line). Finally, although the bout classes defining the different bout types were arbitrarily chosen in our data-set and may not be directly transferable to other data, it is nevertheless interesting to note that certain action sequences are much more commonly observed than others also for some bout pairs involving different bout types (see e.g. 1->11 vs. 11->1 in Figure VI.4B).

Hence, we conclude that the swimming behavior of zebrafish larvae appears to be organized into longer sequences of actions where different bouts of motor

activity are performed consecutively. In particular, the same type of motor program is frequently concatenated into longer stereotypic action chains - apparently reflecting specific states of the motor control networks.

2.4 Pharmacological manipulations reveal mammalian-like motor control principles

The intricate arrangement of locomotion components revealed in the current analyses, involving parsing of motor behavior into complex sequences, clearly suggests that the seemingly simple swimming behavior of zebrafish larvae most likely contain the basic components needed to allow for comparisons to action sequencing in mammalian species. This finding may have important implications, as zebrafish larvae could therefore become a valuable model system for translational research aimed at developing new therapeutic strategies to treat motor dysfunctions in humans. Following the characterization of normal swimming behavior we therefore next performed a second set of experiments aimed at detecting and describing the changes in motor behavior induced by systemic pharmacological manipulations. To this end, three different drugs which are comparatively well-characterized with respect to their behavioral effects in mammals were evaluated. Two of these drugs were selected to target the monoaminergic system (apomorphine and amphetamine). Monoamines are believed to have a key role in the regulation of goal directed behavior, partly through action on neurophysiological processes of the basal ganglia. Apomorphine is a non-selective dopamine agonist which principally activates dopamine receptors of both the D1 and D2 sub-type. Amphetamine, on the other hand, has a broader effect on monoaminergic signaling but is thought to primarily stimulate the dopaminergic and noradrenergic system by various mechanisms that contribute to an increase in the synaptic levels of these transmitter substances. The third drug, MK-801, instead directly targets the glutamatergic system by acting as a non-competitive antagonist on NMDA-receptors. This drug is relevant in the context of motor control for the reason that NMDA-antagonists, when administered in lower doses, has been shown in rodents to induce a moderate form of hyperactivity that includes a shift in the pattern of locomotion towards longer uninterrupted bouts of movements (see e.g. [7, 14]).

All experiments were repeated either two or three times in different groups of animals resulting in a total of 24 or 36 larvae in each treatment group. In the case of apomorphine two different dose regiments were explored based on the

previously reported complex dose-response patterns [11]. For MK-801, slightly older larvae were used (15 dpf) since contradictory responses have previously been reported for this drug in younger animals [25, 28].

All drugs resulted in dose dependent changes in swimming behavior. For example, the number of bouts displayed per unit time decreased throughout the experiments, except for high dose apomorphine which showed a more complex biphasic pattern (as previously reported by e.g. Irons et al. [20] and Ek et al. [11]; Figure VI.5A-D). Notably, for MK-801 the observed reduced locomotive activity (swimming distance per bout was also reduced, by approximately 1/3 at the highest dose) clearly contradicts the previously reported effects in rodents and instead bears a resemblance to the motor effects reported in primates [4]. More interestingly, the sequencing of motor behavior was also found to be altered in a drug and dose dependent manner. An example of this is shown in Figure VI.5E-H, where the relative overrepresentation of a specific bout pair is plotted as a function of time for the different experiments (i.e. the increased probability of observing that specific bout pair in relation to the expected number of observations based on the overall observation frequency of the constituent bout type, i.e. $P(i,j) - P(i)P(j)$).

Because more complex combinations of kinematic features could be affected by the pharmacological manipulations we next expanded our analyses to include a multivariate approach for evaluating the behavioral changes induced by the different drugs. Accordingly, all the different measures presented above (Figure VI.1–VI.4), were used to construct a large kinematic space based on a total of 435 features extracted from the swimming behavior (for a complete summary of these 435 features and estimates of their relative interdependency see Supplementary Table VI.2 and Supplementary Figure VI.3). First, to get an intuitive understanding of the data, visualizations were constructed showing the global behavioral state of each treatment group as a function of time through principal component analysis (In Figure VI.5I-K this is exemplified by the projection of the 435-dimensional state vector onto the first three principal components [explaining 76% of the total variance]). The relatively smooth trajectories showing gradual rather than abrupt state changes between data sets sampled at adjacent time points suggest that this approach generates robust state descriptions. Moreover, the test sensitivity also appears to be good since the different treatment groups are distinguishable even when the space is reduced to only four dimensions. Second, in order to more directly compare the specific effects of individual drugs, dose-response relations were

analyzed for each drug. To this aim, the projection of each sample point in time in the 435-dimensional state space onto the vector defining the mean difference between the behavior of the group of animals that had received the highest dose of each drug and the saline treated control group in the same experiment, <highest dose – control> were calculated (cf. [5]). In this metric, the normal swimming behavior of the control group will on average correspond to the value zero whereas the average behavior displayed by the group of larvae treated with the highest dose will correspond to the value one, regardless of what type of behavioral changes that are induced by each specific drug (Figure VI.5L-O; cross-validations of these data are summarized in Supplementary Table VI.3). It was evident that robust and stable dose-dependent alterations of swimming behavior was induced by each of the drugs evaluated, further corroborating the methodological approach. However, the changes observed in global swimming behavior involve complex combinations of kinematic features and therefore provide limited information on the specific physiological changes underlying the unique behavioral states. As a final step we therefore summarized to what extent the 435 different features were affected by the various pharmacological manipulations by plotting their relative contribution to the difference vector <highest dose – control> (Figure VI.5P-S; in this representation the relative contribution of each feature is denoted by the distance from zero). Interestingly, and as noted for the example features above, both very basic changes in swimming behavior - such as the average speed and duration of bouts - and several more complex changes in the pattern of action sequencing - such as the relative frequency of observation of specific waiting times or bouts pairs – proved to be sensitive to the pharmacological manipulations. Hence, potentially allowing for also more in depth qualitative analyses of behavioral alterations.

3 Discussion

The highly discretized swimming behavior of zebrafish larvae provides a unique opportunity to study patterns of action sequencing. It is however a significant experimental challenge to document and describe the motor repertoire of hundreds of freely behaving millimeter-size individuals. Through the development of an automated high-resolution tracking system we have here managed to overcome this complication and have characterized the swimming behavior with a greater precision in larger groups of zebra fish larvae than what has been obtained with

previously existing systems. Based on these larger data sets it was possible to evaluate how well theoretical models of possible network mechanisms underlying behavioral transitions could reproduce actual swimming behavior. Most importantly, we could also provide direct evidence for action sequencing which opens up for further studies on more complex aspects of motor control in this species. Finally, our results clearly show that zebrafish larvae are sensitive to pharmacological manipulations of motor control networks in the same way as mammals and hint at an important role of the basal ganglia in the control of motor behavior given the clear sensitivity to monoaminergic manipulations with respect to sequencing of actions [9, 41].

Linking mechanisms of motor control and the selection and sequencing of actions in mammals to the zebrafish potentially opens up for new studies on genetic influences on behavior since zebrafish is a very widely used genetic model system. The findings in the present study, showing evidence of complex sequencing of actions, support such a link between vertebrate species and points to a key role of the basal ganglia in action sequencing (cf. [1, 35]). In addition, a more general use of the technology developed herein could have a great value also outside the field of basic neuroscientific research as zebrafish is becoming more frequently used for in vivo screening in drug discovery assays [2, 3, 19, 37]. Diseases affecting the central nervous system (CNS) is a rapidly growing concern for societies all around the world. At the same time, the rate of progress in the search for new treatment options is currently disappointingly slow. It has been argued that a major reason behind this troublesome situation is the fact that early screening of new drug candidates has only to a limited extent been carried out in intact animals, while reduced model systems such as in vitro assays have so far shown low predictive value for clinical treatment of CNS disease [38]. On the other hand, assays based on in vivo screening have proven somewhat more successful in terms of the number of new drugs developed but the very high costs associated with large scale studies in standard laboratory animals like rats and mice unfortunately limits the practical value of this approach. For this reason, simpler animal models have been evaluated with the aim of partially substituting experiments in mammals. In particular zebrafish larvae have attracted attention as a possible in vivo model, allowing for behavioral monitoring of relatively large groups of animals, in parallel, following pharmacological interventions.

A potential limitation when using fish larvae instead of rodents in behavioral analyses is, however, a somewhat reduced complexity in the motor repertoire. It is

also important to keep in mind that environmental factors, including for example the size and color of the individual wells, can influence the behavior displayed. Consequently, the primary behavioral read-out, to date, has been relatively gross measures of general swimming activity displayed per unit time [21].

In this study we show how more advanced procedures for imaging and tracking of motor behavior enables a much more detailed analysis of movement components in swimming bouts and the characterization of higher order organization of swimming bouts in different behavioral states.

From observations in humans and in mammalian models systems of disease, it is known that a wide range of behavioral changes are associated with disease conditions or may occur as a consequence of exposure to drugs acting on the CNS. Various types of abnormal motor behaviors and motor dysfunctions can be categorized as for example: bradykinesia, dystonia, dyskinesia, chorea, stereotopies, ataxia, tremor, myoclonus etc. Clearly, this wide range of different types of motor symptoms cannot be accurately quantified by a single measure of the overall amount of motility displayed. Instead, it is the relative frequency by which certain behaviors are performed over others, or more detailed changes in in the organization of sub-components of the motor repertoire that need to be assessed in order to distinguish these types of abnormalities in the motor behavior. Consequently, if behavioral screening in zebrafish is to become a useful tool for drug development it is important to provide more sensitive techniques for screening of behavioral changes using a system for automatic tracking of swimming behavior. The sensitivity and reliability of our developed tracking system that was here evaluated under diverse experimental conditions, including pharmacological manipulations applying substances that are known to induce changes in the organization of behavior in humans, proves the feasibility of this approach. We therefore believe that the technology and biological findings presented herein should have important implications both for future studies of motor control systems and for researchers interested in improving the technology for in vivo screening of novel compounds in drug development.

4 Materials and methods

4.1 Animals

This study was conducted in accordance with the national legislation of Sweden and the European Community guidelines for animal studies. All procedures were

approved by the ethical committee in Malmö-Lund (Permit, M116-12).

The zebrafish larvae used in this study were from intercrosses of the wild-type AB strain. Embryos were collected and raised in a 14:10-hour light/dark cycle at 28.5°C on petri dishes containing E3 embryo medium (5 mM NaCl, 0.17 mM KCl, 0.33 mM CaCl2, and 0.33 mM MgSO4) in an incubator to 5 days post-fertilization (dpf). At the age of 5 dpf, the larvae were transferred into 0.8 L aquaria and placed in a recirculating system held at 26±1.5°C (Aquaneering, Inc., San Diego, CA) where feeding was initiated. Larvae were fed with a commercial larval diet, ZM000 (ZM Fish Food & Equipment, Winchester, UK), three times daily until the age of 10 dpf. Behavioral experiments were conducted at 10 dpf (and 15 for MK-801 experiments). The age of the zebrafish larvae was chosen to ensure a developed blood-brain barrier, which is an important feature when evaluating potential pharmaceuticals for CNS diseases.

4.2 Experimental setup and video recordings

Each individual experiment was performed using 48 zebrafish larvae, 2x24 well microtiter plates (Cat. No. 303002; Porvair Sciences, Leatherhead, UK) milled to a depth of 9 mm to reduce shadow artifacts and with white walls to increase contrast between larvae and background and to prevent larvae in adjacent wells from acting as visual stimuli. The behavioral chamber consisted of a 300 fps digital camera (Genie HM640, Teledyne DALSA, Waterloo, Canada) connected to a computer set up with video recording software (CamExpert v7.00.00.0912, Teledyne DALSA, Waterloo, Canada; Labview™ 2011 v11.0, National Instruments, Austin, TX. To maintain the environment in the wells at 28°C the microtiter plates were placed parallel to each other in a water bath containing a temperature control unit (Neoheater 25 W thermostat, AQUAEL, Warsaw, Poland). The microtiter plates were positioned on top of a light box, containing LED strips (SMD5050 flexible infrared 850 nm tri-chip). Prior to each experiment, larvae were transferred to the microtiter plates containing 1 ml of E3 medium and all individuals were observed for abnormal swimming behavior and body deformities. Damaged individuals were removed and replaced. The experiments were performed in darkness (using IR-illumination). The zebrafish larvae were habituated for 45 min before the experiment. During the experiment, video recordings were obtained in 5 min time periods in order to be able handle the large amount of data (>5GB/min), creating brief interruptions ($\sim$ 1s) during data transfer to storage devices.

4.3 Pharmacology

For the amphetamine experiments, larvae were analyzed in 4 treatment groups (0.1, 1.0, 10 µM and control). The experiment was performed in triplicate including in total 36 larvae per group. For drug preparation compound (D-amphetamine sulfate, 2813, Tocris Bioscience) was first dissolved in 100% dimethyl sulfoxide to generate a stock solution (10 mM). The stock solution was then diluted in E3 medium to 500 µM, 50 µM and 5 µM and 20 µL of these solutions were added to the wells. The control group was given 20 µL of a 5% DMSO solution corresponding to a final concentration of 0.1% DMSO in the well. For the MK-801 experiments, larvae were analyzed in 4 treatment groups (1, 5, 20 µM and age matched control). The experiment was performed in duplicate including in total 24 larvae per group. For drug preparation compound ((+) MK-801 hydrogen maleate, M107) was first dissolved in 100% dimethyl sulfoxide to generate a stock solution (10 mM). The stock solution was then diluted in E3 medium to 1000 µM, 250 µM and 50 µM and 20 µL of these solutions were added to the wells. The control group was given 20 µL of a 5% DMSO solution corresponding to a final concentration of 0.1% DMSO in the well. For the apomorphine experiments, larvae were analyzed in 8 treatment groups divided in a high dose (10, 25 and 50 µM) and a low dose (0.1, 0.2 and 0.5 µM) treatment paradigm. The high dose paradigm was performed in duplicate and the low dose paradigm in triplicate experiments including in total 24 and 36 larvae, respectively. For drug preparation compound (A4393, Sigma-Aldrich, St. Louis, MO) was first dissolved in 100% dimethyl sulfoxide to generate a stock solution (10 mM). The stock solution was then diluted in E3 medium to 2.5 mM, 1.25 mM, and 500 µM (high), or 25 µM, 10 µM, and 5 µM (low).

4.4 Tracking of swimming behavior

The swimming behavior of the zebrafish larvae was analyzed by first tracking the positions of the larvae in video recordings by an automatic image analysis algorithm and then inputting the tracking results into the behavioral analysis algorithms described below. This section describes the automatic image analysis application.

Given a video sequence of zebrafish larvae recorded using a static camera and static lighting conditions, the static background is computed by modelling each pixel at each point in time as belonging to one of two Gaussian distributions,

where the brighter distribution defines background and the darker distribution defines foreground. The distributions are estimated using a moving average estimation algorithm, described in the following. Background and foreground are modeled using normal distributions for each pixel with expected values B and variances S for the background and $\hat{B}$ and $\hat{S}$ for the foreground. The model parameters are initialized by drawing 100 randomly selected frames and computing the mean M and variance V of them for each pixel, then setting $B^{(0)} = M + \sqrt{V}$, $\hat{B}^{(0)} = M - \sqrt{V}$ and $S^{(0)} = \hat{S}^{(0)} = V$. For 1000 additional randomly selected frames $I^{(k)}$, the probability that $I_{ij}^{(k)} \in \mathcal{N}(B^{(k)}, S^{(k)})$ is compared to the probability that $I_{ij}^{(k)} \in \mathcal{N}(B^{(k)}, S^{(k)})$. All pixels (i, j) that are estimated to be more probable to belong to the background and the corresponding elements of B and S are updated with the updating coefficient α as follows

$$\begin{aligned} B_{ij}^{(k+1)} &= \alpha B_{ij}^{(k)} + (1-\alpha) I_{ij}^{(k)}, \\ S_{ij}^{(k+1)} &= \alpha S_{ij}^{(k)} + (1-\alpha)(B_{ij}^{(k+1)} - I_{ij}^{(k)})^2. \end{aligned} \tag{VI.1}$$

The foreground parameters $B^{(k)}$ and $S^{(k)}$ are updated analogously for the pixels that are estimated to be more probable to belong to the foreground. The resulting expected value of the background distributions define the background image.

Given an image I and the estimated background image B, the difference image $D = B - I$ is computed. Since the pixels corresponding to zebrafish larvae are darker than the background, D is an image in which higher values are more likely to correspond to zebrafish larvae. The larva head is assumed to be at the position of the maximal value (robustly measured given a Gaussian kernel of size 5×5 pixels) of D in each well. An initial value of the head position is given by finding the position of the maximum of D after applying a Gaussian filter. The initial value is improved to subpixel-precision by quadratic interpolation. Given the estimated head position, the tail of the larva is found by the positions of maximal intensity at 7 circles of increasing radii, centered at the head.

4.5 Extraction of swim bouts

As previously shown, the zebrafish larvae appear to move in short bursts of movement (denoted *swim bouts*; see Figure VI.1E, F). Therefore, an important task in the analysis of zebrafish larvae behavior is to segment the tracking data that corresponds to swim bouts.

This is done by estimating the speed of the larvae and defining the time intervals in which the larvae is moving faster than some threshold as swim bouts. In order to make the swim bout segmentation robust to oscillatory tracking errors, the following speed measure is used

$$v(t) = \min_k v_k(t) \tag{VI.2}$$

where

$$v_k(t) = \frac{\|(x(t+k), y(t+k)) - (x(t), y(t))\|_2}{k} \tag{VI.3}$$

In other words, the average speeds over various time windows k are first estimated, and then the minimal value over all time windows is used in subsequent swim bout segmentation. In this particular application (framerate, resolution, fish size, error size, etc.), the time window sizes $k = 1, 12, 24, 48$ have been defined empirically.

An example of applying this robust speed function on tracking data is shown in Figure VI.1F. An initial segmentation of swim bouts can be performed by simply applying a threshold value ($v_{threshold}$) on the robust speed estimates, i.e. the points in time t for which $v(t) > v_{threshold}$ are classified as swim bouts. As is common in this type of application and segmentation applications in general, there are sometimes uncertainties to resolve after applying the threshold and performing the initial segmentation. Firstly, initial swim bouts that are too short (< 0.03 s) are removed and secondly, swim bouts that contain an estimated speed component that is too large (> 5 m/s) are removed. Thus a set of swim bouts can be constructed, each on the form $(x_{1k}, y_{1k}, \ldots, x_{8k}, y_{8k}, k)$, where k is the frame number. In this notation, (x_{1k}, y_{1k}) corresponds to the head and (x_{8k}, y_{8k}) corresponds to the end of the tail.

4.6 Classification of swim bouts

The process of classifying swim bouts as belonging to one of K classes is described here. First, swim bouts from a control experiment are normalized and subsampled. Secondly, the k-means algorithm is applied to separate the data into K groups and compute the group centers (i.e. the mean trajectories of the classes). Thirdly, a swim bout is classified as belonging to the class for which the distance from the corresponding mean trajectory to the normalized and subsampled swim bout is minimized.

To enable meaningful comparison of swim bouts, swim bouts are normalized with respect to starting position, direction and time. The tracking data of each swim bout is rotated and translated such that it starts at the origin at time zero and is facing towards the positive x-axis. In practice, this means that a rigid transformation T is sought such that the transformed head coordinate, $\hat{x}_{11} = Tx_{11}$, of the first frame in the swim bout is at the origin and the transformed tail coordinates, $\hat{x}_{i1} = Tx_{i1}, i = 2, \ldots, 8$, lies approximately (minimizing the sum of squared distances) on the positive x-axis. Then all subsequent frames are transformed using the estimated T. In the current analyses, body curvature was not specifically analyzed. As a consequence, the tail positions were discarded after this step keeping solely the head position. Due to the variety in duration of swim bouts, a method for measuring similarities between tracking data of different lengths is necessary. This is provided by subsampling each swim bout by $M = 20$ (empirically defined) points that are equidistant in space, which effectively removes the time-dependency of the space dimensions while keeping it in the time-dimension. Note that the velocity information is not removed by this interpolation since the time-dimension is kept intact. Thereby, each swim bout is transformed to the normalized and subsampled form $\{(\hat{x}_k, \hat{y}_k, t_k), k = 1, \ldots, M\}$ from which they are compared.

The trajectories of N normalized and subsampled swim bouts are stacked in a matrix

$$X = \begin{bmatrix} X_1 & \ldots & X_N \end{bmatrix} \tag{VI.4}$$

where each X_i is the column stacked coordinates from swim bout i, i.e.

$$X_i = \begin{bmatrix} x_1 & y_1 & t_1 & \ldots & x_M & y_M & t_M \end{bmatrix}^\top \tag{VI.5}$$

To enable meaningful comparisons in the multidimensional space of different quantities (time and space), the matrix X is normalized as follows. All x-coordinates are normalized by subtracting the mean and dividing by the estimated standard deviation of all measured x-coordinates. The y and t-dimensions are normalized analogously.

The k-means algorithm with the Euclidean norm as metric is then applied on the normalized X to separate the data into K groups, and the mean trajectory of each group is then computed and reshaped back to the form $\{(x_k, y_k, t_k), k = 1, \ldots, M\}$. The created mean trajectories are subsequently used for classification of swim bouts by measuring the Euclidean distance from each normalized and

subsampled swim bout to the mean trajectories. A swim bout is then classified as belonging to the class with the closest mean trajectory. An example of estimated mean trajectories for $K = 15$ is shown in Figure VI.2E-F, and the distribution of the population of classes is shown in Figure VI.2G.

4.7 Extraction of kinematic features and higher order organization features

The definitions of the kinematic features presented in Figure VI.2B-D are introduced here. The distance of a swim bout is defined as the sum of Euclidean distances between all consecutive pairs of frames in the swim bout, i.e.

$$d = \sum_{k=1}^{n-1} \|(x_{k+1}, y_{k+1}) - (x_k, y_k)\|_2. \tag{VI.6}$$

Note that to increase robustness to noise, (x_k, y_k) is pre-filtered using a median filter of length 5 (empirically defined). The duration of a swim bout is defined as the length of the interval in which the larva travels from 5% of the total distance to 95% of the total distance, where the method of measuring distance introduced above is used. The angular change of a swim bout is defined as the second encountered extreme angle minus the first encountered extreme angle, i.e. a swim bout in which the minimal angle -34 degrees is attained at $t = 0.15$s and the maximal angle +13 degrees at $t = 0.05$s is considered to have an angular change of -47 degrees.

The bout classes introduced in the previous subsection are also used for analyzing swimming behavior. Tracking data for each zebrafish larva is converted to the form $\{(c_k, t_{0,k}, t_{1,k}), k = 1, \ldots, N\}$ where c_k is the estimated class of swim bout k, $t_{0,k}$ and $t_{1,k}$ are the start and end times of the swim bout, respectively, and N is the number of swim bouts. The vector $w_k = t_{0,k+1} - t_{1,k}$ defines the inter-bout waiting times for $k = 1, \ldots, N$, and the distribution of waiting times for a control experiment is presented in Figure VI.3A. A threshold w_{short} is applied on the vector w to classify each waiting time as short or long. Here, the threshold is individually set to the median of w_k for each larva and thus creates equally populated groups of short and long waiting times for each larva. Higher order organization of waiting times can be analyzed by considering the conditional probability that w_k is short given that the previous N waiting times have been short, i.e. $p^N_{short} = P(w_k < w_{short} | w_{k-1} < w_{short}, \ldots, w_{k-N} < w_{short})$ Fi-

gure VI.3B shows an estimate of p^N_{short} as a function of N together with the theoretical probability for the memoryless process w_k.

The sequencing of swim bouts is analyzed in Figure VI.4B-C. The probability that an observed swim bout subsequence (c_k, c_{k-1}) consists of movement class A followed by movement class B, i.e. $P(c_k = B, c_{k-1} = A)$, is shown for a few examples of (A, B) in Figure VI.4B. Also the probability that both swim bouts in a subsequence (c_k, c_{k-1}) are of the same class is shown as (X, X) in the figure. The conditional probability that a swim bout is of the same class as all the N previous swim bouts, given that all N previous are of the same class, is presented in Figure VI.4C.

4.8 Dose-response tests

The behavioral alterations induced by drug administration is investigated by quantizing the kinematic features and higher order organization features presented above. Each of the features that follows is evaluated independently for each zebrafish larva and each time period. The values given by the following features are entered into a feature matrix F of size $M \times N \times K$, where M is the number of unique features, N is the number of time bins and K is the number of fish. First, the distributions of swim bout duration, distance, turning, inter-swim bouts waiting times and swim bouts classes are estimated and normalized in 10, 10, 20, 15 and 10 bins, respectively. The sequencing of inter-swim bout waiting times is represented by all possible conditional probabilities of order 5 or lower. The sequencing of swim bout classes is entered as the conditional probabilities of order 2. For each drug type, the mean vector from the control group to the highest group was computed and subsequently used to define the dose-response vector. For each dose in the drug group and each time bin, the mean vector was computed and projected on the dose-response vector. The results are shown in Figure VI.5L-O.

5 Statistical tests

Statistical tests used are specified in the main text in the context they were used.

Acknowledgements

The authors wish to thank Anna Fornell and Mauritz Brorsson for technical assistance in the construction of the recording set-up and are also very thankful to Pär Halje for valuable discussions on the mathematical formalization of the Ornstein-Uhlenbeck process and to Peder Svensson, Joel Sjöbom and Pär Halje for thoughtful comments on the manuscript.

Competing interests

The authors declare no competing financial interest.

Figures and legends

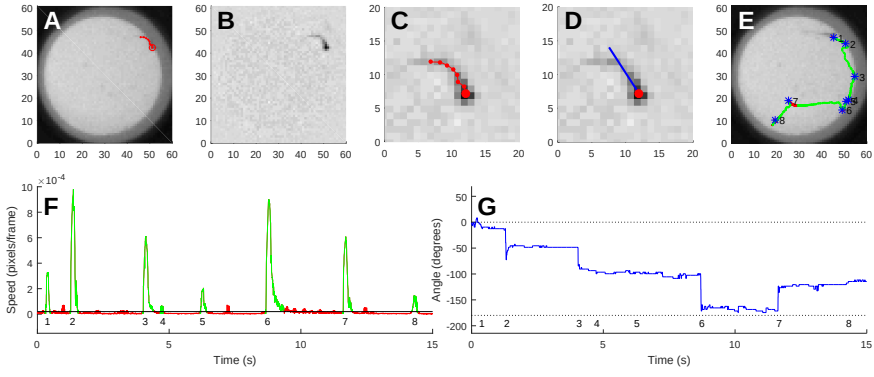


Figure VI.1: Automated procedures reveal detailed kinematics of the highly discretized swimming behavior.

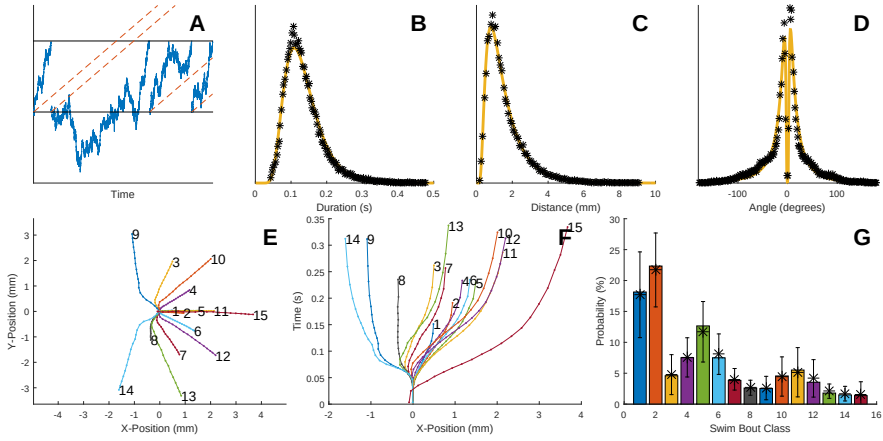


Figure VI.2: Characterization of swimming bouts based on kinematic features.

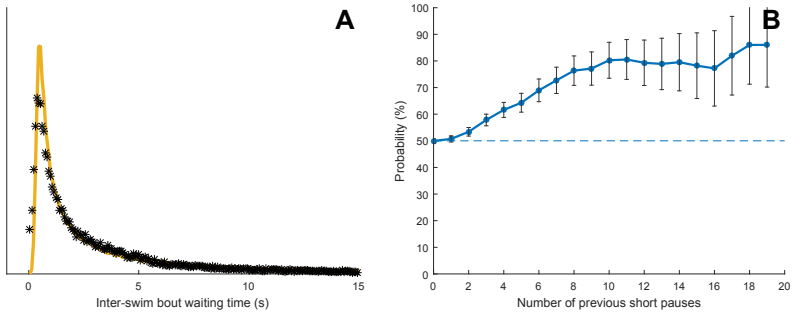


Figure VI.3: The organization of inter-bout waiting times.

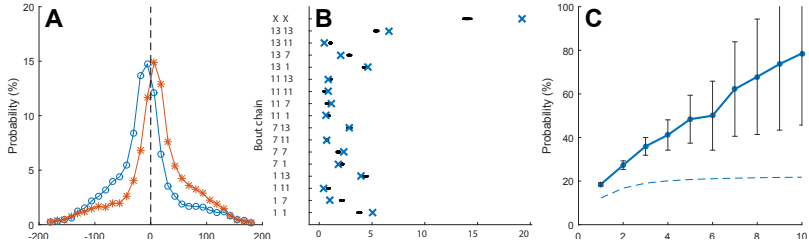


Figure VI.4: The organization of swimming behavior.

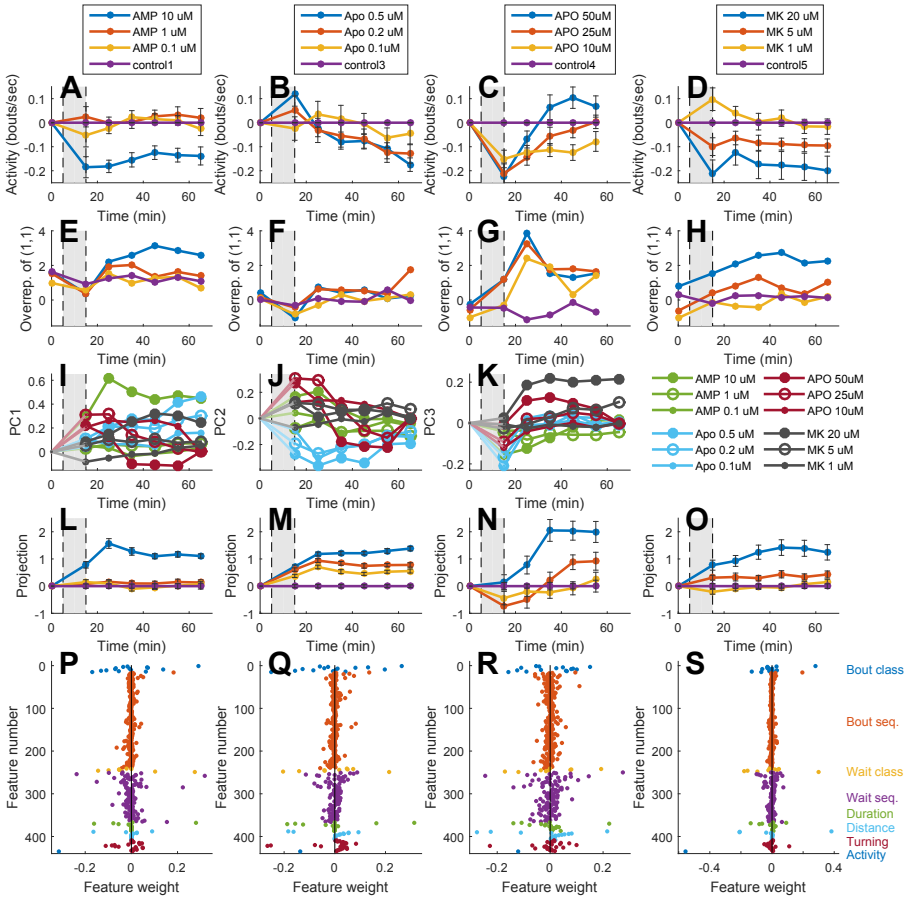


Figure VI.5: Pharmacological manipulations induce specific changes in swimming behavior involving both simple kinematic features and action sequencing.

Figure VI.1

[ht!] Automated procedures reveal detailed kinematics of the highly discretized swimming behavior. (A) Example image frame from high-speed video recording of spontaneous swimming behavior. Each of the 48 larva, recorded in parallel, is kept isolated in a 15mm well (red pixels overlaid in the gray scale image denote the body of the larva as automatically classified by the system). (B) The corresponding image as shown in (A) after background subtraction of pixels lacking temporal dynamics. (C) Close-up of the foreground image with superimposed tracking data, where the large dot marks the putative head. Note the improved contrast compared to (A) following background subtraction. (D) Example episode of spontaneous swimming illustrated by the tracked trajectory of the head position during a 15s time interval (first frame in the sequence is shown as background). The trajectories of single swimming bouts are colored in green. The blue asterisks represent the starting point of each detected bout during the 15s period (numbered from 1 to 8; red pixels mark a section of the trajectory where movement speed was below the threshold used for classification of swimming behavior and thus not considered part of a bout). (E) Swimming speed plotted as a function of time for the 15s period shown in (D), [numbering of individual bouts as in (D); only swimming episodes with a speed above threshold level (black line) above a minimum duration were classified as bouts (#1-8)]. Note that there are some time intervals that are not classified as swim bouts even though the estimated movement speed is larger than the threshold. This is because of the removal of intervals shorter than 0.03s. (F) Allocentric head angle (East = 0 degrees) plotted as a function of time during the 15s period shown in (D). Bout numbering as in D-E.

Figure VI.2

[ht!] Characterization of swimming bouts based on kinematic features. (A) Schematic representation of the underlying assumptions of the Ornstein–Uhlenbeck model. Transitions in behavior occur when the state of the network controlling the behavior passes a certain threshold (upper line). The state of the network is at any point described by a random-walk process which will eventually lead to spontaneous transitions events. In addition, a command signal that actively pushes the state towards the threshold may be present (indicated by the dashed lines). (B-D) Distributions of kinematic features extracted for each bout (total $n \approx 120000$),

(B) bout duration, (C) bout distance and (D) turning angle. Yellow lines denote fitting of the Ornstein–Uhlenbeck model to experimental data and asterisks show binning density. (E-F) Classification of swimming trajectories reveals differences between swimming behavior observed in individual bouts. (E) Trajectories representing the mean trajectories of 15 bout types identified by the algorithm are shown in the horizontal plane (x-y). (F) The trajectories in (E) shown with time from bout on-set on the y-axis (x-t). The apparent earlier onset of type 15 is caused by data subsampling (see Methods). (G) The relative frequency of the number of observed bouts of each type (class code as in E-F; showing that shorter and straight bouts tend to be more common). Whiskers denote mean and SD of the distribution over all individuals – note the relatively small SD and that the pooled probability (bars) closely matches the average over individuals (asterisks) indicating a small inter-individual variance.

Figure VI.3

[ht!] The organization of inter-bout waiting times. (A) The distribution of inter-bout waiting times (total $n \approx 120000$), yellow line denotes fitting of the Ornstein-Uhlenbeck model to experimental data and asterisks show binning density. (B) The probability of observing a short waiting time, given that the previous waiting times were also short increases with the number of similar waiting times observed in a row (short= the shortest 50% of the total amount of waiting times observed in each larva). Error bars denote SEM.

Figure VI.4

[ht!] The organization of swimming behavior. Conditioned turning probabilities reveals a tendency to repeat turning in the same direction. Blue curve shows the total turning angle of bouts that directly follow a left turn >10 degrees and red curves the corresponding data for right turns. (B) Examples of how different types of bouts are preferentially performed in certain combinations and orders. Blue crosses denoted the actual probability of observing a given bout pair sequence and black dots represent the expected probability based on the overall frequency of observation of the two bout types making up the pair (constructed from 100 shuffled swimming sequences – note the clear separation between the observed randomized data for several of the pairs; The (X-X) sub-chain (top row) is the sum of probabilities for (1-1), (2-2), (3-3), etc.). (C) Swimming bouts of a given

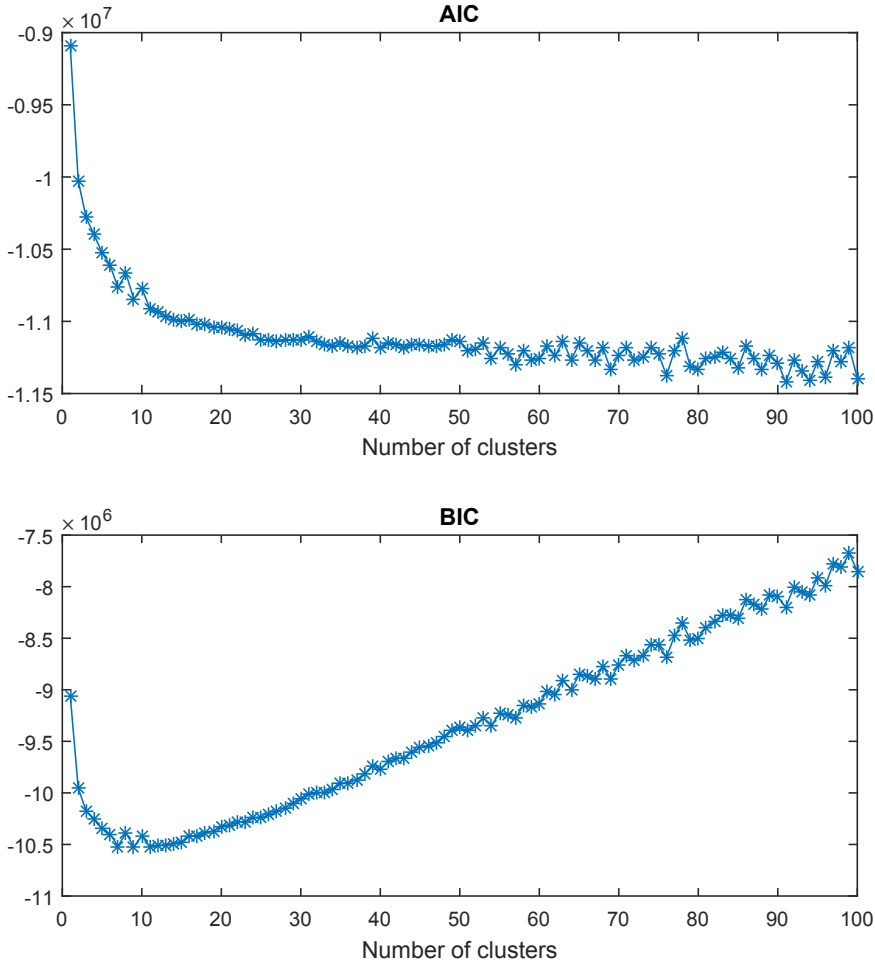
type tend to be concatenated into longer sequences of repetitive actions. The probability of observing one of the 15 types of swimming bouts is shown as a function of the number of previous observations of the same bout in a row. The theoretical probabilities of the corresponding memoryless system with the same bout class distributions as the actual data are plotted as a dashed line. Error bars represent SEM.

Figure VI.5

[ht!] Pharmacological manipulations induce specific changes in swimming behavior involving both simple kinematic features and action sequencing. (A-D) Example of pharmacological effects on a single kinematic feature: Changes in the average number of bouts displayed per second following different pharmacological manipulations throughout the recording period reveal drug and dose specific effects (shaded area denotes period of drug administration). (E-H) Example of pharmacological effects on higher order organization of behavior: Changes in the relative overrepresentation of one specific bout pair (type1-type 1, cf. Figure VI.2) following different pharmacological manipulations throughout the recording period reveal drug and dose specific effects. Note that the value presented represents the probability (in %) of observing the pair [1, 1] minus the theoretical probability based on the overall observed frequency of the constituent bout type during the same time period (thereby correcting for changes in the relative frequency of single bouts of this type). (I-K) Global changes in swimming behavior shown for each of the treatment groups for the first three dimension in a PCA sub-space constructed from the larger 435-dimensional feature space. Note the relative separation of the different drug treated groups in certain dimensions and the relative smoothness of the curves indicating gradual rather than abrupt changes in the global motor behavior of each treatment group over time. (L-O) Dose dependence of the specific behavioral changes induced by each drug. The average behavior displayed by the group receiving the highest dose minus the behavior of the control group represents the specific direction and size (=1) of the vector in feature space that represents the effect of each drug. Geometrical projection of the feature vectors obtained for the different doses/time points are plotted for each set of experiments. Note that lower doses generally show intermediate behavioral changes and that all treatment groups show a robust effect over time (x-axis represent consecutive non-overlapping sample points in the time interval from 0 to 65 min). (P-S) The relative contribution of the different kinematic features to the global

representation of the behavioral changes characterizing each of the four drugs. Deviations from zero indicate a difference in this kinematic feature between each drug-treated group and the corresponding control group (e.g. the blue dot with feature #435 marks the reduced activity shown in A-D) . Note that a substantial fraction of all the features measured show clear drug-induced changes. Whiskers denote SEM.

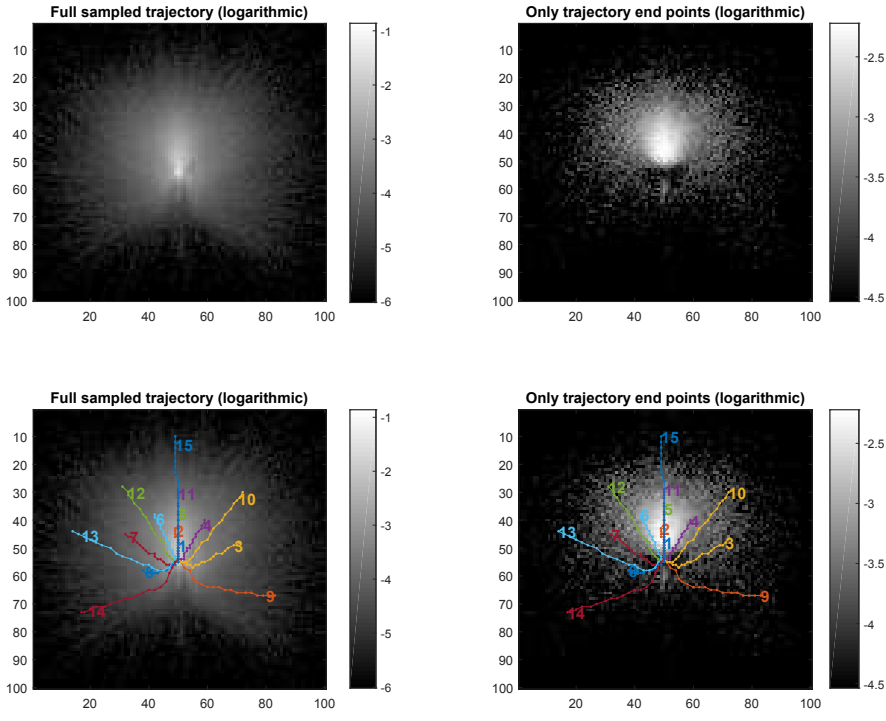
Supplementary information



Supplementary Figure VI.1

Supplementary Figure VI.1

Plots of the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) values used for deciding the number of clusters to use for the clustering

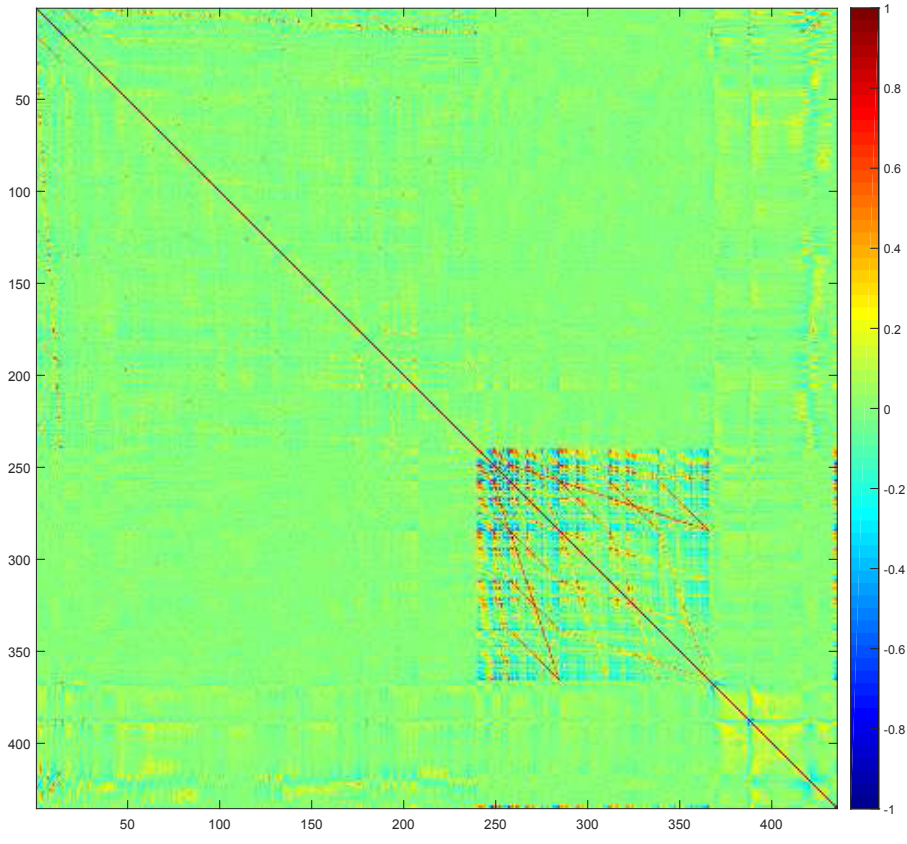


Supplementary Figure VI.2

of swim bouts.

Supplementary Figure VI.2

Logarithmic density images of the data used for defining the swim bout classes. The x and y-coordinates of all points on the subsampled trajectories of normalized swim bouts (as described in the text) are used in the left column, and only the end points are used in the right column. The top row shows only the logarithmic density images while the bottom row show the same images with the computed swim bout-classes superimposed.



Supplementary Figure VI.3

Supplementary Figure VI.3

The estimated correlation matrix of the 435-dimensional feature matrix indicates that the interdependence of the selected features is relatively limited.

Supplementary Table VI.1A-E

Quantitative comparison of goodness-of-fit to experimental data for five different models (cf. Figure VI.2 and VI.3) in terms of Negative log-likelihood, Akaike information criterion (AIC) and Bayesian information criterion (BIC).

Name	Negative log-likelihood	Number of parameters	AIC	BIC
O-U	-57857.3	2	-115710.6	-115693.7
Random walk	-24993.0	1	-49984.1	-49975.7
Poisson	-55242.4	2	-110480.8	-110463.9
Exponential	-34998.4	1	-69994.9	-69986.5
Normal	-52862.5	2	-105721.1	-105704.1

(a) Duration. Estimated O-U drive: 0.0069

Name	Negative log-likelihood	Number of parameters	AIC	BIC
O-U	42000.8	2	84005.7	84022.7
Random walk	58244.7	1	116491.5	116500.0
Poisson	48438.8	2	96881.7	96898.6
Exponential	50680.6	1	101363.3	101371.7
Normal	52505.8	2	105015.7	105032.7

(b) Distance. Estimated O-U drive: 0.0021

Name	Negative log-likelihood	Number of parameters	AIC	BIC
O-U	3552.8	2	7109.7	7125.5
Random walk	6558.0	1	13118.0	13125.9
Poisson	11229.3	2	22462.7	22478.5
Exponential	5353.1	1	10708.2	10716.1
Normal	14683.1	2	29370.3	29386.0

(c) Positive angles. Estimated O-U drive: 0.0007

Supplementary Table VI.1: Quantitative comparison of goodness-of-fit to experimental data.

Name	Negative log-likelihood	Number of parameters	AIC	BIC
O-U	4846.5	2	9697.0	9712.3
Random walk	7094.8	1	14191.6	14199.2
Poisson	10062.9	2	20129.9	20145.2
Exponential	5928.7	1	11859.4	11867.0
Normal	12507.9	2	25019.8	25035.1

(d) Negative angles. Estimated O-U drive: 0.0006

Name	Negative log-likelihood	Number of parameters	AIC	BIC
O-U	80715.9	2	161435.8	161452.7
Random walk	81342.7	1	162687.4	162695.8
Poisson	104435.2	2	208874.5	208891.4
Exponential	83921.4	1	167844.9	167853.4
Normal	117529.9	2	235063.8	235080.6

(e) Inter-bout waiting time. Estimated O-U drive: 0.0001

Supplementary Table VI.1: Quantitative comparison of goodness-of-fit to experimental data.

Supplementary Table VI.2

List of features.

Supplementary Table VI.3A-D

Cross-validation of dose response data shown in Figure VI.5M-P. Half of the data set was used to construct the <highest dose - control> vector and the remaining part of the data set was projected onto this vector.

The measures of separation used in the tables are defined as follows.

1 Ratio of highest dose projections significantly larger ($p < 0.01$) than control.

Feature name	Feature description for a given zebrafish larvae and a time interval.	Number of values
Distribution of bout classes	The fraction of swim bouts that are of class X.	1-15
Distribution of second order bout classes chains	The fraction of pairs of consecutive swim bouts where the first is of class X and the second of class Y.	16-240
Distribution of inter-bout waiting times	The fraction of waiting times that are among the k/N and (k+1)/N shortest waiting times (by the all-time waiting time distribution of the current zebrafish larvae). N is the number of bins and k goes from 0 to N-1. Here, N=9 is used.	241-249
Distribution of second, third and fourth order inter-bout waiting time chains	The fraction of (two/three/four)-pairs of consecutive waiting times where the first is of type X, the second of type Y, etc. Here the waiting times are binned by N = 3.	250-366
Distribution of swim bout durations	The fraction of swim bouts with durations between 0.05k seconds and 0.05(k+1) seconds, where k goes from 0 to 19. Longer swim bouts are placed in bin number 21.	367-387
Distribution of swim bout distances	The fraction of swim bouts with distances between 0.1k mm and 0.1 (k+1) mm, where k goes from 0 to 19. Longer swim bouts are placed in bin number 21.	388-408

Supplementary Table VI.2: List of features

2 Ratio of highest dose projections significantly larger ($p < 0.01$) than the lowest dose (excluding control).

3 Ratio of highest dose projections significantly larger ($p < 0.01$) than the middle dose.

Distribution of swim bout cumulative turning	The fraction of swim bouts with cumulative turning between 15k degrees and 15(k+1) degrees, where k goes from -12 to 11. Swim bouts with cumulative turning less than -180 degrees are placed in bin number 1 and swim bouts with cumulative turning greater than 180 degrees are placed in bin number 26.	409-434
Number of bouts per second	The number of bouts per second.	435

Supplementary Table VI.2: List of features

4 Ratio of highest dose projections significantly larger ($p < 0.01$) than all other doses (including control) simultaneously.

	AMP 10 µM	Apo 0.5 µM	APO 50µM	MK 20 µM
1	100.0%	100.0%	83.3%	100.0%
2	100.0%	100.0%	16.7%	100.0%
3	100.0%	83.3%	33.3%	16.7%
4	100.0%	83.3%	16.7%	16.7%

(a) Dose response vectors defined on subset 1 and evaluated on subset 1

	AMP 10 µM	Apo 0.5 µM	APO 50µM	MK 20 µM
1	100.0%	100.0%	33.3%	100.0%
2	100.0%	50.0%	0.0%	100.0%
3	100.0%	16.7%	0.0%	33.3%
4	100.0%	16.7%	0.0%	33.3%

(b) Dose response vectors defined on subset 1 and evaluated on subset 2

	AMP 10 µM	Apo 0.5 µM	APO 50µM	MK 20 µM
1	100.0%	100.0%	16.7%	100.0%
2	100.0%	100.0%	33.3%	66.7%
3	100.0%	0.0%	0.0%	16.7%
4	100.0%	0.0%	0.0%	16.7%

(c) Dose response vectors defined on subset 2 and evaluated on subset 1

	AMP 10 µM	Apo 0.5 µM	APO 50µM	MK 20 µM
1	100.0%	100.0%	50.0%	100.0%
2	100.0%	100.0%	66.7%	100.0%
3	100.0%	83.3%	16.7%	50.0%
4	100.0%	83.3%	0.0%	50.0%

(d) Dose response vectors defined on subset 2 and evaluated on subset 2

Supplementary Table VI.3: Cross-validation of dose response data

References

[1] J Wayne Aldridge, Kent C Berridge, Mark Herman, and Lee Zimmer. "Neuronal coding of serial order: syntax of grooming in the neostriatum". In: *Psychological Science* 4.6 (1993), pp. 391–395.

[2] Jordan M Bailey, Anthony N Oliveri, and Edward D Levin. "Pharmacological analyses of learning and memory in zebrafish (Danio rerio)". In: *Pharmacology Biochemistry and Behavior* 139 (2015), pp. 103–111.

[3] TP Barros, WK Alderton, HM Reynolds, AG Roach, and S Berghmans. "Zebrafish: an emerging technology for in vivo pharmacological assessment to identify potential safety liabilities in early drug discovery". In: *British journal of pharmacology* 154.7 (2008), pp. 1400–1413.

[4] S Boyce, NMJ Rupniak, MJ Steventon, G Cook, and SD Iversen. "Psychomotor activity and cognitive disruption attributable to NMDA, but not sigma, interactions in primates". In: *Behavioural brain research* 42.2 (1991), pp. 115–121.

[5] Giancarlo Bruni, Andrew J Rennekamp, Andrea Velenich, Matthew McCarroll, Leo Gendelev, Ethan Fertsch, Jack Taylor, Parth Lakhani, Dennis Lensen, Tama Evron, et al. "Zebrafish behavioral profiling identifies multitarget antipsychotic-like compounds". In: *Nature chemical biology* (2016).

[6] Harold A Burgess and Michael Granato. "Modulation of locomotor activity in larval zebrafish during light adaptation". In: *Journal of Experimental Biology* 210.14 (2007), pp. 2526–2539.

[7] Vincent Castagné, Toni Wolinsky, Leann Quinn, and David Virley. "Differential behavioral profiling of stimulant substances in the rat using the LABORAS™ system". In: *Pharmacology Biochemistry and Behavior* 101.4 (2012), pp. 553–563.

[8] Ruth M Colwill and Robbert Creton. "Locomotor behaviors in zebrafish (Danio rerio) larvae". In: *Behavioural processes* 86.2 (2011), pp. 222–229.

[9] Xiaojuan Dan, Bradley R King, Julien Doyon, and Piu Chan. "Motor sequence learning and consolidation in unilateral de novo patients with Parkinson's disease". In: *PloS one* 10.7 (2015), e0134291.

[10] Timothy W Dunn, Christoph Gebhardt, Eva A Naumann, Clemens Riegler, Misha B Ahrens, Florian Engert, and Filippo Del Bene. "Neural circuits underlying visually evoked escapes in larval zebrafish". In: *Neuron* 89.3 (2016), pp. 613–628.

[11] Fredrik Ek, Marcus Malo, Madelene Åberg Andersson, Christoffer Wedding, Joel Kronborg, Peder Svensson, Susanna Waters, Per Petersson, and Roger Olsson. "Behavioral Analysis of Dopaminergic Activation in Zebrafish and Rats Reveals Similar Phenotypes". In: *ACS chemical neuroscience* 7.5 (2016), pp. 633–646.

[12] Jesper Ericsson, Gilad Silberberg, Brita Robertson, Martin A Wikström, and Sten Grillner. "Striatal cellular properties conserved from lampreys to mammals". In: *The Journal of physiology* 589.12 (2011), pp. 2979–2992.

[13] Jesper Ericsson, Marcus Stephenson-Jones, Juan Pérez-Fernández, Brita Robertson, Gilad Silberberg, and Sten Grillner. "Dopamine differentially modulates the excitability of striatal neurons of the direct and indirect pathways in lamprey". In: *The Journal of Neuroscience* 33.18 (2013), pp. 8045–8054.

[14] Anders Fredriksson and Trevor Archer. "Hyperactivity following postnatal NMDA antagonist treatment: reversal by D-amphetamine". In: *Neurotoxicity research* 5.7 (2003), pp. 549–564.

[15] Arild O Gautestad and Atle Mysterud. "The Lévy flight foraging hypothesis: forgetting about memory may lead to false verification of Brownian motion". In: *Movement ecology* 1.1 (2013), p. 1.

[16] George L Gerstein and Benoit Mandelbrot. "Random walk models for the spike activity of a single neuron". In: *Biophysical journal* 4.1 Pt 1 (1964), p. 41.

[17] Sten Grillner, Jeanette Hellgren, Ariane Menard, Kazuya Saitoh, and Martin A Wikström. "Mechanisms for selection of basic motor programs–roles for the striatum and pallidum". In: *Trends in neurosciences* 28.7 (2005), pp. 364–370.

[18] Sten Grillner, Brita Robertson, and Marcus Stephenson-Jones. "The evolutionary origin of the vertebrate basal ganglia and its role in action selection". In: *The Journal of physiology* 591.22 (2013), pp. 5425–5431.

[19] Anderson Manoel Herculano and Caio Maximino. “Serotonergic modulation of zebrafish behavior: towards a paradox”. In: *Progress in Neuro-Psychopharmacology and Biological Psychiatry* 55 (2014), pp. 50–66.

[20] TD Irons, PE Kelly, DL Hunter, RC Macphail, and S Padilla. “Acute administration of dopaminergic drugs has differential effects on locomotion in larval zebrafish”. In: *Pharmacology Biochemistry and Behavior* 103.4 (2013), pp. 792–813.

[21] Allan V Kalueff, Michael Gebhardt, Adam Michael Stewart, Jonathan M Cachat, Mallorie Brimmer, Jonathan S Chawla, Cassandra Craddock, Evan J Kyzar, Andrew Roth, Samuel Landsman, et al. “Towards a comprehensive catalog of zebrafish behavior 1.0 and beyond”. In: *Zebrafish* 10.1 (2013), pp. 70–86.

[22] Renate Kopp, Juliette Legler, and Jessica Legradi. “Alterations in locomotor activity of feeding zebrafish larvae as a consequence of exposure to different environmental factors”. In: *Environmental Science and Pollution Research* (2016), pp. 1–9.

[23] Alexandros Kyriakatos, Riyadh Mahmood, Jessica Ausborn, Christian P Porres, Ansgar Büschges, and Abdeljabbar El Manira. “Initiation of locomotion in adult zebrafish”. In: *The Journal of Neuroscience* 31.23 (2011), pp. 8422–8431.

[24] Chuan-Ching Lai, Li-Jen Lee, and Hsiang-Shu Yin. “Behavioral and Neurochemical Changes Induced by Repetitive Combined Treatments of Ketamine and Amphetamine in Mice”. In: *Neurochemical research* 39.11 (2014), pp. 2180–2188.

[25] Xiuyun Liu, Ning Guo, Jia Lin, Yinglan Zhang, Xiao Qian Chen, Sheng Li, Lin He, and Qiang Li. “Strain-dependent differential behavioral responses of zebrafish larvae to acute MK-801 treatment”. In: *Pharmacology Biochemistry and Behavior* 127 (2014), pp. 82–89.

[26] Corrine K Lutz. “Stereotypic Behavior in Nonhuman Primates as a Model for the Human Condition”. In: *ILAR Journal* 55.2 (2014), pp. 284–296.

[27] Robert K McNamara, Aaron Logue, Kevin Stanford, Ming Xu, Jianhua Zhang, and Neil M Richtand. “Dose–response analysis of locomotor activity and stereotypy in dopamine D3 receptor mutant mice following acute amphetamine”. In: *Synapse* 60.5 (2006), pp. 399–405.

[28] Fabiano Peres Menezes, Luiza Wilges Kist, Maurício Reis Bogo, Carla Denise Bonan, and Rosane Souza Da Silva. “Evaluation of age-dependent response to NMDA receptor antagonism in zebrafish”. In: *Zebrafish* 12.2 (2015), pp. 137–143.

[29] Jonathan W Mink. “The basal ganglia: focused selection and inhibition of competing motor programs”. In: *Progress in neurobiology* 50.4 (1996), pp. 381–425.

[30] Olivier Mirat, Jenna R Sternberg, Kristen E Severi, and Claire Wyart. “ZebraZoom: an automated program for high-throughput behavioral analysis and categorization.” In: *Frontiers in Neural Circuits* 7 (2013).

[31] Michael F Presti, Barry C Gibney, and Mark H Lewis. “Effects of intrastriatal administration of selective dopaminergic ligands on spontaneous stereotypy in mice”. In: *Physiology & behavior* 80.4 (2004), pp. 433–439.

[32] Peter Redgrave, Tony J Prescott, and Kevin Gurney. “The basal ganglia: a vertebrate solution to the selection problem?” In: *Neuroscience* 89.4 (1999), pp. 1009–1023.

[33] Ueli Schibler. “Circadian time keeping: the daily ups and downs of genes, cells, and organisms”. In: *Progress in brain research* 153 (2006), pp. 271–282.

[34] David W Sims, Emily J Southall, Nicolas E Humphries, Graeme C Hays, Corey JA Bradshaw, Jonathan W Pitchford, Alex James, Mohammed Z Ahmed, Andrew S Brierley, Mark A Hindell, et al. “Scaling laws of marine predator search behaviour”. In: *Nature* 451.7182 (2008), pp. 1098–1102.

[35] Joel Sjöbom, Martin Tamtè, Pär Halje, and Petersson Per. “The role of corticostriatal circuits in transition of behavior”. In: *Society for Neuroscience* Program No. 720.10 (2016).

[36] Richard Stephenson, Joonbum Lim, Svetlana Famina, Aimee M Caron, and Harold B Dowse. “Sleep-Wake Behavior in the Rat Ultradian Rhythms in a Light-Dark Cycle and Continuous Bright Light”. In: *Journal of biological rhythms* 27.6 (2012), pp. 490–501.

[37] Adam Michael Stewart, Michael Nguyen, Keith Wong, Manoj K Poudel, and Allan V Kalueff. “Developing zebrafish models of autism spectrum disorder (ASD)”. In: *Progress in Neuro-Psychopharmacology and Biological Psychiatry* 50 (2014), pp. 27–36.

[38] David C Swinney and Jason Anthony. “How were new medicines discovered?” In: *Nature reviews Drug discovery* 10.7 (2011), pp. 507–519.

[39] Jessica A Tiedeken and John S Ramsdell. “Embryonic exposure to domoic acid increases the susceptibility of zebrafish larvae to the chemical convulsant pentylenetetrazole”. In: *Environmental health perspectives* (2007), pp. 1547–1552.

[40] George E Uhlenbeck and Leonard S Ornstein. “On the theory of the Brownian motion”. In: *Physical review* 36.5 (1930), p. 823.

[41] Jeffery R Wickens, Jon C Horvitz, Rui M Costa, and Simon Killcross. “Dopaminergic mechanisms in actions and habits”. In: *The Journal of neuroscience* 27.31 (2007), pp. 8181–8183.

[42] Yangzhong Zhou, Richard T Cattley, Clinton L Cario, Qing Bai, and Edward A Burton. “Quantification of larval zebrafish motor function in multi-well plates using open-source MATLAB® applications”. In: *Nature protocols* 9.7 (2014), p. 1533.

Paper VII

Tracking the Motion of Box Jellyfish

TOBIAS KJELLBERG, MAGNUS OSKARSSON, TOBIAS PALMÉR AND KALLE ÅSTRÖM

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract: In this paper we investigate a system for tracking the motion of box jellyfish *tripedalia cystophora* in a special test setup. The goal is to measure the motor response of the animal given certain visual stimuli. The approach is based on tracking the special sensory structures – the rhopalia – of the box jellyfish from high-speed video sequences. We have focused on a real-time system with simple building blocks in our system. However, using a combination of simple intensity based detection and model based tracking we achieve promising tracking results with up to 95% accuracy.

1 Introduction

Box jellyfish, or cubomedusae, have very special visual systems, [1, 3]. The visual system is based on four identical sensory structures, which are called rhopalia. Each rhopalia consists of six different eyes: the upper and lower lens eyes, the pit eyes, and the slit eyes [5, 7, 11, 12, 15, 18]. The lens eyes have image-forming optics and resemble vertebrate and cephalopod eyes [10, 13, 14].

The role of vision in box jellyfish is known to involve phototaxis, obstacle avoidance, control of swim-pulse rate and advanced visually guided behaviour [2, 6, 8]. Most known species of box jellyfish are found in shallow water habitats where obstacles are abundant [4]. Medusae of the studied species, *T. cystophora*, live between the prop roots in Caribbean mangrove swamps [19, 21]. They stay close to the surface [19] where they prey on a phototactic copepod that gathers in high densities in the light shafts formed by openings in the mangrove canopy. The medusae are not found in the open lagoons, where they risk starvation [2]. The overall goal of this project is to learn more about the visual system and the neural processes involved. One interesting problem is the study of the connection between visual stimuli presented to the animals and the motor response of the animal. However, tracking free-swimming jellyfish poses a very demanding tracking problem. Instead we look at a special experimental setup where the animals are tethered whilst they are submitted to different light stimuli. The goal is then to track how the animals direct themselves, i.e. how they would like to move. In

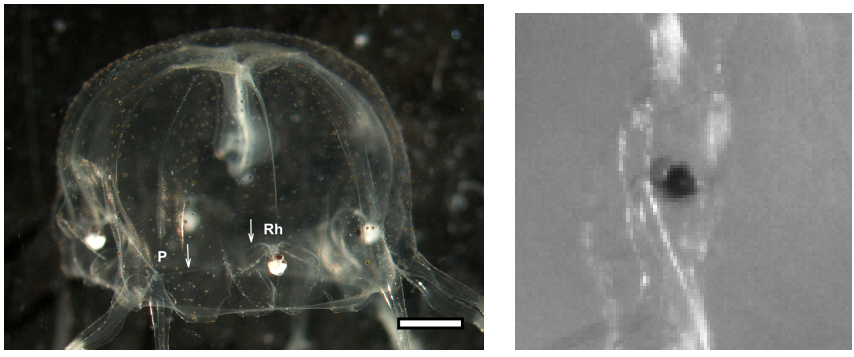


Figure VII.1: Left: The box jellyfish *tripedalia cystophora* is only a couple of mm large and almost completely transparent. Right: A close-up of the rhopalia from one frame recorded in the experimental setup.

order to do this, an initial goal which we discuss in this paper is how to track the four rhopalia. These appear as four dark discs, situated on the perimeter of the bell, see figure VII.2. There have been a number of previous studies on motor response from controlled visual stimuli, see e.g. [16, 17].

2 Experimental setup

In this study we used in total 33 animals, with sizes ranging from 0.43 to 0.89 cm. The animals were tethered by the top of the bell during the experiments, using a glass pipette with gentle suction, and placed in a Plexiglas tank with inside dimensions of $5 \times 5 \times 5$ cm. The vertical walls of the tank were covered with diffusing paper and a neutral density filter. Each vertical wall was illuminated from the outside by four blue-green LEDs. The diffuser was used to make a plane light source, while the neutral density filter was used to increase the contrast between lit and dark panels and switching one or more panels off was used as the behavioural trigger. The colour of the LEDs matched the maximum spectral sensitivity of the animals and had a peak emission at 500 nm. During the experiments a box was placed over the set-up in order to eliminate visual cues coming from outside. Image sequences were recorded with a high-speed camera operated at 150 frames per second. The dataset consists of 15 video sequences, each with around 100 greyscale frames. Each greyscale frame has a resolution of 800×864 pixels.

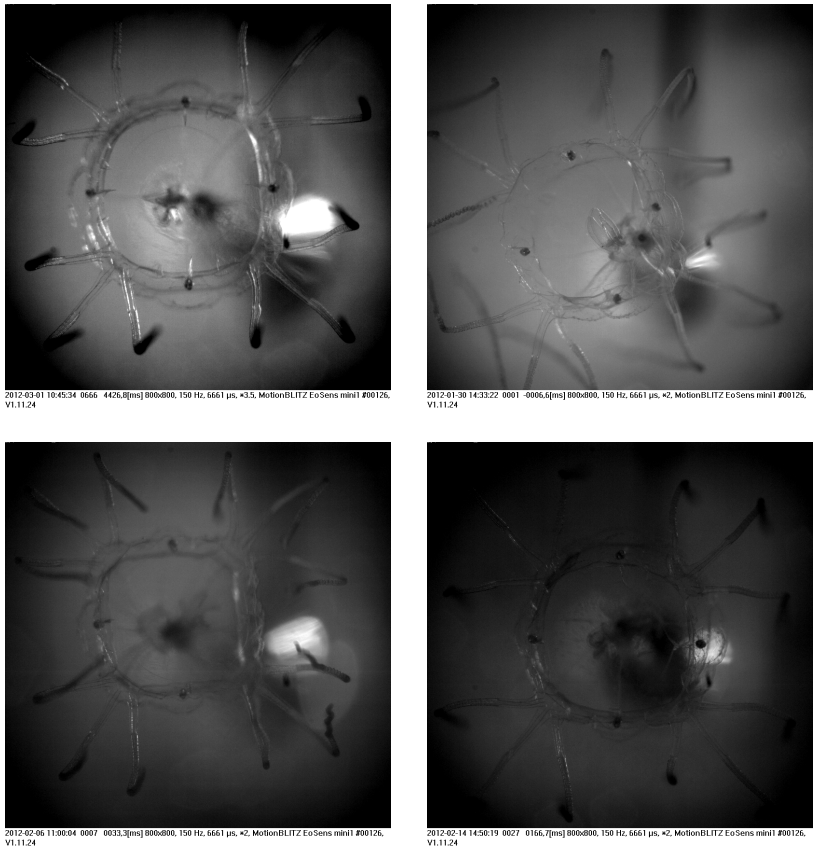


Figure VII.2: Example input frames from a number of different sequences. Notice the high variance in lighting conditions. In some frames the rhopalia are barely discernible and in many frames there are structures that have an appearance very similar to the rhopalia.

Depending on how the jellyfish is set up, the light and shadows form differently which will make the video sequences different from each other, see figure VII.2. Even though great measure has been done in order to minimize artefacts in the film sequences the difference between them can be quite large. Some of the film sequences are brighter, making it easier to find the rhopalia while some are darker and thus making it hard to distinguish the rhopalia from the background. The tethering of the animal is also visible in all sequences. This tether-shadow causes problem when the jellyfish is contracting and the rhopalia is moving over the tether-shadow. Since the box jellyfish is moving in every video sequence some parts of the jellyfish are moving in and out of focus. The physical nature of the jellyfish, i.e. being transparent, also affects the appearance and causes refraction on the surroundings.

The rhopalia in each frame in each video sequence has been manually annotated in order to perform evaluation and testing of algorithms.

3 System overview

In this section we will describe our system. Since the focus is on real-time, the chosen building blocks are quite simple in nature, especially the first steps. In figure VII.3 an overview of the system is shown. We have divided it into three parts. For every frame i, a first detection step – which only uses the local intensity distribution around a point – produces a large number of detections, Xd_i. These points are then clustered into a set of points Xc_i, in order to remove multiple detections, as well as for improved positional accuracy in the detected points. Finally these clustered positions are sent to the tracking step, which also gets as input the previous frame's four detected points, Xt_{i-1}. The output is the four detected points Xt_i. In the following subsections we will describe the different steps in more detail.

3.1 Detection

The rhopalia appear as dark discs on a brighter background in the images that are quite consistent in size and appearance in the images. See figure VII.4 for a close-up view. For this reason we have tested a number of template based approaches for detection. The template is based on the assumption that we have a number of pixels near the rhopalia that are dark. Further outside the rhopalia we should have pixels that are brighter. Figure VII.4 shows some example templates that

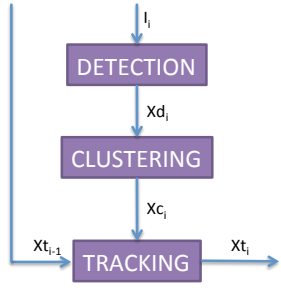


Figure VII.3: An overview of the system. For each input frame I_i we run the detection algorithm. This produces a number of tentative points Xd_i. This set of points is then sent to the clustering algorithm, which then outputs a smaller number of refined positions Xc_i. These points are fed into the tracking algorithm, alongside the four point positions from the previous frame, Xt_{i-1}. The final output is then the four detected points Xt_i.

we have tried. For speed we have adopted quite sparse templates. Top row shows pixels that should be inside the rhopalia, and hence should be darker. Bottom row shows the pixels that are assumed to be outside the rhopalia. For each point $X_i(j)$ in the input image I_i, we can then define a number of inside and outside points, $\Omega_{in}(j)$ and $\Omega_{out}(j)$. Examples of these point sets can be seen in figure VII.4. We have looked at two types of measures, one absolute and one relative. For the absolute measure we define a threshold for the inner pixels, t_{in} and one for the outer pixels, t_{out}. We then count the number of inside and outside pixels that fulfil the constraints, i.e.

$$N_{abs}(j) = \sum \Gamma(\Omega_{in}(j) \leq t_{in}) + \sum \Gamma(\Omega_{out}(j) \geq t_{out}), \tag{VII.1}$$

where $\Gamma(x) = 1$ if x is true and zero otherwise, and

$$Xd_i = \{X_i(j) | N_{abs}(j) > N_{det}\}, \tag{VII.2}$$

where N_{det} is some bound.

For the relative measure we randomly compare n inside and outside pixels, and count how many of the inside pixels are darker than the outside pixels. So if

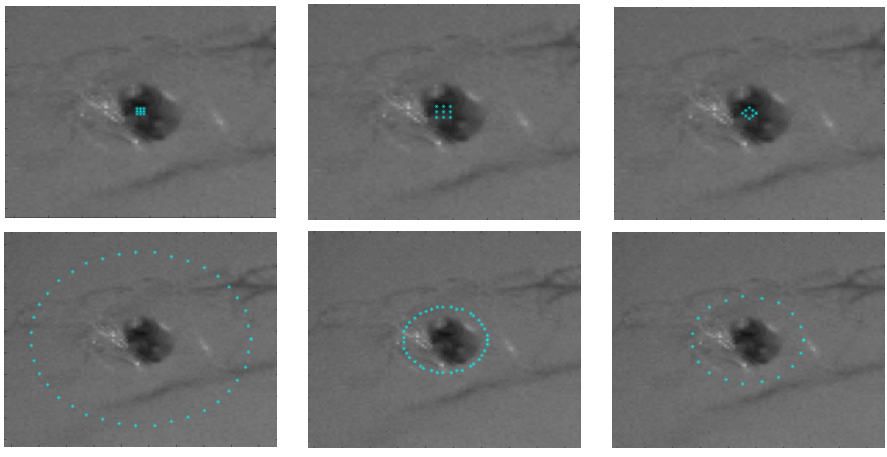


Figure VII.4: A number of templates for the detection step. Top row shows pixels that should be inside the rhopalia, and hence should be darker. Bottom row shows the pixels that are assumed to be outside the rhopalia.

we let $R(\Omega)$ denote a function that randomly chooses a point from the set Ω we have,

$$N_{rel}(j) = \sum_{k=1}^{n} \Gamma(R(\Omega_{in}(j)) < R(\Omega_{out}(j)), \tag{VII.3}$$

and

$$Xd_i = \{X_i(j) | N_{rel}(j) > N_{det}\}. \tag{VII.4}$$

We have evaluated our whole system in order to find templates that generate enough possible detections, i.e. that in most cases at least generates the four correct points, but that do not generate excessive amounts of false positives. In figure VII.5 a typical detection result is shown. Since there are many rhopalia-like structures in the images we get a large number of false positives, but these will be eliminated in the following clustering and tracking steps.

3.2 Clustering

We take a scale space approach for clustering the detections; by smoothing the input image I_i isotropically with a Gaussian kernel we get a low scale version

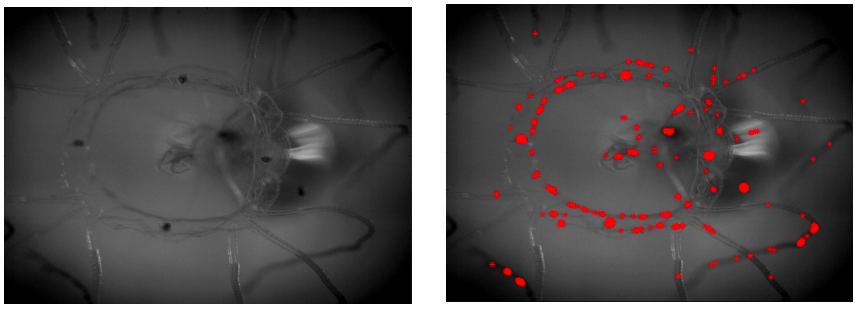


Figure VII.5: A typical detection result is shown. Since there are many rhopalia-like structures in the images we get a large number of false positives, but these will be eliminated in the following clustering and tracking steps.

I_{sm}. We then find all local minima X_{loc} of I_{sm}. The reason for this is the appearance of the rhopalia as dark spots in the images. We then calculate how many detections we have within a vicinity of each X_{loc},

$$N_{loc}(j) = \sum_{k=1}^{N_d} \Gamma(||X_{loc}(j) - Xd(k)||_2 < \epsilon_{cluster}), \tag{VII.5}$$

and if there are a minimum number N_{min} of detections, then we add this local minimum to our clustered points Xc_i,

$$Xc_i = \{X_{loc}(j) | N_{loc}(j) \geq N_{min}\}. \tag{VII.6}$$

This gives a fast, accurate and quite robust way of clustering the detections. See figure VII.6b for an example result of the clustering.

3.3 Tracking

For the tracking step we have looked at a number of simple algorithms. The input consists of the four points from the previous frame, Xt_{i-1} and a number of possible candidate points Xc_i. In our model we could also look at the scenario where the general statistical distribution of the four points is given in some form. Arguably the most simple tracking is just choosing as new points, the four closest points from the candidate points to the points from the previous frame, i.e.

$$Xt_i(j) = \underset{Xc_i(k)}{\operatorname{argmin}} ||Xt_{i-1}(j) - Xc_i(k)||_2, \quad j = 1, \ldots, 4. \tag{VII.7}$$

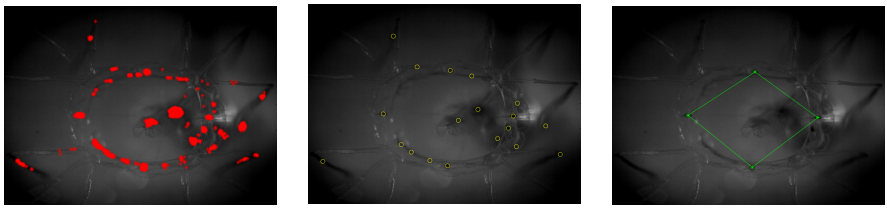


Figure VII.6: The figure shows the output of the three steps of our system for an example input image. Left all the detected points are shown in red. These are fed into the clustering which outputs the resulting yellow clustered points in the middle. Finally the tracking outputs the four points to the right, depicted in green.

We have also looked at learning an affine shape model of the four points. In this case we can fix a canonical coordinate system by placing the first three points at $(0,0)$, $(0,1)$ and $(1,0)$. The final points will then be placed at some point X_a. Using a large number of ground truth positions we can estimate the mean X_m and covariance matrix Σ for X_a. This gives us a way of finding the four points that statistically most resemble a four-point configuration, given affine deformations. For all possible subsets of four points Y_i of Xd_i we change coordinate system so that the first three points are at $(0,0)$, $(0,1)$ and $(1,0)$ and the fourth point at Y_a. If there are n such subsets we find the best subset by

$$k_{opt} = \underset{k}{\operatorname{argmin}} \ (Y_a(k) - X_m)^T \Sigma^{-1} (Y_a(k) - X_m), \tag{VII.8}$$

and

$$Xt_i = Y_i(k_{opt}). \tag{VII.9}$$

For both equation (VII.7) and (VII.8) we can choose $Xt_i = Xt_{i-1}$ if the optimal value is to large, i.e. if the best clustered points are too far away from the previous frame's points we choose the previous frame's point positions for the new frame. See figure VII.6c for an example result of the tracking.

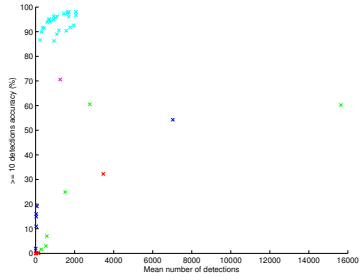
We have not yet focused on the tracking, and more complex motion models are of course possible and quite easy to implement into the framework, such as e.g. Kalman filters [20] or Particle filters [9].

4 Experimental results

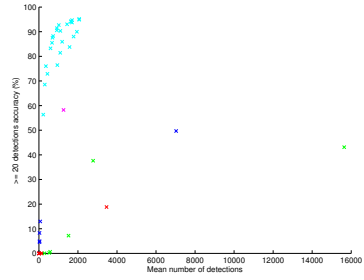
In total we have in our test setup 1469 frames, and we have manually marked the true coordinates of the rhopalia in each frame. We have tested our system on the video sequences and compared it to ground truth in the following way. For the detection accuracy we count for each of the four rhopalia in the images if there are N or more detections within a circle with radius of 10 pixels around the true coordinate. We have used $N = 10$ and $N = 20$. For the cluster accuracy we have counted the percentage of rhopalia that have a cluster within a circle of 10 pixels around the correct coordinate, and likewise for the tracking accuracy we count the percentage of rhopalia with a tracked point within 10 pixels. We do this for all the 1469 frames, with four rhopalia in each frame. We have mainly tested different detection parameters and settings. In figure VII.7 the resulting accuracy percentages can be seen. We see that the best performing systems have a tracking accuracy of 95%.

5 Conclusions

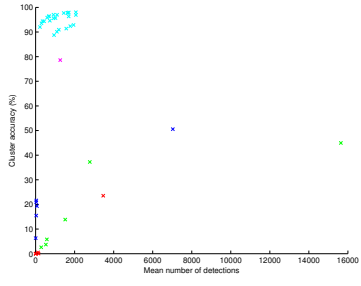
We have in this paper investigated how a system for detection of the special eyes – the rhopalia – of box jellyfish can be constructed. We have shown that using a low-level detection method in combination with clustering and tracking we can get very good performance on a varying dataset. The system can be run in real-time. The basic idea is that the system should be used in order to learn more about the visual system and neural processes of box jellyfish. The next step would be to, from the tracking of the rhopalia, measure the motion of the whole bell of the jellyfish in order to measure the motor response of the animal given certain visual stimuli.



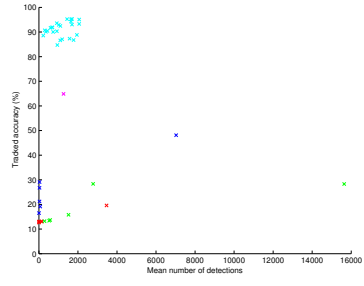
(a) ≥ 10 detections accuracy.



(b) ≥ 20 detections accuracy.



(c) Cluster accuracy.



(d) Tracked accuracy.

Figure VII.7: Evaluation of a number of different detection parameters and settings. The accuracy for the different steps in the system is shown. The best performing systems have a tracking accuracy of 95%.

References

[1] Edward W Berger. “The histological structure of the eyes of cubomedusae”. In: *Journal of Comparative Neurology* 8.3 (1898), pp. 223–230.

[2] E Buskey. “Behavioral adaptations of the cubozoan medusa Tripedalia cystophora for feeding on copepod (Dioithona oculata) swarms”. In: *Marine Biology* 142.2 (2003), pp. 225–232.

[3] C Claus. “Untersuchungen uber Charybdea marsupialis”. In: *Arb. Zool. Inst. Wien, Hft* 2 (1878), pp. 221–276.

[4] Melissa M Coates. “Visual ecology and functional morphology of Cubozoa (Cnidaria)”. In: *Integrative and comparative biology* 43.4 (2003), pp. 542–548.

[5] Franklin Story Conant. *The Cubomedusæ: a memorial volume*. Vol. 4. 1. The Johns Hopkins press, 1898.

[6] A Garm and S Mori. “Multiple photoreceptor systems control the swim pacemaker activity in box jellyfish”. In: *The Journal of experimental biology* 212.24 (2009), pp. 3951–3960.

[7] Anders Garm, Fredrik Andersson, and Dan-E Nilsson. “Unique structure and optics of the lesser eyes of the box jellyfish Tripedalia cystophora”. In: *Vision research* 48.8 (2008), pp. 1061–1073.

[8] Anders Garm, Magnus Oskarsson, and Dan-Eric Nilsson. “Box jellyfish use terrestrial visual cues for navigation”. In: *Current Biology* 21.9 (2011), pp. 798–803.

[9] Neil J Gordon, David J Salmond, and Adrian FM Smith. “Novel approach to nonlinear/non-Gaussian Bayesian state estimation”. In: *IEE Proceedings F (Radar and Signal Processing)*. Vol. 140. 2. IET. 1993, pp. 107–113.

[10] Mitsumasa Koyanagi, Kosuke Takano, Hisao Tsukamoto, Kohzoh Ohtsu, Fumio Tokunaga, and Akihisa Terakita. “Jellyfish vision starts with cAMP signaling mediated by opsin-Gs cascade”. In: *Proceedings of the National Academy of Sciences* 105.40 (2008), pp. 15576–15580.

[11] Zbynek Kozmik, Shivalingappa K Swamynathan, Jana Ruzickova, Kristyna Jonasova, Vaclav Paces, Cestmir Vlcek, and Joram Piatigorsky. "Cubozoan crystallins: evidence for convergent evolution of pax regulatory sequences". In: *Evolution & development* 10.1 (2008), pp. 52–61.

[12] G Laska and M Hündgen. "Morphologie und ultrastruktur der lichtsinnesorgane von Tripedalia cystophora Conant (Cnidaria, Cubozoa)". In: *Zool Jb Anat* 108 (1982), pp. 107–123.

[13] Vicki J Martin. "Photoreceptors of cubozoan jellyfish". In: *Coelenterate Biology 2003*. Springer, 2004, pp. 135–144.

[14] Dan-E Nilsson, Lars Gislén, Melissa M Coates, Charlotta Skogh, and Anders Garm. "Advanced optics in a jellyfish eye". In: *Nature* 435.7039 (2005), pp. 201–205.

[15] John S Pearse and Vicki B Pearse. "Vision of cubomedusan jellyfishes." In: *Science (New York, NY)* 199.4327 (1978), pp. 458–458.

[16] Ronald Petie, Anders Garm, and Dan-Eric Nilsson. "Contrast and rate of light intensity decrease control directional swimming in the box jellyfish Tripedalia cystophora (Cnidaria, Cubomedusae)". In: *Hydrobiologia* 703.1 (2013), pp. 69–77.

[17] Ronald Petie, Anders Garm, and Dan-Eric Nilsson. "Velarium control and visual steering in box jellyfish". In: *Journal of Comparative Physiology A* 199.4 (2013), pp. 315–324.

[18] Joram Piatigorsky and Zbynek Kozmik. "Cubozoan jellyfish: an Evo/Devo model for eyes and other sensory systems". In: *Int J Dev Biol* 48.8-9 (2004), pp. 719–729.

[19] Scott E Stewart. "Field behavior of Tripedalia cystophora (class Cubozoa)". In: *Marine & Freshwater Behaviour & Phy* 27.2-3 (1996), pp. 175–188.

[20] Greg Welch and Gary Bishop. *An introduction to the Kalman filter*. 1995.

[21] Bernhard Werner, CHARLES E CUTRESS, and JOHN P STUDEBAKER. "Life cycle of Tripedalia cystophora Conant (Cubomedusae)". In: (1971).

Paper VIII

Calibration, Positioning and Tracking in a Refractive and Reflective Scene

TOBIAS PALMÉR[1], GIUSEPPE BIANCO[2], MIKAEL T. EKVALL[3],
LARS-ANDERS HANSSON[3] AND KALLE ÅSTRÖM[1]
[1] *Centre for Mathematical Sciences, Lund University*
[2] *Centre for Animal Movement Research, Department of Biology, Lund University*
[3] *Aquatic Ecology, Department of Biology, Lund University*

Abstract: We propose a framework for calibration, positioning and tracking in a scene viewed by multiple cameras, through a flat refractive surface and one or several flat reflective walls. Refractions are explicitly modeled by Snell's law and reflections are handled using virtual points. A novel bundle adjustment framework is introduced for solving the nonlinear equations of refractions and the linear equations of reflections, which in addition enables optimization for calibration and positioning. The numerical accuracy of the solutions is investigated on synthetic data, and the influence of noise in image points for several settings of refractive and reflective planes is presented. The performance of the framework is evaluated on real data and confirms the validity of the physical model. Examples of how to use the framework to back-project image coordinates, forward-project scene points and estimate the refractive and reflective planes are presented. Lastly, an application of the system on real data from a biological experiment on small aquatic organisms is presented.

1 Introduction

There has been a lot of interest in the computer vision society on the subject of using and modeling cameras whose line of sight is somehow disturbed by refractions [1, 7, 10, 12, 17]. There are many different problems that can be posed by assuming different configurations of known relationships between refractive planes, cameras, scene points etc. In the case where the presence of refractions is due to the camera being under water, there are additional radiometric effects that alters the imagery [11, 14, 15, 16]. Modeling of refractions has also been combined with reflections. For example, the recovery of refractive and reflective objects has been studied by light-path triangulation [5, 13], assuming known scene points. Furthermore, the problem of reconstruction or pose estimation has been studied in the case of flat reflections [18] and catadioptric cameras with parabolic reflections [8, 9].

However, positioning and tracking of objects within a scene where there are both refractions and reflections present has to our knowledge not been considered in previous research. This is an important application in for example biological sciences, where it is common to study the behavior of objects (e.g. animals) inside of an aquarium while viewing the scene from the outside [2, 4, 6]. An equivalent problem is to estimate structure inside a refractive (and possibly reflective) scene. We therefore present theory and methods for the calibration of refractive and reflective scene parameters, assuming pre-calibrated cameras. We describe the subsequent use of methods for finding the positions of objects within the scene. A framework is created for this purpose and is applied for a study on the behavior of small aquatic organisms.

In Section 2, we explain how refractions and reflections are modeled, using Snell's law and virtual points. In Section 3, we show how the models are applied to create a residual vector and how to minimize it efficiently by bundle adjustment. Examples of how to use the framework to forward-project a scene point (Section 3.4), back-project an image point (Section 3.4) and estimate the refractive and reflective planes (Section 4) are presented. The numerical accuracy of the method is thoroughly tested on synthetic data in Section 5.1, and the validity of the framework on real data is tested using images of a calibration pattern submerged in an aquarium in Section 5.2. Lastly, an actual application of the system is presented in Section 6, providing a qualitative evaluation of the system on real data.

1.1 Related work

In [1], Agrawal et al. presents theory of flat refractive geometry, deriving e.g. that a pinhole camera viewing a scene through (multiple) parallel flat refractive surfaces corresponds to an axial camera. It is also shown that the case of double refractions in the case of air-glass-water, where the glass is thin, is well approximated by disregarding the glass. This result is also reported in [10].

A method for structure-from-motion in underwater settings where the camera-refractive plane pose is known is presented by Jordt-Sedlazeck and Koch in [12], deriving an error function for virtual cameras for use in bundle adjustment. Here, the standard methods for bundle adjustment in refractive structure-from-motion are deemed infeasible.

The problem of estimating the absolute pose of a calibrated camera viewing a refractive scene is solved minimally and near-minimally for a few different settings

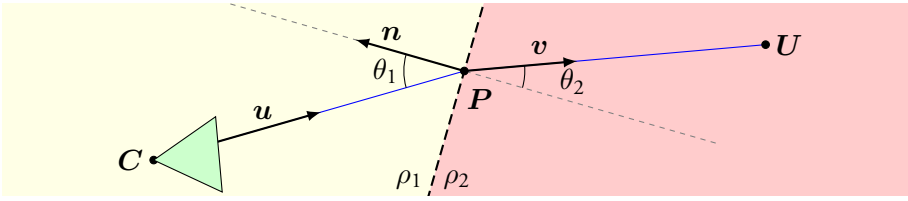


Figure VIII.1: Snell's law. A ray originating from the camera center C with direction $\boldsymbol{u}$ undergoes a change of direction according to $\rho_1 \sin\theta_1 = \rho_2 \sin\theta_2$. This causes the usually linear equations for projections, for example, to become nonlinear and much harder to solve.

of assumptions, by Haner and Åström [10].

Concerning reflections and ray-tracing, Sturm and Bonfort considered the task of computing the pose of an object without a direct view [18], viewing a calibration pattern through a flat reflective surface. In [5], Chari and Sturm use both reflections and refractions for estimating the structure of transparent refractive objects. In the case of catadioptric cameras, i.e. cameras with parabolic mirrors, Geyer and Daniilidis derive epipolar constraints and use them for structure and motion in [8, 9].

In the field of biology, Ekvall et al. studied the movement of small aquatic organisms [6] inside a rectangular aquarium using four cameras, positioned pairwise and orthogonal to two of the surfaces and approximating the cameras as orthogonal.

2 Theory of refractions and reflections

In this section, we introduce the equations for modeling refractions and reflections that are later used for defining the error functions in Section 3.2. Note that some indices of variables, later introduced in Section 3.1, are omitted here.

2.1 Refractions (Snell's law)

Snell's law describes how the angle θ_1 of an impinging ray $\boldsymbol{u}$, with respect to the refractive plane normal, is related to the angle θ_2 of the outgoing ray $\boldsymbol{v}$ by

$$\rho_1 \sin(\theta_1) = \rho_2 \sin(\theta_2), \tag{VIII.1}$$

where ρ_1 and ρ_2 are the refractive indices of the surfaces. A visualization of this is given in Fig VIII.1. This nonlinear relation is the main source of the challenges encountered in underwater structure-from-motion. Another issue is ambiguity in Eq (VIII.1) that gives rise to false solutions.

In order to express Snell's law as a function of the ray directions and plane normal, a property of the cross product is applied:

$$\boldsymbol{u} \times \boldsymbol{n} = \|\boldsymbol{u}\|\|\boldsymbol{n}\| \sin(\theta_1)\boldsymbol{w}, \tag{VIII.2}$$

where $\boldsymbol{w}$ is a unit vector orthogonal to both $\boldsymbol{u}$ and $\boldsymbol{n}$, and θ_1 is the angle between $\boldsymbol{u}$ and $\boldsymbol{n}$. Since any vector orthogonal to $\boldsymbol{u}$ and $\boldsymbol{n}$ is also orthogonal to $\boldsymbol{n}$, Eq (VIII.2) can be used to reformulate Eq (VIII.1) to

$$\rho_1 \frac{\boldsymbol{u} \times \boldsymbol{n}}{\|\boldsymbol{u}\|\|\boldsymbol{n}\|} = \rho_2 \frac{\boldsymbol{v} \times \boldsymbol{n}}{\|\boldsymbol{v}\|\|\boldsymbol{n}\|}, \tag{VIII.3}$$

where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the previously defined ray directions. Assuming that $\|\boldsymbol{n}\| = 1$ and multiplying by $\|\boldsymbol{u}\|$ and $\|\boldsymbol{v}\|$ gives the expression

$$\rho_1\|\boldsymbol{v}\|(\boldsymbol{u} \times \boldsymbol{n}) = \rho_2\|\boldsymbol{u}\|(\boldsymbol{v} \times \boldsymbol{n}). \tag{VIII.4}$$

Squaring both sides of the equation element-wise gives expressions that are polynomial in all variables, of which only two are independent (ref e.g. [10]). Solutions to this equation will later be sought for the purpose of computing forward projections.

Note that in the case where $\boldsymbol{u}$ and $\boldsymbol{n}$ are known, solving for $\boldsymbol{v}$ gives the expression

$$\boldsymbol{v} = r\boldsymbol{u} + (r\cos\theta_1 - \text{sign}(\cos\theta_1)\cos(\theta_2))\boldsymbol{n}, \tag{VIII.5}$$

where

$$\begin{aligned} \cos\theta_1 &= -\boldsymbol{n} \cdot \boldsymbol{u}, \\ \cos\theta_2 &= \sqrt{1 - r^2(1 - \cos^2\theta_1)}, \end{aligned} \tag{VIII.6}$$

and $r = \rho_1/\rho_2$. This provides a convenient method for computing the refraction of directions and will later be used for back-projections.

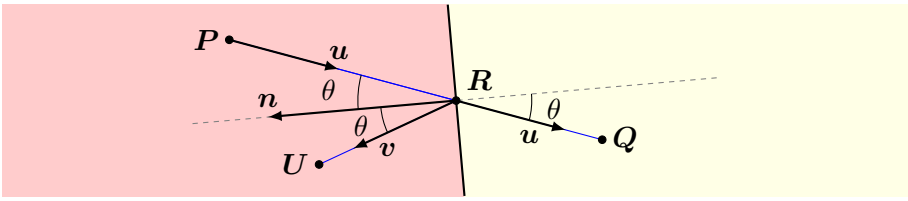


Figure VIII.2: Reflections. A ray with direction $\boldsymbol{u}$ originating from a point $\boldsymbol{P}$ intersects a reflective plane $(\boldsymbol{n}, d)$ at a point $\boldsymbol{R}$, where the ray changes direction to $\boldsymbol{v}$ and subsequently intersects a point $\boldsymbol{U}$. Note that the ray segment from $\boldsymbol{R}$ to $\boldsymbol{U}$ is the mirror image of the virtual ray segment from $\boldsymbol{R}$ to $\boldsymbol{Q}$.

2.2 Reflections

The law of reflection states that the angle θ_1 of an impinging ray is related to the angle θ_2 of the outgoing ray as

$$\theta_1 = \theta_2, \tag{VIII.7}$$

with respect to the normal of the reflective surface at the point of reflection, as depicted in Fig VIII.2. This relation can be reformulated on vector form as a linear transformation from the incoming ray direction $\boldsymbol{v}$ to the outgoing ray direction $\boldsymbol{w}$ as

$$\boldsymbol{w} = (\boldsymbol{I} - 2\boldsymbol{n}\boldsymbol{n}^\top)\boldsymbol{v}, \tag{VIII.8}$$

where $\boldsymbol{n}$ is the normal direction of the surface at the point of intersection.

An equivalent way to model reflections is to reflect points instead of directions. The observation of a point $\boldsymbol{U}$ that is reflected in a surface appears the same as the observation of a virtual point $\boldsymbol{Q}$ that is the mirror image of $\boldsymbol{U}$ and travels straight through the surface. Adapting Eq VIII.8 for points gives the following linear transformation for the reflection of a point $\boldsymbol{U}$ in $\boldsymbol{\pi}$:

$$\begin{bmatrix} \boldsymbol{Q} \\ 1 \end{bmatrix} = \begin{bmatrix} (\boldsymbol{I} - 2\boldsymbol{n}\boldsymbol{n}^\top) & -2\boldsymbol{n}d \\ \boldsymbol{0}^\top & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{U} \\ 1 \end{bmatrix}. \tag{VIII.9}$$

This property of modeling the reflection of rays provides a convenient method for computing forward projections through one or multiple reflections.

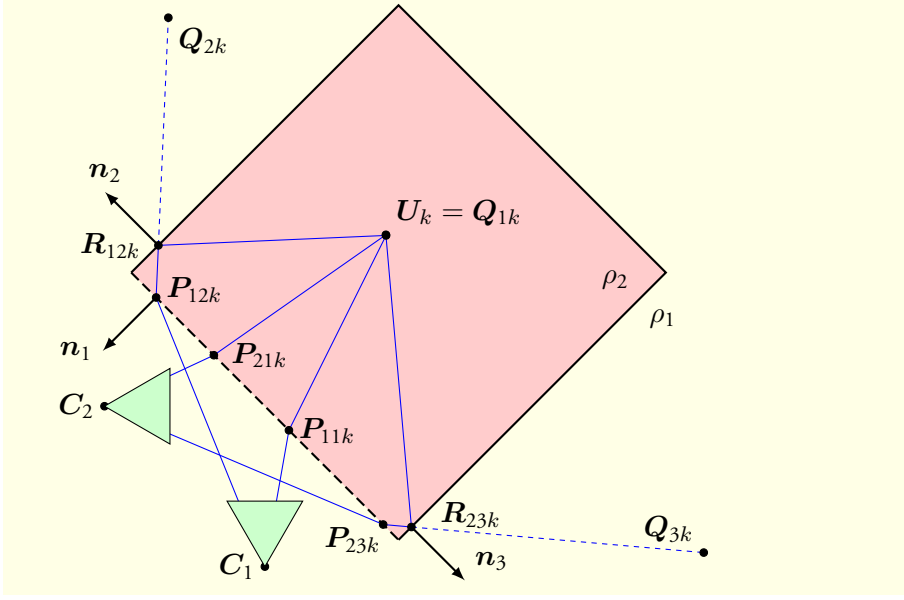


Figure VIII.3: The refraction and reflections of two pairs of rays from two camera focal points $\boldsymbol{C}_i$, all corresponding to the same scene point $\boldsymbol{U}_k$. All of the rays intersect the plane of refraction $(\boldsymbol{n}_1, d_1)$ where they change direction, resulting in second linear pieces of the ray originating at the points of refraction $\boldsymbol{P}_{ijk}$. Two of the rays also intersect a plane of reflection $(\boldsymbol{n}_j, d_j)$, $j > 1$, resulting in additional linear pieces of the ray originating at the points of reflection $\boldsymbol{R}_{ijk}$. In the case of forward projection, computing the points of reflection is conveniently avoided by instead forward projecting the virtual point $\boldsymbol{Q}_{jk}$ that is the reflection of the scene point in zero, one or several planes of reflection.

3 Method

In this section, we describe how the theory of reflections and refractions introduced in Section 2 can be applied. First the notation is introduced, then the equations of refractions and reflections are adapted for use in a residual vector, and lastly initialization and practical application is presented.

3.1 Data structure and notation

The modeling of a ray subject to refraction and reflection is based on the use of a *help point* at the refractive plane and a *virtual point* that is the reflected scene point, as shown in Fig VIII.3. Scene point k is denoted $\boldsymbol{U}_k$ and the reflection of $\boldsymbol{U}_k$ in *reflective plane configuration* $j \in \mathbb{N}$ is denoted $\boldsymbol{Q}_{jk}$. A reflective plane configuration can be for example reflection in $\boldsymbol{\pi}_2$, followed by reflection in $\boldsymbol{\pi}_4$. By defining $j = 1$ as the empty reflection, $\boldsymbol{Q}_{jk}$ denotes an *end point* that is either the scene point for $j = 1$ or a virtual point for $j > 1$. The observation of an end point by camera i corresponds to a ray in two linear pieces - from $\boldsymbol{Q}_{jk}$ to $\boldsymbol{P}_{ijk}$ and from $\boldsymbol{P}_{ijk}$ to $\boldsymbol{C}_i$, where $\boldsymbol{C}_i$ is the focal point and $\boldsymbol{P}_{ijk}$ is a *help point*. The *help point* $\boldsymbol{P}_{ijk}$ lies on the refractive plane and is defined such that the angles of the vectors $\boldsymbol{u} = \boldsymbol{P}_{ijk} - \boldsymbol{C}_i$ and $\boldsymbol{v} = \boldsymbol{Q}_{jk} - \boldsymbol{P}_{ijk}$ satisfies Eq (VIII.4).

A ray corresponding to the observation of an *end point* $\boldsymbol{Q}_{jk}$ by camera i is represented by the points it interpolates as

$$(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk}) \,, \tag{VIII.10}$$

where $\boldsymbol{P}_{ijk}$ is the *help point*. The normalized image coordinates in camera i relating to an end point $\boldsymbol{Q}_{jk}$ are denoted $\boldsymbol{u}_{ijk}$ and are given by the projection of the help point $\boldsymbol{P}_{ijk}$.

Note that in the notation used throughout the paper, it is assumed that all scene points are subject to a single refraction and that the plane of refraction separates the end points from the camera focal points.

3.2 Bundle adjustment

Equations describing refractions and reflections, as introduced in Section 2.1 and Section 2.2, together with the data structures introduced in Section 3.1 are used to create a residual vector $\boldsymbol{r}$.

The error function for the refraction of a ray $(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk})$ is adapted from Eq (VIII.4) and defined as

$$\begin{aligned} \boldsymbol{r}_{refr}(\boldsymbol{C}_i,&\boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk}, \boldsymbol{\mu}, \boldsymbol{n}_1) = \\ &\mu_1^2 \left((\boldsymbol{C}_i - \boldsymbol{P}_{ijk}) \times \boldsymbol{n}_1\right)^2 \left\|\boldsymbol{Q}_{jk} - \boldsymbol{P}_{ijk}\right\|_2^2 \\ &- \mu_2^2 \left((\boldsymbol{Q}_{jk} - \boldsymbol{P}_{ijk}) \times \boldsymbol{n}_1\right)^2 \left\|\boldsymbol{C}_i - \boldsymbol{P}_{ijk}\right\|_2^2 , \end{aligned} \tag{VIII.11}$$

where the square of the cross product is computed element-wise. The refractive error function is complemented by an error function constraining the help points to lie on the plane of refraction.

The error function for the reflection of a scene point $\boldsymbol{U}_k$ to a virtual point $\boldsymbol{Q}_{jk}$ is adapted from Eq (VIII.9) as

$$\begin{aligned}\boldsymbol{r}_{refl}(\boldsymbol{U}_k,&\boldsymbol{Q}_{jk}, \boldsymbol{\pi}_j) = \\ &\left(\boldsymbol{I} - 2\boldsymbol{n}_j\boldsymbol{n}_j^\top\right)\boldsymbol{Q}_{jk} - \boldsymbol{U}_k - 2\boldsymbol{n}_j d_j.\end{aligned} \tag{VIII.12}$$

Also, error functions for the re-projections of help points $\boldsymbol{P}_{ijk}$ to image points $\boldsymbol{u}_{ijk}$, $\boldsymbol{r}_{proj}(\boldsymbol{P}_{ijk}, \boldsymbol{u}_{ijk})$, and constraints $\|\boldsymbol{n}_j\| = 1$ for the plane normal vectors $\boldsymbol{n}_j$, $\boldsymbol{r}_n(\boldsymbol{n}_j)$, are added.

3.3 Using the Schur complement

Given an end point $\boldsymbol{Q}_{jk}$ and a camera center $\boldsymbol{C}_i$, there is always a help point $\boldsymbol{P}_{ijk}$ such that the ray $(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk})$ satisfies Snell's law. Finding the help points is computationally very cheap - each help point is subject to four constraints and only a few iterations using Gauss-Newton's method are enough to find it accurately. Furthermore, all help points are independent from other help points. The existence of $\boldsymbol{P}_{ijk}$ and the computational efficiency in estimation can be leveraged to increase effiency in the estimation of other variables as follows.

First, the residual vector and variables are partitioned as

$$\begin{aligned}\boldsymbol{r} &= \begin{bmatrix}\boldsymbol{r}_1 & \boldsymbol{r}_2\end{bmatrix}^\top, \\ \boldsymbol{x} &= \begin{bmatrix}\boldsymbol{x}_A & \boldsymbol{x}_B\end{bmatrix}^\top,\end{aligned} \tag{VIII.13}$$

where $\boldsymbol{r}_1$ are the residuals for refractions and reflections, $\boldsymbol{r}_2$ the residuals for projections and plane normalization, $\boldsymbol{x}_A$ is the vector containing help points and $\boldsymbol{x}_B$ contains some other disjoint subset of $\boldsymbol{x}$, for example the reflective plane parameters. The linearization of the problem at $\boldsymbol{x}$ is

$$\begin{cases}\delta\boldsymbol{r}_1 = \boldsymbol{J}_{1A}\delta\boldsymbol{x}_A + \boldsymbol{J}_{1B}\delta\boldsymbol{x}_B, \\ \delta\boldsymbol{r}_2 = \boldsymbol{J}_{2A}\delta\boldsymbol{x}_A + \boldsymbol{J}_{2B}\delta\boldsymbol{x}_B.\end{cases} \tag{VIII.14}$$

where $\boldsymbol{J}_{1A}$ is the matrix of partial derivatives of $\boldsymbol{r}_1$ with respect to $\boldsymbol{x}_A$ and $\boldsymbol{J}_{1B}$, $\boldsymbol{J}_{2A}$, $\boldsymbol{J}_{2B}$ are defined analogously. First solving for help points with respect to

refractions and reflections gives (approximately) $\|\boldsymbol{r}_1\| = 0$. Inserting $\boldsymbol{r}_1$ into Eq (VIII.14) and solving the first line for $\delta\boldsymbol{x}_A$ gives $\delta\boldsymbol{x}_A$ as a function of $\delta\boldsymbol{x}_B$. Inserting the solution $\delta\boldsymbol{x}_A$ into the second equation, and solving the second equation for $\delta\boldsymbol{x}_B$ results in

$$\begin{cases} \delta\boldsymbol{x}_A = -\boldsymbol{J}_{1A}^{-1}\boldsymbol{J}_{1B}\delta\boldsymbol{x}_B, \\ \delta\boldsymbol{x}_B = -(\boldsymbol{J}_{2A}\boldsymbol{J}_{1A}^{-1}\boldsymbol{J}_{1B} + \boldsymbol{J}_{2B})^{-1}\delta\boldsymbol{r}_2, \end{cases} \tag{VIII.15}$$

where $\boldsymbol{J}_{2A}\boldsymbol{J}_{1A}^{-1}\boldsymbol{J}_{1B} + \boldsymbol{J}_{2B}$ is known as the Schur-complement. This is applied in the estimation of e.g. the refractive plane, by first optimizing the help points, then optimizing in the direction $\delta\boldsymbol{x}$ obtained by solving Eq (VIII.15) for $\delta\boldsymbol{x}_A$ and $\delta\boldsymbol{x}_B$. The procedure is repeated from the newly obtained $\boldsymbol{x}$ until some stopping criteria is met.

3.4 Computing the back-projections and forward projections

The bundle adjustment methods assumes that there is an initial value for each variable, thus there is need for methods of initializing those variables. In this section, initialization for forward projections and backward projections are explained.

In the case of forward-projections, a scene point $\boldsymbol{U}_k$ is known and the set of image points $\boldsymbol{u}_{ijk}$ that corresponds to $\boldsymbol{U}_k$ are sought. This amounts to finding and then projecting the point of refraction $\boldsymbol{P}_{ijk}$ for any end point, i.e. for the scene point or any reflection $\boldsymbol{Q}_{jk}$ of it. For this purpose, the scene point is reflected in some set of plane configurations, resulting in the set of end points $\{\boldsymbol{Q}_{jk}\}$. For each such end point $\boldsymbol{Q}_{jk}$ and camera center $\boldsymbol{C}_i$, an initial value for the corresponding help point $\boldsymbol{P}_{ijk}$ is given by the intersection of the straight line from $\boldsymbol{Q}_{jk}$ to $\boldsymbol{C}_i$ and the refractive plane $\boldsymbol{\pi}_1$. For each camera i and plane configuration j, the error of refraction for the ray $\boldsymbol{r}_{refr}(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk}, \boldsymbol{\mu}, \boldsymbol{n}_1)$ is subsequently minimized with respect to the help points $\boldsymbol{P}_{ijk}$. The help points are then projected by their corresponding camera matrix to provide the sought image coordinates $\boldsymbol{u}_{ijk}$.

In the case of back-projecting an image point, assuming that the refractive and reflective planes are known, the help point and refracted ray direction can be obtained by tracing the ray from the camera focal point to the plane of refraction and computing the new direction by Eq (VIII.5). The process is analogous for the reflective planes using Eq (VIII.8). Note that the bundle adjustment framework is not used for this task.

In the case of back-projecting a set of image points $\boldsymbol{u}_{ijk}$ corresponding to the same scene point $\boldsymbol{U}_k$, assuming knowledge of the reflective plane configuration j each ray undergoes, the scene point $\boldsymbol{U}_k$ is sought. Help points $\boldsymbol{P}_{ijk}$ and refracted and reflected ray directions are computed as was presented in the case of single back-projection. This results in a set of points and directions whose intersection is used as an initial value for the scene point $\boldsymbol{U}_k$. Note that in the case where there are at least two rays unaffected by reflections, the initial value is more conveniently computed using only those rays. The virtual points are then initialized by reflecting $\boldsymbol{U}_k$ in the specified reflective plane configurations using Eq (VIII.9), Thus the ray $(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk})$ is known for each end point. The residual vector for this estimation is composed of $\boldsymbol{r}_{refl}(\boldsymbol{U}_k, \boldsymbol{Q}_{jk}, \boldsymbol{\pi}_j)$, $\boldsymbol{r}_{refr}(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk}, \boldsymbol{\mu}, \boldsymbol{n}_1)$ and $\boldsymbol{r}_{proj}(\boldsymbol{P}_{ijk}, \boldsymbol{u}_{ijk})$, and is subsequently minimized with respect to the virtual points $\boldsymbol{Q}_{jk}$ and scene points $\boldsymbol{U}_k$.

An issue in tracking applications is that the configuration of reflections that each ray undergoes on its way to the end point can not always be assumed to be known. However, the above methods can be combined to create a method that can be used to find the configurations. Given a putative pair of image point correspondences and assume that their corresponding rays do not undergo reflections before meeting their common end point. Then the end point can be estimated by back-projection as previously explained. If the estimated end point is located outside of the scene (e.g. in the case of an aquarium), the estimate is discarded. A set of reprojected points are given by reflecting the end point in several reflection configurations and forward projecting each of the resulting virtual points. Each such forward projection is feasible only if the help point is located inside of the bounded refractive plane. Lastly, the distances to detected image points are evaluated and points sufficiently close to the re-projections are assumed to correspond to the same scene point. If additional image point correspondences are found, the position of the scene point can subsequently be further optimized.

4 Calibrating the scene

The calibration process is divided in two steps - pre-calibration and refractive scene calibration. The pre-calibration is, in this case, composed of estimating the intrinsic parameters of each camera and the relative poses of the cameras, i.e. radial distortion, skew, focal length, etc. and position and orientation. This is done using the standard methods with a moving checkerboard [3, 19].

The refractive scene calibration consists of estimating the parameters of the planes of refraction and reflections in the scene, i.e. $(\pi_1, \pi_2, ...)$, and is done using a set of observed image points. It is assumed that the plane configurations that each of the corresponding rays are reflected in before they meet their scene points are known, and that there are sufficiently small errors in the initial values of the scene parameters. The method for back-projection explained in Section 3.4 is used to find a ray $(\boldsymbol{C}_i, \boldsymbol{P}_{ijk}, \boldsymbol{Q}_{jk})$ for each image point. Each ray is entered into the framework and the corresponding residual vector of refractions, reflections and projections is minimized with respect to the help points, end points and planes. The estimated planes are the results of calibration.

In the experimental evaluation (section 5.2), the refractive scene calibration was performed using a checkerboard submerged in a water tank and the checkerboard points were found using the same checkerboard-detection algorithms as for the pre-calibration. In this case, heuristics for deciding the refractive/reflective configuration that each ray undergoes could be easily decided due to the scene-geometry of this particular application. A close-enough initial calibration was attained by manually selecting the points in the corners of the tank for which the geometry was known.

5 Experiments

The performance of the system is evaluated on both real data and synthetic data. The accuracy and robustness in estimates of image points, scene points and refractive and reflective planes is analyzed in Section 5.1. Real images are analyzed in Section 5.2, where a checkerboard is submerged in an aquarium.

5.1 Synthetic experiments

The method for forward-projection is evaluated by simulating 1000 random scenes, each consisting of a refractive plane, a reflective plane and two camera poses. Scene points are generated and subsequently forward-projected to image points in the cameras, as explained in Section 3.4. The difference in θ_2 given by solving $\|\boldsymbol{n}\|\|\boldsymbol{v}\| \cos(\theta_2) = \boldsymbol{n}^\top \boldsymbol{v}$ and Eq (VIII.1) for θ_2, using the estimated θ_1, is evaluated and presented in Fig VIII.4. This shows that the forward projections of the synthetic data are accurate, thereby proving the validity of the synthetic data that the following analyses are based on.

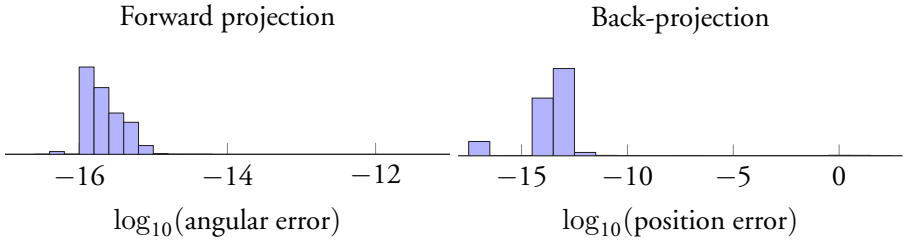


Figure VIII.4: Left: the distribution of angular errors with respect to Snell's law for forward-projection in 1000 random synthetic problem instances with no added noise. Right: The distribution of error in position, defined as the euclidean distance from an estimate to ground truth, for back-projection in 1000 random synthetic problem instances with no added noise. Points were simulated at distances approximately 200 mm from the camera focal points.

The numerical accuracy of the method for back-projection (introduced in Section 3.4) is tested by back-projecting simulated image points and estimating the 3d positions that corresponding pairs of image points are observations of. The estimated 3d points are compared to the ground truth 3d points, and the distribution of distances between the points is presented in Fig VIII.4. Sensitivity to noise was tested by adding zero mean normal distribution noise with increasing variance to the image points, resulting in the distribution of distances shown in Fig VIII.5.

The numerical accuracy of the method for estimating refractive and reflective planes, as introduced in Section 4, is tested on 1000 random simulated problem instances. Each instance consists of the observed image coordinates of 3 scene points (the minimal case) in a fixed, simulated scene with two cameras and planes similar to the real scene analyzed in Section 6. The performance of the plane estimation method is tested by perturbing the initial plane parameters slightly and using the image points to estimate the planes. Fig VIII.5 presents the results of plane estimation with respect to estimating the position of the refractive plane, where the position of the plane is defined as the point at the center of the plane, computed using the ground truth planes for all other planes. Additionally, sensitivity to noise in image coordinates is tested by adding zero-mean normal distributed noise to the image coordinates and subsequently estimating the planes. The result of estimating the refractive plane position as a function of added noise variance is presented in Fig VIII.6.

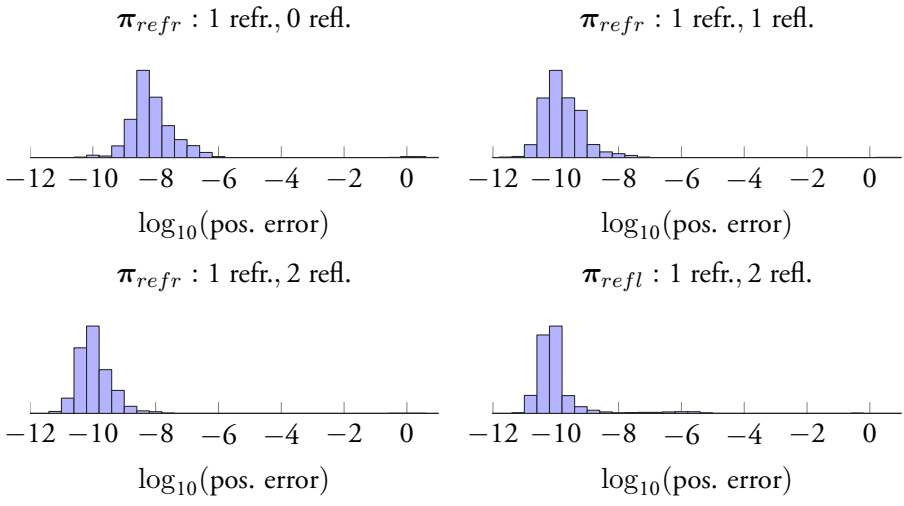


Figure VIII.5: Distribution of errors in estimated plane positions compared to ground truth for three scene settings, given slightly perturbed initial plane parameters. Note that the distributions in the top row and the left column shows the errors in position of the refractive plane, and the bottom right shows the distribution of errors in position of both reflective planes.

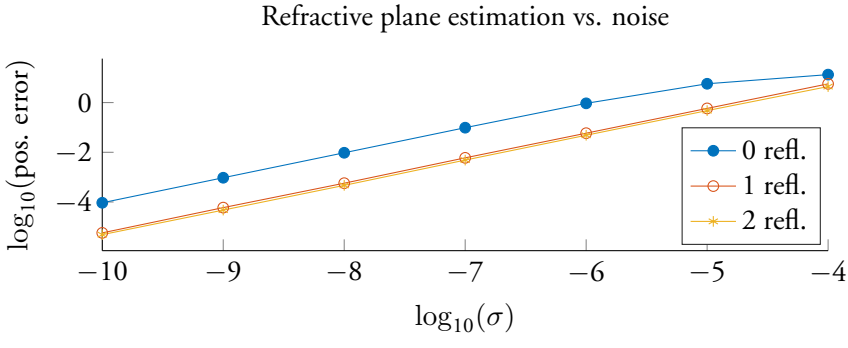


Figure VIII.6: The median of error in position of the refractive plane for three scene settings as a function of the noise variance σ^2. For each σ, zero-mean Gaussian noise with variance σ^2 is added to the normalized image coordinates, and the planes are estimated given slightly perturbed initial plane parameters. Note that the difference between one and two reflective planes is small.

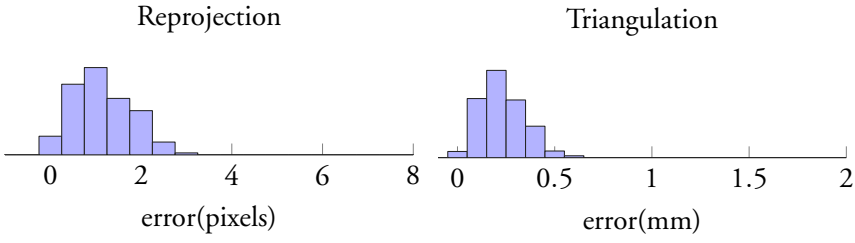


Figure VIII.7: The distribution of re-projection errors (left) and estimated scene points position errors (right) for a real experiment using a checkerboard.

5.2 Real checkerboard experiments

The setup presented in Section 6 was used for testing the performance of the method on real data. A checkerboard of known dimensions was submerged in the water in an aquarium overlooked by four cameras, as presented in Fig VIII.8, in order to evaluate the performance of the system. The checkerboard was moved in a range of approximately 200 mm to 400 mm from the cameras. The cameras have been pre-calibrated and initial estimates of the planes of the aquarium are used. The method of optimizing the refractive and reflective planes, presented in Section 4 and synthetically evaluated in Section 5.1, was applied to observed image points from the scene.

The re-projection errors obtained during the estimation are presented in Fig. VIII.7. In addition, the relative positions of the estimated scene points are analyzed as follows. Since the absolute pose of the checkerboard is unknown, the estimated scene points are transformed by an euclidean transformation to the local coordinate system of the checkerboard. The distribution of distances between the transformed estimated scene points and the checkerboard points is presented in Fig VIII.7.

6 Application – tracking small aquatic organisms

The proposed system is applied for a biological experiment on *Daphnias* (small aquatic organisms), where the behavioral response to e.g. ultraviolet radiation is studied. A setup consisting of an aquarium overlooked by four cameras arranged as vertical stacked stereo-pairs was used to acquire videos, as depicted in Fig VIII.8. The cameras were pre-calibrated and the refractive plane was esti-

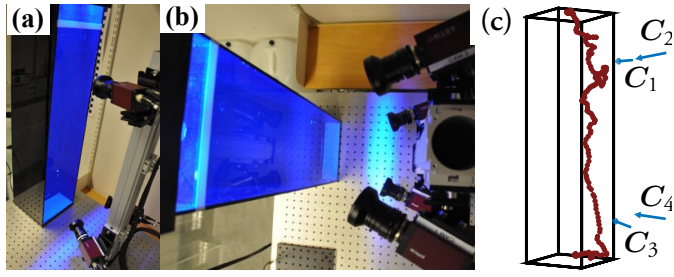


Figure VIII.8: (a) and (b): Side and top view of the experimental setup, composed of an aquarium and four cameras arranged as vertical stacked stereo-pairs that overlook the entire aquarium from the side. (c): The track of one *Daphnia*, the model of the aquarium and the four cameras.

mated, as described in Section 4. Note that since the cameras approximately lie in a plane which is approximately parallel to the refractive surface, a good initial estimate of the refractive surface normal can be made. Measuring the distance from the refractive plane to the cameras by ruler enables initiation of the fourth parameter of the refractive plane. Since the aquarium is rectangular and distances between parallel surfaces are known, initial values of all planes can be made with sufficient precision.

Videos of *Daphnia* were recorded and the *Daphnia* were subsequently detected. The *Daphnia*'s image points were input into the framework and their corresponding 3-d positions were estimated as explained in Section 3.4. The track of one *Daphnia* is shown in Fig VIII.8.

7 Conclusion

We have presented a method that can be used for the tracking of objects within a scene where both flat refractions and flat reflections are present. Evaluations on both synthetic data and real data have confirmed that the information that is found in the reflections can be used to improve the performance of estimating the refractive and reflective planes and the position of objects in the scene. Lastly, we have presented an application of the system that produces qualitatively good results.

References

[1] Amit Agrawal, Srikumar Ramalingam, Yuichi Taguchi, and Visesh Chari. "A theory of multi-layer flat refractive geometry". In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 3346–3353.

[2] Giuseppe Bianco, Vincenzo Botte, Laurent Dubroca, Maurizio Ribera d'Alcalà, and Maria Grazia Mazzocchi. "Unexpected regularity in swimming behavior of Clausocalanus furcatus revealed by a telecentric 3D computer vision system". In: *PloS one* 8.6 (2013), e67640.

[3] Jean-Yves Bouguet. "Camera calibration toolbox for matlab". In: (2004).

[4] Jonathan Cachat, Adam Stewart, Eli Utterback, Peter Hart, Siddharth Gaikwad, Keith Wong, Evan Kyzar, Nadine Wu, and Allan V Kalueff. "Three-dimensional neurophenotyping of adult zebrafish behavior". In: *PloS one* 6.3 (2011), e17597.

[5] Visesh Chari and Peter Sturm. "A Theory of Refractive Photo-Light-Path Triangulation". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2013.

[6] Mikael T Ekvall, Giuseppe Bianco, Sara Linse, Heiner Linke, Johan Bäckman, and Lars-Anders Hansson. "Three-dimensional tracking of small aquatic organisms using fluorescent nanoparticles". In: *PloS one* 8.11 (2013), e78498.

[7] Jason Gedge, Minglun Gong, and Yee-Hong Yang. "Refractive epipolar geometry for underwater stereo matching". In: *Computer and Robot Vision (CRV), 2011 Canadian Conference on*. IEEE. 2011, pp. 146–152.

[8] Christopher Geyer and Kostas Daniilidis. "Mirrors in motion: Epipolar geometry and motion estimation". In: *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE. 2003, pp. 766–773.

[9] Christopher Geyer and Kostas Daniilidis. "Structure and motion from uncalibrated catadioptric views". In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE. 2001, pp. I–279.

[10] Sebastian Haner and Kalle Åström. “Absolute Pose for Cameras Under Flat Refractive Interfaces”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1428–1436.

[11] Anne Jordt. “Underwater 3D Reconstruction Based on Physical Models for Refraction and Underwater Light Propagation”. PhD thesis. Universitätsbibliothek Kiel, 2013.

[12] Anne Jordt-Sedlazeck and Reinhard Koch. “Refractive Structure-from-Motion on Underwater Images”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2013.

[13] Kiriakos N Kutulakos and Eron Steger. “A theory of refractive and specular 3D shape by light-path triangulation”. In: *International Journal of Computer Vision* 76.1 (2008), pp. 13–29.

[14] Erickson Nascimento, Mario Campos, and Wagner Barros. “Stereo based structure recovery of underwater scenes from automatically restored images”. In: *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*. IEEE. 2009, pp. 330–337.

[15] Yoav Y Schechner and Nir Karpel. “Clear underwater vision”. In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 1. IEEE. 2004, pp. I–536.

[16] Yoav Y Schechner and Nir Karpel. “Recovery of underwater visibility and structure by polarization analysis”. In: *IEEE Journal of Oceanic Engineering* 30.3 (2005), pp. 570–587.

[17] Anne Sedlazeck and Reinhard Koch. “Perspective and non-perspective camera models in underwater imaging–overview and error analysis”. In: *Outdoor and Large-Scale Real-World Scene Analysis*. Springer, 2012, pp. 212–242.

[18] Peter Sturm and Thomas Bonfort. “How to compute the pose of an object without a direct view?” In: *Computer Vision–ACCV 2006*. Springer, 2006, pp. 21–31.

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

Paper IX

Daphnia Response to Multiple Threats from UV-A and Predation

MIKAEL T. EKVALL*[1], TOBIAS PALMÉR[2], GIUSEPPE BIANCO[3], JAN HEUSCHELE[4], JOHAN BÄCKMAN[3], KALLE ÅSTRÖM[2] AND LARS-ANDERS HANSSON[1]

[1] *Aquatic Ecology, Department of Biology, Lund University, Sweden*
[2] *Centre for Mathematical Sciences, Lund University, Sweden*
[3] *Centre for Animal Movement Research, Department of Biology, Lund University, Sweden*

Abstract: Organisms are often faced with multiple threats, which may induce both behavioral and morphological responses. One such behavior is diel vertical migration of zooplankton (DVM), which may be induced by several drivers, including predation, temperature, food availability and also ultraviolet radiation (UVR). In order to disentangle the relative importance of predation risk and UVR for this global phenomenon, we exposed two species of the freshwater cladoceran *Daphnia* to both UVR and a predation threat in the form of either a fish or an invertebrate predator to investigate how *Daphnia* handle these two threats simultaneously. We were unable to identify any significant anti-predator responses in either of the two *Daphnia* species, but both species showed a strong response in both vertical position and speed when exposed to UVR. Our data lends support for the recent transparency-regulator hypothesis arguing that UVR is a stronger driver of DVM than predation in clearwater systems.

1 Introduction

All organisms on Earth are faced with various threats during their lifetime and how these threats are perceived and handled have large effects on the fitness of the organism. Moreover, the threats organisms face often occur simultaneously but can vary both spatially and temporally. In order to maximize the likelihood of surviving, an organism needs to be able to perceive the present threat level and also respond accordingly. A common threat is predation, and in order to avoid being eaten an organism has to recognize the presence of a predator and then respond in a way that reduces the probability of being consumed [9]. Such behavioral responses include e.g. hiding or altering of speed [13].

Zooplankton in the world's lakes and oceans are known for their large vertical migrations that occur on a daily basis, the so-called diel vertical migration

(DVM). Originally this behavior was assigned as a response to the threat of predation, where organisms moved away from the well-lit surface waters during day, migrating into deeper and darker waters where the risk posed from visual predators is reduced (see e.g. [5, 15, 19, 22]). During recent years the importance of detrimental ultraviolet radiation (UVR) has also been proposed as a proximate cue driving the diel vertical migration (reviewed in [3, 21]). As UVR attenuates with depth [8, 14] downward migration provides a refuge that protects the organisms from the dangerous radiation. The freshwater zooplankter *Daphnia* has been shown to respond behaviorally both to UVR (e.g. [7, 12, 16]) and predation [1, 4]. Dodson [1] early on showed that some *Daphnia* species are able to distinguish between different predators and respond differently to the presence of these. Hence, as both the predator evasion hypothesis and UV avoidance hypothesis may explain the DVM of *Daphnia* we here aim at quantifying the behavioral response of *Daphnia* when exposed to these two threats simultaneously. Presence of visual predators (e.g. fish) have been shown to induce downward migration in zooplankton, whereas invertebrate predators may induce reversed migration where the organisms favor surface waters during daytime and migrate down during night (e.g. [10]). In this study we expose *Daphnia magna* and *Daphnia pulex* to UVR alone and in presence of a pelagic (fish) or a benthic (invertebrate) predator, respectively, in order to quantify potential trade-offs in the response when faced with simultaneous threats from different directions. We did this by tracking the individual *Daphnia* responses using a recently developed nanoparticle labeling technique [2]. We hypothesized that *Daphnia* will migrate downward when exposed to UVR alone and in combination with a fish predator. However, when exposed to UVR and a benthic invertebrate predator we hypothesize that *Daphnia* may reduce the migration in order to minimize exposure to the predator.

2 Materials and methods

Daphnia magna originating from Lake Bysjön (55.6753 latitude, 13.5452 longitude) and *Daphnia pulex* originating from Lake Dalby quarry (55.6666 latitude, 13.3500 longitude) were used throughout the experiment. Cultures of both *D. magna* and *D. pulex* were kept in separate 30 L plastic aquaria with a light:dark cycle of 14:10 hours and fed *ad libitum* three times per week with an algal suspension dominated by *Scenedesmus* sp.

To quantify the potential trade-off in *Daphnia* threat response we exposed

them to UVR in the absence and presence of both a benthic invertebrate predator or a fish predator. As invertebrate predator we used damselfly larvae (*Calopteryx* sp.) collected in an artificial stream in Lund, southern Sweden. Damselfly larvae were kept in a 20 L aquarium and fed with a mixture of *D. magna* and *D. pulex*. A preliminary 24h feeding trial experiment conducted in 1L glass jars revealed that per capita ingestion rate of damselfly larvae were 4.60 ± 2.89 (mean $\pm$ SD, $n = 5$) *D. magna*. In the fish predator treatment we used a ninespine stickleback (*Pungitius pungitius*) caught in the same stream as the damselfly larvae. Sticklebacks were kept in a 20 L aquaria and fed $\sim$ 50 *D. magna* and 25 *D. pulex* per individual three times per week.

Prior to the trials we labeled three *D. magna* and three *D. pulex* with fluorescent nanoparticles (quantum dots) according to Ekvall et al. [2]. Labeling using fluorescent nanoparticles makes it possible to track the movements of the *Daphniids* through the fluorescence emitted from the nanoparticles upon excitation. In order to separate the two species we used nanoparticles fluorescent at 585 nm (yellow) and 655 nm (red) wavelength. Labeling color was switched between trials using a randomized block design. All experiments were performed in an aquarium with the dimensions ($L \times W \times H$) $0.2 \times 0.2 \times 0.85$ m, filled with 30 L of water, resulting in a 0.75 m water column. In order to minimize reflections in the aquaria walls, the bottom and three walls were made of sanded black Plexiglas whereas the fourth wall was made of transparent Plexiglas. The nanoparticles were excited by integrating eight light-emitting diodes (LED) array with peak emission at 465 nm (VANQ Technology) in a custom-made aquaria top (0.3×0.3 m).

To ensure the presence of predator cues in the water we pre-treated the water by leaving either one damselfly larva or a stickleback in a 25 L holding tank 24 hours prior to the start of the experiments. Throughout the trials a live predator was put at the bottom of the tank using a small plastic net cage ($70 \times 50 \times 50$ mm) that was lowered down just before the start of the experiment. The experiments were video recorded using four synchronized digital cameras (Pike F-210C, Allied Vision Technologies GmbH) equipped with 8 mm focal length lens (SPACE Inc. VHF8MK) and arranged as two stereo-pairs along one side of the aquarium. Each experiment lasted for a total of six minutes and was built up by two phases each lasting three minutes. *Daphniids* were allowed to acclimatize in the aquaria with the excitation light turned on during 15 minutes prior to the start of video the recording. During phase one only the top mounted excitation light used for exciting the nanoparticles was switched on, this was then followed

by three minutes where a UVR threat was added by switching on a UVR LED array (peak emission at 380 nm, VANQ Technology) mounted at the top of the aquarium. As control we used the same design as described above with the removal of the predator (i.e. UVR exposure only). Each treatment was replicated six times with new inexperienced *Daphniids* and the videos were tracked to obtain three-dimensional (3D) information regarding the position and speed of the organisms.

The vertical position and speed of the *Daphnia* was evaluated using linear mixed effect models (nlme package [11]) with UVR phase (before or during UVR exposure), predator (No predator, Fish or Damselfly) and species (*D. magna* or *D. pulex*) as fixed factors and the respective response variable as dependent variable. The model was reduced to the minimum adequate model using stepwise backward deletion. To control for individual and replicate we added individual ID as a random factor nested under replicate ID. All statistics were calculated based on median values for the two UVR phases using R, version 3.1.2 [17].

3 Results

By analyzing the vertical position data using a linear mixed effect model we found that UVR had a strong effect on the vertical position of both species of *Daphnia* ($F_{1, 83}=39.55$, $p<0.001$) and that the vertical response did not differ between the two *Daphnia* species ($F_{1, 74}=0.624$, $p=0.4319$) nor between predator treatments ($F_{2, 15}=0.232$, $p=0.7959$), Fig. IX.1. The final model using stepwise backward deletion contained only species and UV as explanatory factors.

Speed differed between the two species ($F_{1, 74}=57.24$, $p<0.001$), with *D. magna* being much faster than the smaller *D. pulex*, Fig. IX.2. Speed was also affected by UVR ($F_{1, 83}=4.035$, $p=0.0478$), but not significantly by the presence of any of the predators ($F_{2, 15}=2.510$, $p=0.1147$).

4 Discussion

Although previous studies have found that *Daphnia* species are able to perceive and respond to the presence of predators in the surrounding (e.g. [1, 20]) and respond behaviorally [3, 5] we found no significant effects of the presence of neither a fish nor an invertebrate predator when it comes to the vertical position and speed of *D. magna* and *D. pulex* (Fig. IX.1 and Fig. IX.2), possibly caused by large individual variance with respect to predator response. However, in contrast to the

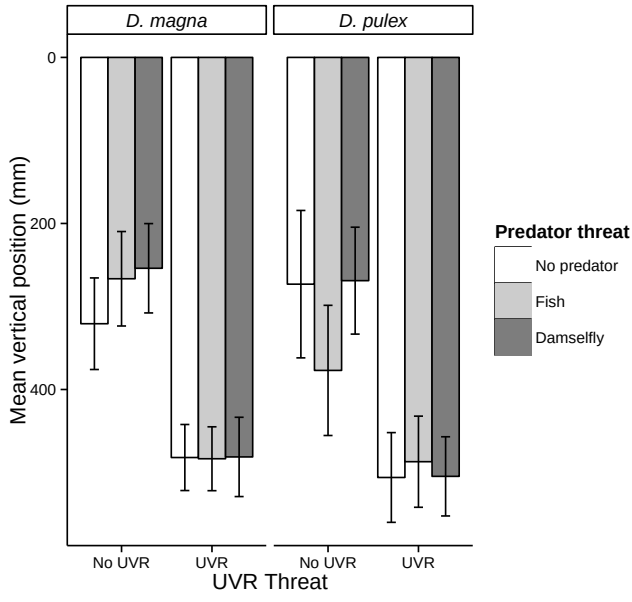


Figure IX.1: Vertical position of *D. magna* (left) and *D. pulex* (right) before and during exposure to ultraviolet radiation (UVR Threat). Different colors identify the different predator threats: No predator (white), Fish (light grey) and Damselfly (dark grey). Bars and whiskers denote mean +/- SE.

effects from predators, the response to UV radiation was immediate and strong for both *Daphnia* species and both with respect to vertical position and speed. These results are in line with the results presented by Hylander [6] who evaluated the behavioral response of zooplankton after exposing zooplankton communities to UVR and predator cues during a five months mesocosm experiment. The authors found that *Daphnia* responded strongly to an UVR threat but found no response to the presence of predator cues from fish [6]. However, it was shown very early by e.g. Dodson [1] that *Daphnia* are able to respond differently to different predators and that they responded more to predators which they co-occurred in their lake of origin. Our *D. magna* culture has been kept as a laboratory culture for more than 100 generations and hence the animals have not experienced predators during this time, suggesting that they might now be less reactive to predator scent. On the other hand, the *D. pulex* clone was brought in directly from a na-

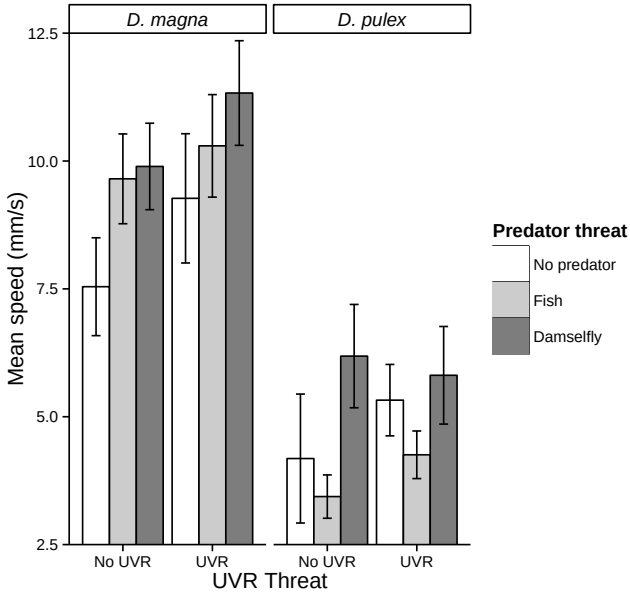


Figure IX.2: Speed of *D. magna* (left) and *D. pulex* (right) before and during exposure to ultraviolet radiation (UVR Threat). Different colors identify the different predator threats: No predator (white), Fish (light grey) and Damselfly (dark grey). Bars and whiskers denote mean +/- SE.

tural system, and may, accordingly, be expected to have a more recent experience of predation, which was also the case (Fig. IX.1 and Fig. IX.2). The observed difference in speed between the two *Daphnia* species is likely attributed to the size difference of the two species where the larger *D. magna* have a higher capacity for reaching higher speeds compared to the smaller *D. pulex*.

Our results do to some extent confirm the proposed transparency-regulator hypothesis [20], which state that UVR is the strongest structural driver for DVM in clearwater systems. All our experiments were conducted in clear water and based on the transparency-regulator hypothesis we would expect to find a strong response to UVR but not to predation, which is in line with the observed responses. Evidence for this was also presented by Tiberti and Iacobuzio[18] who compared zooplankton distribution in clearwater alpine lakes with and without predators. They observed that the distribution did not differ between lakes with

and without predators and that the observed patterns where likely driven by UVR.

In conclusion we here show that *Daphnia magna* without experience from predators did not respond behaviorally to the presence of neither a fish nor an invertebrate predator, whereas *D. pulex* derived directly from a natural system with predators, showed a tendency for diving deeper and swimming slower in the presence of fish than when exposed to a benthic predator (damselfly). However, when also exposed to UVR all tendencies for a predator response was erased and both species immediately escaped to the bottom-water refuge. These laboratory results support the proposed transparency-regulator hypothesis, which argues that predation is a less important cue for DVM compared to UVR in clearwater systems.

References

[1] Stanley I Dodson. “The ecological role of chemical stimuli for the zooplankton: predator-induced morphology in Daphnia”. In: *Oecologia* 78.3 (1989), pp. 361–367.

[2] Mikael T Ekvall, Giuseppe Bianco, Sara Linse, Heiner Linke, Johan Bäckman, and Lars-Anders Hansson. “Three-dimensional tracking of small aquatic organisms using fluorescent nanoparticles”. In: *PloS one* 8.11 (2013), e78498.

[3] Lars-Anders Hansson and Samuel Hylander. “Effects of ultraviolet radiation on pigmentation, photoenzymatic repair, behavior, and community ecology of zooplankton”. In: *Photochemical & Photobiological Sciences* 8.9 (2009), pp. 1266–1275.

[4] Lars-Anders Hansson and Samuel Hylander. “Size-structured risk assessments govern Daphnia migration”. In: *Proceedings of the Royal Society of London B: Biological Sciences* 276.1655 (2009), pp. 331–336.

[5] Graeme C Hays. “A review of the adaptive significance and ecosystem consequences of zooplankton diel vertical migrations”. In: *Migrations and Dispersal of Marine Organisms*. Springer, 2003, pp. 163–170.

[6] Samuel Hylander. “Zooplankton responses to threats from UV and predation”. PhD thesis. Lund University, 2009.

[7] Samuel Hylander, Mikael T Ekvall, Giuseppe Bianco, Xi Yang, and Lars-Anders Hansson. “Induced tolerance expressed as relaxed behavioural threat response in millimetre-sized aquatic organisms”. In: *Proceedings of the Royal Society of London B: Biological Sciences* 281.1788 (2014), p. 20140364.

[8] JTO Kirk. “Optics of UV-B radiation in natural waters.” In: *ERGEB. LIMNOL. ADV. LIMNOL* 43 (1994), pp. 1–16.

[9] Steven L Lima and Lawrence M Dill. “Behavioral decisions made under the risk of predation: a review and prospectus”. In: *Canadian Journal of Zoology* 68.4 (1990), pp. 619–640.

[10] Mark D Ohman, Bruce W Frost, and Edward B Cohen. “Reverse diel vertical migration: an escape from invertebrate predators”. In: *Science* 220.4604 (1983), pp. 1404–1407.

[11] Jose Pinheiro, Douglas Bates, Saikat DebRoy, Deepayan Sarkar, and R Core Team. *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-128. 2016. URL: `http://CRAN.R-project.org/package=nlme`.

[12] Stephan C Rhode, Markus Pawlowski, and Ralph Tollrian. "The impact of ultraviolet radiation on the vertical distribution of zooplankton of the genus Daphnia". In: *Nature* 412.6842 (2001), pp. 69–72.

[13] Nancy M Schoeppner and Rick A Relyea. "Interpreting the smells of predation: how alarm cues and kairomones induce different prey defences". In: *Functional Ecology* 23.6 (2009), pp. 1114–1121.

[14] NM Scully and DRS Lean. "The attenuation of ultraviolet radiation in temperate lakes". In: *Ergebnisse der Limnologie* 43 (1994), pp. 135–135.

[15] Hans-Bernd Stich and Winfried Lampert. "Predator evasion as an explanation of diurnal vertical migration by zooplankton". In: *Nature* 293.5831 (1981), pp. 396–398.

[16] UC Storz and RJ Paul. "Phototaxis in water fleas (Daphnia magna) is differently influenced by visible and UV light". In: *Journal of Comparative Physiology A* 183.6 (1998), pp. 709–717.

[17] R Core Team. *R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 2013*. 2014. URL: `URL:%20http://www.R-project.org/`.

[18] Rocco Tiberti and Rocco Iacobuzio. "Does the fish presence influence the diurnal vertical distribution of zooplankton in high transparency lakes?" In: *Hydrobiologia* 709.1 (2013), pp. 27–39.

[19] Erik Van Gool and Joop Ringelberg. "What goes down must come up: symmetry in light-induced migration behaviour of Daphnia". In: *Hydrobiologia* 491.1-3 (2003), pp. 301–307.

[20] Linda C Weiss, Sebastian Kruppert, Christian Laforsch, and Ralph Tollrian. "Chaoborus and Gasterosteus anti-predator responses in Daphnia pulex are mediated by independent cholinergic and gabaergic neuronal signals". In: *PloS one* 7.5 (2012), e36879.

[21] Craig E Williamson, Janet M Fischer, Stephen M Bollens, Erin P Overholt, and Joanne K Breckenridge. “Toward a more comprehensive theory of zooplankton diel vertical migration: integrating ultraviolet radiation and water transparency into the biotic paradigm”. In: *Limnology and Oceanography* 56.5 (2011), pp. 1603–1623.

[22] Thomas M Zaret and J Samuel Suffern. “Vertical migration in zooplankton as a predator avoidance mechanism”. In: *Limnology and oceanography* 21.6 (1976), pp. 804–813.

Paper X

The Misty Three Point Algorithm for Relative Pose

TOBIAS PALMÉR[1], KALLE ÅSTRÖM[1] AND JAN-MICHAEL FRAHM[2]
[1] *Centre for Mathematical Sciences, Lund University*
[2] *Department of Computer Science, The University of North Carolina at Chapel Hill*

Abstract: There is a significant interest in scene reconstruction from underwater images given its utility for oceanic research and for recreational image manipulation. In this paper we propose a novel algorithm for two view camera motion estimation for underwater imagery. Our method leverages the constraints provided by the attenuation properties of water and its effects on the appearance of the color to determine the depth difference of a point with respect to the two observing views of the underwater cameras. Additionally, we propose an algorithm, leveraging the depth differences of three such observed points, to estimate the relative pose of the cameras. Given the unknown underwater attenuation coefficients, our method estimates the relative motion up to scale. The results are represented as a generalized camera. We evaluate our method on both real data and simulated data.

1 Introduction

The recovery of 3D scene geometry from images has always been one of the core goals of computer vision and has progressed significantly over the last decade [1, 3, 6, 16]. One essential building block of all these system is the ability to successfully estimate the two-view geometry between two overlapping cameras under the assumption of a central perspective camera. This camera model though is only valid for cameras taking photos through air. Hence, this mode of reconstruction is not valid for cameras submerged in water. However, 71% of the earth's surface is covered by oceans and with both the scientific interest in underwater imagery and the now ubiquitous availability of underwater cameras to users, for example through GoPro cameras, performing structure from motion in underwater environments moves into the focus of research [9, 14]. In this paper we target the successful estimation of two-view geometry for underwater cameras, which due to their imaging conditions do not comply with the traditionally used pinhole camera model [14]. In this paper we propose a novel method to enable SFM under such circumstances. In particular we propose a novel minimal solver to enable two-view geometry estimation and a method for relative point depth estimation.

Our proposed method leverages the observation that when looking at photos

for example of an under water ship-wreck, one can get a sense of depth. The reason for this is the easily observable depth dependent attenuation of light under water, which is more significant than in air and weakens as well as distorts an object's color [12, 13]. We leverage this observation along with the 2D correspondences of the projections of the same 3D scene point into two different images to deduce a novel constraint on the relative depth change of the point with respect to the two capturing cameras (see Section 2.3). Given the relative depth changes we then propose a novel three point algorithm that leverages these changes in order to infer the relative camera motion up to scale (see Section 3). In combination this enables us to obtain the two view geometry under a generalized camera model between two images of calibrated underwater cameras. Next we discuss the related work in the area of underwater image based pose estimation.

1.1 Related work

Agrawal et al. [2] present theory and methods for multi-layer flat refractive scenes, using an arbitrary number of interfaces. A method for calibrating such a system is presented, and multi-layer systems are shown to be well approximated by single/two layer systems. A few special cases encountered in flat refractive geometry are solved using polynomial solvers and geometric algebra [4]. They develop near-minimal solvers for the general calibrated and unknown focal length absolute pose cases, i.e. for cases where the scene coordinates are assumed to be known. In this paper we use similar polynomial solving techniques but with unknown scene points.

In the field of underwater structure from motion, Sedlazeck et al. [15] have created a system for simulating deep sea underwater images using physical models for light attenuation, scattering and refraction. Furthermore, Sedlazeck et al. have shown that approximating underwater structure from motion by pinhole cameras produces a systematic error [14]. They propose a method for underwater structure from motion using a virtual camera model [8, 9]. Jordt-Sedlazeck [8] provides a broader overview of the theory and the field of underwater SFM. Their proposed methods uses at least five points for pose estimation (assuming the camera is calibrated) while our method only uses three.

Schechner et.al. [12] propose to recover the scene object radiance, and as a by-product the relative distances in the scene are estimated and yield range maps for the scene. These relative distances are used as a ratio of improvement of the visibility range achieved by the recovery method. Furthermore, Schechner et.al.

[13] reconstruct dense 3-D images of the scene using the depth map and the recovered image. Their approach treats depth maps as a by-product of estimating the scene radiance, whereas our proposed method provides a fusion of their two separate methods.

Queiroz et al.[11] uses the same physical imaging model as in this paper, and leverages color information to improve dense stereo maps. However, they assume a manual pre-calibration of all underwater imagery parameters as well as both geometric and radiometric pre-calibration. The physical imaging model is only used for adding an automated penalty function for the depth map estimation. The algorithm is later automatized by Nascimento et al. [10], but the physical imaging model is still not used for camera pose estimation. Our method, however, automatically estimates the underwater imagery parameters and uses the physical imaging model for pose estimation.

To our knowledge, optimal two-view structure-and-motion using three points and known depth differences has not been solved to date. Neither has the problem of two-view generalized camera structure-and-motion using three points and their colors.

1.2 Innovations

In particular, we propose the following methods:

- We present a novel minimal solver, the Three Point Delta algorithm (TPD), for estimating the relative motion of a generalized camera given three pairs of point correspondences and differences in depth for each such correspondence.
- We present a novel method, the Three Colors Depth Difference algorithm (TCDD), for using a physical model of light propagation under water to estimate relative depth differences.
- We present a novel method, the Misty Three Point algorithm (MTP), for estimating the relative motion of a generalized camera given three pairs of point-and-color correspondences.

Moreover, we show that the methods perform very well numerically and are stable to large amounts of outliers when implemented within a RANSAC-framework, which is enabled by the fact that it only requires 3 points for estimating the relative motion.

1.3 Additional applications

Although the presented minimal solver (TPD) is applied on underwater images in this paper, we note a few more potential areas of application.

- Estimating relative motion using both images and sound.
- Estimating relative motion of cameras in foggy conditions.
- Estimating relative motion from any other pseudo-depth camera.
- Estimating relative orientation of two microphone arrays.

2 Estimating relative depth in underwater imagery

The physical conditions for imaging under water are significantly different than in air leading to a distinct set of challenges for computer vision methods.

One of the main differences is the nonlinear geometry [2, 8, 9] - as cameras usually need to be enclosed in protective housing, which changes the direction of the light rays. In air, rays from the source travel in straight lines to the camera lens. In water, rays are refracted at the surface of the (usually flat) port of the underwater housing. Given that the orientation and position of the refractive surface relative to the camera is known, Snell's law (see Fig X.2) can be used to determine the outgoing rays from a point on the camera sensor. It states that the angle θ_1, relative to the normal of the interface, of an incident ray is related to the angle of the refracted ray θ_2 by the equation

$$\rho_1 \sin\theta_1 = \rho_2 \sin\theta_2, \tag{X.1}$$

where ρ_1 and ρ_2 are the refractive indices of the two media. The fact that this equation is nonlinear is one of the main causes of the challenge in the field of underwater structure from motion. For example, finding the ray in the scene corresponding to an observed pixel is not much more difficult than for regular cameras. Note, though, that it requires that the intrinsics of the camera and the relation to and geometry of the underwater housing is known. However, the reversed problem of finding the projection of a point in space on the image plane is significantly more challenging than for regular cameras.

Another significant difference is the appearance - colors look different under water than when imaged through air [7, 12, 13]. There are two sources of light

that can be observed: light from the scene object and light as part of the ambient light. The observed ambient light part will be termed the *veiling light*. In this paper, the observation of the veiling light is caused by ambient light being scattered towards the cameras line of sight by the particles in the water. The light signal from the scene object can by itself be seen as composed of two components - direct transmission and forward scattering. Whereby the direct transmission is the signal after losing energy due to particles absorbing and scattering (in all directions) the light from the scene object, causing an exponential decrease in energy. The forward scattering is caused by particles scattering the light in small angles relative to the line of sight, causing a blur as well a decrease in energy. A crucial part regarding attenuation is that different wavelengths are attenuated at different rates - red is absorbed 10 times faster than green, which in pure water is absorbed approximately twice as fast as blue. This causes the natural ambient light to be blue/green since the red/yellow components of the natural sunlight are absorbed too quickly to be perceived.

In the remainder of this section, we will present our novel method for taking advantage of this difference in observed color, by assuming that a scene object will have the same unknown radiance when viewed in different images. We then introduce how the observed difference in color between observations of a scene point can be used to estimate differences in distance to the point.

2.1 The physical model

The Jaffe-McGlamery equation is a commonly used equation that models the effect of absorption and scattering [7]. In this paper, a simplified version of the equation (Eq. (X.2)) is used, which does not take the forward scattering into account. The forward scattering can be neglected as the dominant cause for image contrast degradation is the veiling light [12, 13], and furthermore a large part of the forward scattering can be seen as attenuation, which is already captured in the simplified model (Eq. (X.2)).

First, we introduce our notation. The point in space with index k is represented by (U_k, E_k), where the three-vector U_k is the position and the three-vector $E_k = (E_k^r, E_k^g, E_k^b)$ is the radiance of the object in the color channels $\lambda \in \{r, g, b\}$. The observation of the point with index k in camera i is represented in the coordinate system of camera i by (u_{ik}, I_{ik}, p_{ik}), where u_{ik} is the direction of the ray in water, p_{ik} is a point on the ray and $I_{ik} = (I_{ik}^r, I_{ik}^g, I_{ik}^b)$ is the observed color. Note that in the case of the pinhole camera model, all $p_{ik} = 0$,

and in the case of the generalized camera model to account for refractions, the p_{ik} are points on the refractive plane. The depths z_{ik} denote the Euclidean distance from p_{ik} to U_k, i.e. $z_{ik} = \|p_{ik} - (RU_k - Rt)\|_2$, where R is a rotation matrix and t a translation vector that together transform scene points to the local coordinate system of the camera.

The equation used for modelling the physical effects towards the observed colors is the simplified version of the Jaffe-McGlamery equation

$$I_{ik}^{\lambda} = \alpha_{\lambda} \left(E_k^{\lambda} e^{-\eta_{\lambda} z_{ik}} + B_{\infty}^{\lambda} (1 - e^{-\eta_{\lambda} z_{ik}}) \right) + \beta_{\lambda}, \tag{X.2}$$

where I_{ik}^{λ} is the pixel value in color channel λ for camera i and point k, α_{λ} and β_{λ} are color correction coefficients [8], E_k^{λ} is the "true" but unknown color of point k, η_{λ} is the attenuation coefficient of the water, z_{ik} is the distance from the outer refraction plane of camera i to point k and B_{∞}^{λ} is the "veiling light". This is a convex combination of the true color and the veiling light, with added color correction. Note that we here assume that the two cameras have the same color correction coefficients. The equation can be reformulated to:

$$I_{ik}^{\lambda} = \alpha_{\lambda} \left(E_k^{\lambda} - B_{\infty}^{\lambda} \right) e^{-\eta_{\lambda} z_{ik}} + \gamma_{\lambda}, \tag{X.3}$$

where $\gamma_{\lambda} = \alpha_{\lambda} B_{\infty}^{\lambda} + \beta_{\lambda}$ is the observation of the veiling light. Given observations $\{I_{ik}^{\lambda}\}$ and $\{I_{jk}^{\lambda}\}$, $\lambda \in \{g, b\}$, of the color of point k in cameras i and j at depth z_{ik} and z_{jk}, the equation can be reduced to

$$\frac{I_{ik}^{\lambda} - \gamma_{\lambda}}{I_{jk}^{\lambda} - \gamma_{\lambda}} = \frac{\alpha_{\lambda} \left(E_k^{\lambda} - B_{\infty}^{\lambda} \right) e^{-\eta_{\lambda} z_{ik}}}{\alpha_{\lambda} \left(E_k^{\lambda} - B_{\infty}^{\lambda} \right) e^{-\eta_{\lambda} z_{jk}}} = e^{\eta_{\lambda} \Delta z_{ijk}}. \tag{X.4}$$

2.2 Estimating the constant parameters

The red color channel is not used due to the fact that in practical applications the red colors are by practical means completely absent, as discussed in Section 2. Accordingly, there are 4 unknown constant parameters to estimate, and each pairwise correspondence introduces 1 unknown variable (Δz_{ijk}) and provides two constraints by using only the green and blue color channels in Eq. (X.4). This means that to solve for all $4 + n$ unknown constants and variables, at least $n = 4$ pairwise correspondences are needed. However, a scale ambiguity exists, thus η_{λ}

and z_{ik} can be redefined as $\hat{\eta}_\lambda = \eta_\lambda/\eta_g$ and $\hat{z}_{ik} = \eta_g z_{ik}$. Thus for each point correspondence we have the constraints

$$\begin{aligned} I^g_{ik} - \gamma_g &= (I^g_{jk} - \gamma_g)e^{\Delta\hat{z}_{ijk}}, \\ I^b_{ik} - \gamma_b &= (I^b_{jk} - \gamma_b)e^{\hat{\eta}_b\Delta\hat{z}_{ijk}}. \end{aligned} \tag{X.5}$$

Furthermore, solving for γ_λ and reformulating Eq. (X.5) leads to

$$\gamma_\lambda = I^\lambda_{jk} + \frac{1}{1 - e^{\hat{\eta}_b\Delta\hat{z}_{ijk}}}\left(I^\lambda_{ik} - I^\lambda_{jk}\right), \tag{X.6}$$

which is monotonous in $\Delta\hat{z}_{ijk}$. This monotony can be used to show that there does not always exist a real solution $(\Delta\hat{z}_{ijk}, \boldsymbol{\gamma}, \hat{\boldsymbol{\eta}})$.

2.3 The Three Colors Depth Difference algorithm

After introducing the obtained constraint for each point correspondence, we now introduce our proposed depth estimation. The underwater error function for the color channel λ is defined as (rewritten from Eq. (X.4))

$$r_\lambda(\gamma, \eta, \Delta z_{ijk}) = I^\lambda_{ik} - \gamma_\lambda - (I^\lambda_{jk} - \gamma_\lambda)e^{\eta_\lambda\Delta z_{ijk}}, \tag{X.7}$$

and the combined error function for the green and blue channel as

$$r(\gamma, \eta, \Delta z_{ijk}) = r_g^2 + r_b^2. \tag{X.8}$$

The Jacobian for Eq. (X.8) is computed and used in a Gauss-Newton algorithm to find the parameters for which the error is minimized.

Note that if $I^\lambda_{ik} - \gamma_\lambda$ has a different sign than $I^\lambda_{jk} - \gamma_\lambda$ there is no solution to $r_\lambda = 0$ (since $e^x > 0$), and the minimum is found at $\Delta z_{ijk} = 0$. This would correspond to a point being observed beyond infinity. Thus it can be concluded that

$$\min_{\Delta z} r^2_\lambda(\gamma, \eta, \Delta z) \geq (I^\lambda_{ik} - \gamma_\lambda)^2 \tag{X.9}$$

This means that a point in one camera with green and blue color intensities larger than γ will still have color intensities larger than γ when observed in another camera. This lower bound will later be used for fast outlier rejection when implementing the three-point relative pose algorithm within RANSAC (see Section 4.1).

3 The Three Point Delta algorithm

Given three point correspondences in two calibrated generalized cameras as well as the difference in depth (with known scale or, equivalently, either η_g or η_b is known i.e. up to scale), the relative motion of the cameras is sought. Note that in the particular case where the generalized camera is a pinhole camera, an unknown scale of depth differences simply gives a scale ambiguity in the solution. In the general case, an unknown scale of depth differences make the solution non-existing or invalid.

We denote $P_i = \begin{bmatrix} I & 0 \end{bmatrix}$ and $P_j = R \begin{bmatrix} I & -t \end{bmatrix}$ as the ith and jth camera's projection matrices with the ith camera without loss of generality being at the origin and in canonical orientation. Let $\{x_{ik}\}$ and $\{x_{jk}\}$ denote sets of points in the camera planes of the generalized cameras i and j, respectively, corresponding to observations of the scene points $k = 1, 2, 3$, and let $\Delta z_{ijk} = z_{jk} - z_{ik}$ be the measured depth differences of the points between camera i and j. The backward projection of each pair of corresponding points (x_{ik}, x_{jk}) into space can be parametrized by

$$\begin{aligned} p_{ik} + z_{ik} u_{ik}, & \qquad z_{ik} \in \mathbb{R}, \\ R^\top (p_{jk} + z_{jk} u_{jk}) + t, & \qquad z_{jk} \in \mathbb{R}, \end{aligned} \tag{X.10}$$

where p_{ik}, p_{jk}, u_{ik} and u_{jk} are known since the cameras are intrinsically calibrated. In the case where the generalized cameras are cameras enclosed in underwater housings, the p_{ik}'s and p_{jk}'s are points on the outer surfaces of the underwater housing ports, and the u_{ik}'s and u_{jk}'s are the directions into the water.

Since x_{ik} and x_{jk} correspond to the same scene point, there exists z_{ik} and z_{jk} such that $p_{ik} + z_{ik} u_{ik} = R^\top (p_{jk} + z_{jk} u_{jk}) + t$ (using the parametrization from Eq. (X.10)). The unknown depths z_{jk} can be reduced by substituting $z_{jk} = z_{ik} + \Delta z_{ijk}$, which gives the sets of equations

$$\begin{aligned} p_{ik} + z_{ik} u_{ik} = R^\top (p_{jk} + (z_{ik} + \Delta z_{ijk}) u_{jk}) + t, \\ k = 1, 2, 3, \end{aligned} \tag{X.11}$$

where depths z_{ik}, camera j's rotation R and camera j's translation t are the unknown variables that are sought. We propose to parametrize R using quaternions $q = (s, v)$, where s is scalar and v is a three-vector,

$$R = 2(vv^\top - s[v]_\times) + (s^2 - v^\top v) I, \tag{X.12}$$

and adding the constraint $s^2 + v^\top v = 1$ to ensure that the determinant is one, gives a total of 10 equations in 10 unknowns.
The equations are solved by noting that Eq. (X.11) implies that there is a rigid transformation from $p_{ik} + z_{ik}u_{ik}$ to $p_{jk} + (z_{ik} + \Delta z_{ijk})u_{jk}$. Thus the Gramians for the two sets of points $U_k = p_{ik} + z_{ik}u_{ik}$ and $U'_k = p_{jk} + (z_{ik} + \Delta z_{ijk})u_{jk}$ are equal [17]. The Gramian for U is defined as $V^\top V$, where

$$V = \begin{bmatrix} U_2 - U_1, & U_3 - U_1 \end{bmatrix}. \tag{X.13}$$

Inserting the expressions for the U_k's and U'_k's gives the Gramians

$$\begin{aligned} V &= \begin{bmatrix} z_{i2}u_{i2} - z_{i1}u_{i1} + p_{i2} - p_{i1} \\ z_{i3}u_{i3} - z_{i1}u_{i1} + p_{i3} - p_{i1} \end{bmatrix}, \\ V' &= \begin{bmatrix} (z_{i2} + \Delta z_{ij2})u_{j2} - (z_{i1} + \Delta z_{ij1})u_{j1} + p_{j2} - p_{j1} \\ (z_{i3} + \Delta z_{ij3})u_{j3} - (z_{i1} + \Delta z_{ij1})u_{j1} + p_{j3} - p_{j1} \end{bmatrix}. \end{aligned} \tag{X.14}$$

Thus the constraint that the Gramians are equal amounts to

$$V^\top V - (V')^\top (V') = 0, \tag{X.15}$$

and provides three equations that are quadratic in the three unknowns z_{ik}.

It turns out that the coefficients for all z_{ik}^2 terms are zero since the u_{ik}'s and u_{jk}'s are normalized. Thus the equations are of the form

$$\begin{cases} A_{11}xy + A_{14}x + A_{15}y + A_{17} = 0, \\ A_{21}xy + A_{22}xz + A_{23}yz + A_{24}x + A_{25}y + A_{26}z + A_{27} = 0, \\ A_{32}xz + A_{34}x + A_{36}z + A_{37} = 0, \end{cases} \tag{X.16}$$

where x, y and z correspond to z_1, z_2 and z_3. The system in equation (X.16) can be represented as a matrix-vector multiplication $Av = 0$, where

$$A = \begin{bmatrix} A_{11} & 0 & 0 & A_{14} & A_{15} & 0 & A_{17} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} & A_{27} \\ 0 & A_{32} & 0 & A_{34} & 0 & A_{36} & A_{37} \end{bmatrix}, \tag{X.17}$$

and v is the vector of monomials

$$v = (xy, xz, yz, x, y, z, 1). \tag{X.18}$$

Note that each line in Eq. (X.16) has been divided by the coefficient of the leading term. By performing row-operations on the system, it can be simplified to the form $\hat{A}v = 0$, where

$$\hat{A} = \begin{bmatrix} 1 & 0 & 0 & \hat{A}_{14} & \hat{A}_{15} & 0 & \hat{A}_{17} \\ 0 & 1 & 0 & \hat{A}_{24} & 0 & \hat{A}_{26} & \hat{A}_{27} \\ 0 & 0 & 1 & \hat{A}_{34} & \hat{A}_{35} & \hat{A}_{36} & \hat{A}_{37} \end{bmatrix}. \tag{X.19}$$

To solve the system, x times the third equation, y times the second equation and z times the first equation are added, i.e. the equations

$$\begin{cases} xyz + C_{14}xz & +C_{15}yz + C_{17}z & = 0, \\ xyz + C_{24}xy & +C_{26}yz + C_{27}y & = 0, \\ xyz + C_{35}xy & +C_{36}xz + C_{37}x & = 0, \end{cases} \tag{X.20}$$

are added. Since Eq. (X.19) gives reductions from xy,xz and yz to x, y and z, xyz can also be reduced to x, y and z. Thus the system can be formulated as

$$\begin{cases} M_{11}y + M_{12}z + M_{13} = xy, \\ M_{21}y + M_{22}z + M_{23} = xz, \\ M_{31}y + M_{32}z + M_{33} = x, \end{cases} \tag{X.21}$$

i.e. as the eigenvalue equation

$$M \begin{bmatrix} y \\ z \\ 1 \end{bmatrix} = x \begin{bmatrix} y \\ z \\ 1 \end{bmatrix}. \tag{X.22}$$

Thus the depth z_{i1} is an eigenvalue of M, and z_{i2} and z_{i3} are the two first elements of the corresponding eigenvector after normalizing with the third element. Since M is a 3 by 3 matrix, three solutions are found and need to be evaluated by Eq. (X.16).

Assuming that one solution (z_{i1},z_{i2},z_{i3}) to Eq. (X.16) was found, the z_{jk}'s can be computed by $z_{jk} = z_{ik} + \Delta z_{ijk}$. Now that all depths are known, the U_k's can be computed from the z_{ik}'s as in $U_k = p_{ik} + z_{ik}u_{ik}$. Eq. (X.11) gives that $U_k = R^\top(p_{jk} + z_{jk}u_{jk}) + t$ where R and t are the only remaining unknowns, which means that the remaining problem is now to find a projective transformation from U_k to $p_{jk} + z_{jk}u_{jk}$. This problem is solved by the three point resection

method [5], providing up to four solutions for (R, t). All solutions (R, t) that gives negative depths when projecting the U_k's are rejected. Then the distance from each point p_{jk} to the corresponding transformed scene point $RU_k - Rt$ is computed as

$$z'_{jk} = \|(RU_k - Rt) - p_{jk}\|_2. \tag{X.23}$$

If the relative pose (R, t) is valid, then $z_{jk} = z'_{jk}$ must hold. Thus, for each solution (z_{i1}, z_{i2}, z_{i3}) to Eq. (X.16) there are at most four solutions (R, t), which makes a total of up to twelve solutions to be evaluated as described.

In conclusion, we have shown how given only three pairs of corresponding points and their difference in distance to the camera, first the absolute distances can be found and subsequently the object and the relative motion. Furthermore, by combining this method with the method for estimating relative depths given the colors of three pairs of corresponding points, we have found a method for estimating relative motion of a generalized camera given three corresponding points with their associated colors in the images.

4 Robust estimation with the Misty Three Point algorithm

In this section we show how the Misty Three Point (MTP) algorithm can be embedded in a RANSAC-framework using a sequencing of the algorithm that enables fast rejection of an estimate as well as fast rejection of outliers. Given a set of inliers, we also show how the solution can be optimized for all parameters while taking both reprojection errors and the physical model for underwater imaging into account.

4.1 RANSAC

Since the Three Point Delta (TPD) algorithm introduced in Section 3 uses relative depths as input, the Three Colors Depth Difference (TCDD) algorithm defined in Section 2.3 must be the first step of relative pose estimations. In general, the underwater imaging parameters that the TCDD provides are not necessarily feasible for all points, giving an opportunity to ignore those points in the following steps. In addition, there is not always a solution to the depth difference problem, in those cases the estimate based on those points can be instantly rejected. Assuming that the algorithm found feasible parameters, the TPD is then used to

estimate the relative pose. At this point, the scene points are reconstructed by triangulation, and the reprojection errors are computed and used to evaluate how well the estimated pose fits the dataset. Furthermore, the depth differences are also estimated for all feasible points, adding one more measure of how well the estimate fits the dataset.

4.2 Bundle adjustment

The results from the RANSAC method are optimized in a bundle adjustment algorithm that seeks to minimize the errors both for the reprojections and the underwater imagery equations using all variables. In particular, it optimizes for the relative motion of the generalized cameras and the 3-D points. We define the residual vector, whose norm is the target for minimization, by combining the reprojection errors with Eq. (X.3):

$$r(\boldsymbol{x}, \boldsymbol{\theta}) = \begin{bmatrix} \cdots \\ I^g_{1k} - \hat{E}^g_k e^{-\eta_g z_k} - \gamma_g \\ I^b_{1k} - \hat{E}^b_k e^{-\eta_g z_k} - \gamma_b \\ I^g_{2k} - \hat{E}^g_k e^{-\eta_g z'_k} - \gamma_g \\ I^b_{2k} - \hat{E}^b_k e^{-\eta_g z'_k} - \gamma_b \\ U_k - z_k u_k \\ RU_k - Rt - z'_k u'_k \\ \cdots \\ s^2 + v^\top v - 1 \end{bmatrix} \tag{X.24}$$

where radiance $\hat{E}^\lambda_k = \alpha(E^\lambda_k - B^\lambda_\infty)$ and R is parametrized as in Eq. (X.12). The partial derivatives of r are computed analytically and are used for minimizing $\|r(\boldsymbol{x}, \boldsymbol{\theta})\|$ using Gauss-Newton's method with respect to $\boldsymbol{x}$.

5 Experiments

We are evaluating our proposed method on simulated and real data. First, we show that the Three Point Delta (TPD, see Section 3) algorithm is numerically stable with respect to estimating the depth. Secondly, we show that the Three Colors Depth Difference (TCDD, see Section 2.3) algorithm and the combination of the TCDD algorithm followed by the TPD algorithm (The Misty Three Point

algorithm, see Section 4) are also numerically stable. Thirdly, we show that the RANSAC based algorithm introduced in Section 4.1 handles outliers well.

The simulated experiments are then complemented with real data experiments. Note however, that for the real experiments we do not have ground truth to compare with, hence, we compare them only qualitatively.

5.1 Synthetic data

In order to test the system, a scene consisting of three 3D-points with RGB-colors and two cameras was generated along with underwater imaging parameters. The points were projected into the cameras and the known distance from each 3D-point was used to find the attenuated colors according to Eq. (X.2). To test the accuracy of our proposed pose estimation, the MTP algorithm was used to estimate the relative pose, using the projected points and the observed colors. The relative translational error and the relative angular error for the estimate were computed and are presented in Fig X.5. Similarly, the TPD was used to estimate the relative pose using the projected points and the given difference in depth, the results of which are also presented in Fig X.5. These results show that we can accurately estimate the camera motion. Furthermore, the MTP algorithm was used to estimate the absolute distance given the three projected points and their colors, and the TPD algorithm provided estimates using the three projected points and the given change in depth. The relative errors of the estimates are presented in Fig X.5. The numerical accuracy of the TCDD algorithm was tested by providing the observed colors of the three points. The estimated difference in depth as well as the underwater imaging parameters are compared to the ground truth in Fig X.5.

5.2 Real data

For the purpose of validating that the models and methods are practically applicable, a video taken from diving an underwater ship wreck was downloaded from YouTube. The video was captured using a GoPro camera enclosed in a flat port protective underwater housing. Despite the unknown calibration, we could still evaluate our method qualitatively and provide a proof-of-concept, as seen in Fig X.8 and Fig X.9. As can be seen in the figures, the MTP succeeds in finding qualitatively correct corresponding points, reconstructed structure and also qualitatively correct motion.

6 Conclusion

We have proposed a novel method for estimating relative motion given three points and their colors. Using physical models for underwater imaging, we have shown that the depth information that is present in the observed colors can be estimated and used in practice. We also demonstrate that our algorithms perform quantitatively well in the synthetic experiments when compared to ground truth (see Fig X.5), when exposed to noise (see Fig X.7), and in a RANSAC-framework when exposed to high ratios of outliers (see Fig X.6). Furthermore, we have shown that our proposed method is qualitatively correct in the reconstruction of structure and estimation of motion (see Fig X.8 and Fig X.9).

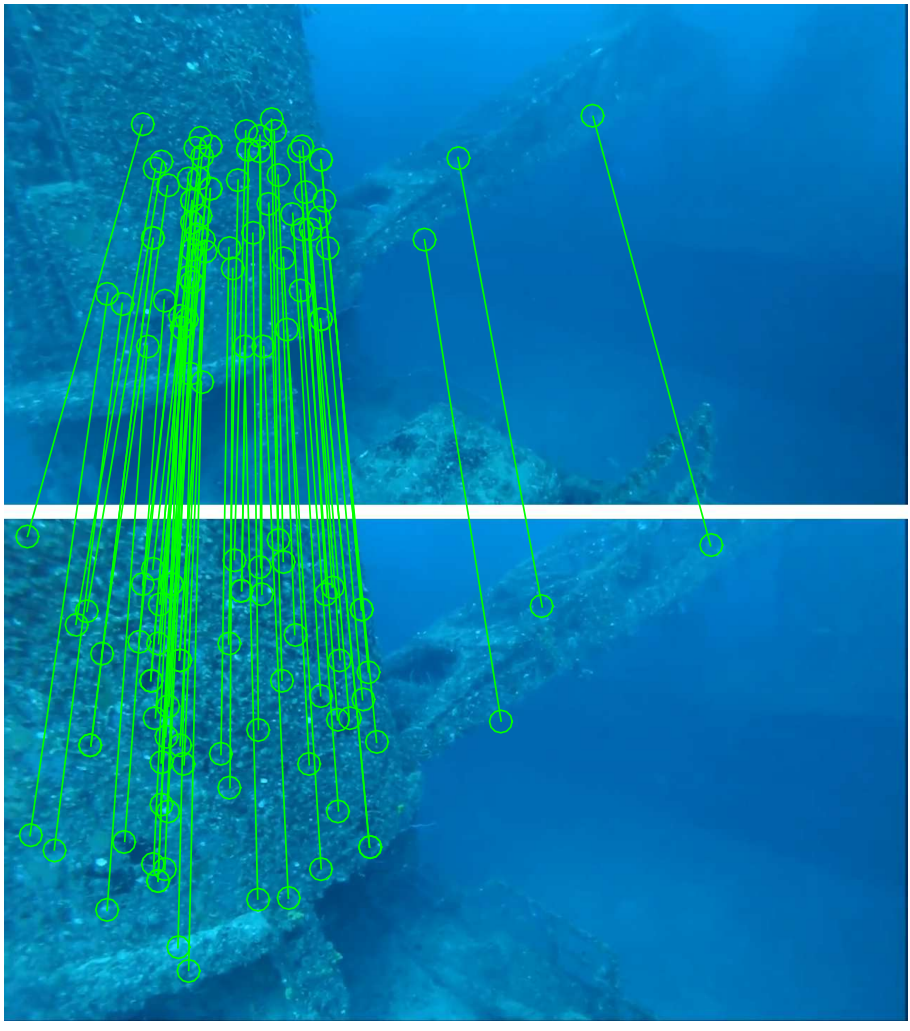


Figure X.1: Real images with corresponding points found using the Misty Three Point algorithm proposed in this paper.

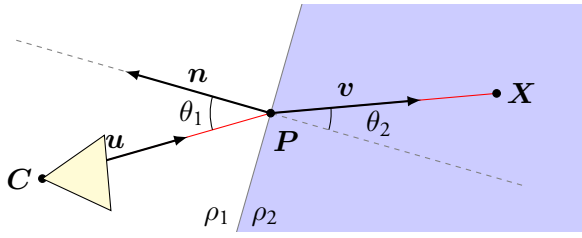


Figure X.2: Snell's law. A ray originating from the camera center C with direction $\boldsymbol{u}$ undergoes a change of direction according to $\rho_1 \sin\theta_1 = \rho_2 \sin\theta_2$. This causes the usually linear equations for projections, for example, to become nonlinear and much harder to solve.

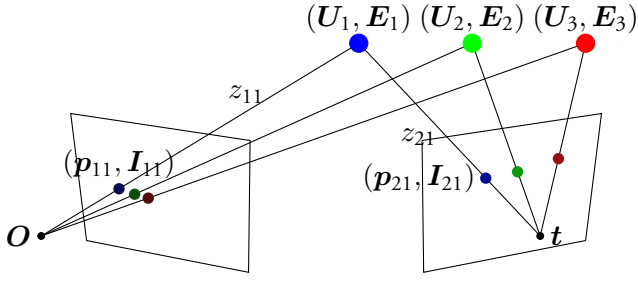


Figure X.3: The relative pose problem solved in this paper. Two cameras centered at the origin and t, respectively, are observing three points of unknown position and unknown color. Note that we are not only using the direction u_{ik} from each observed point, but also the observed color. The relative depths $\Delta_{ik} = z_{jk} - z_{ik}$ are also crucial parts of this method. In the left cameras, with larger distance to the objects, the colors look very similar. In the right camera, however, the observed colors are more similar to the actual color of the object. These differences is what enables the estimation of depth differences.

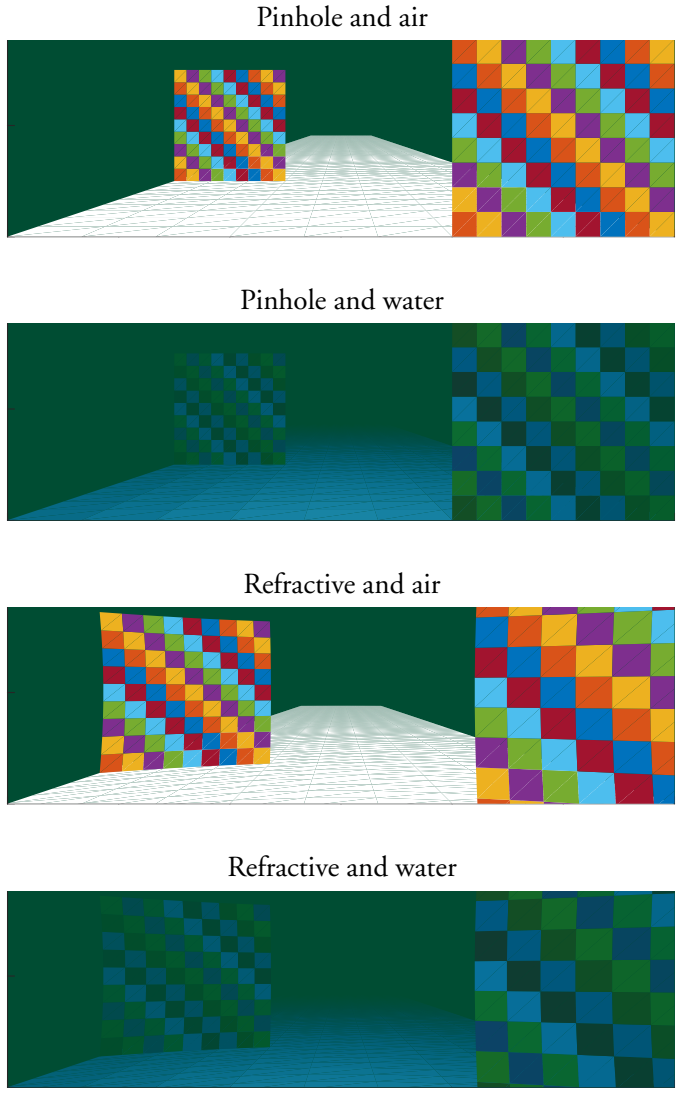


Figure X.4: A visualization of the imagery effects of refraction and attenuation. The top row images shows projections of the scene in a pinhole camera and the bottom row shows projections of the scene in refractive cameras. The left column images shows the images with no loss of colors, and the right row shows the images with attenuated colors according to Eq. (X.2).

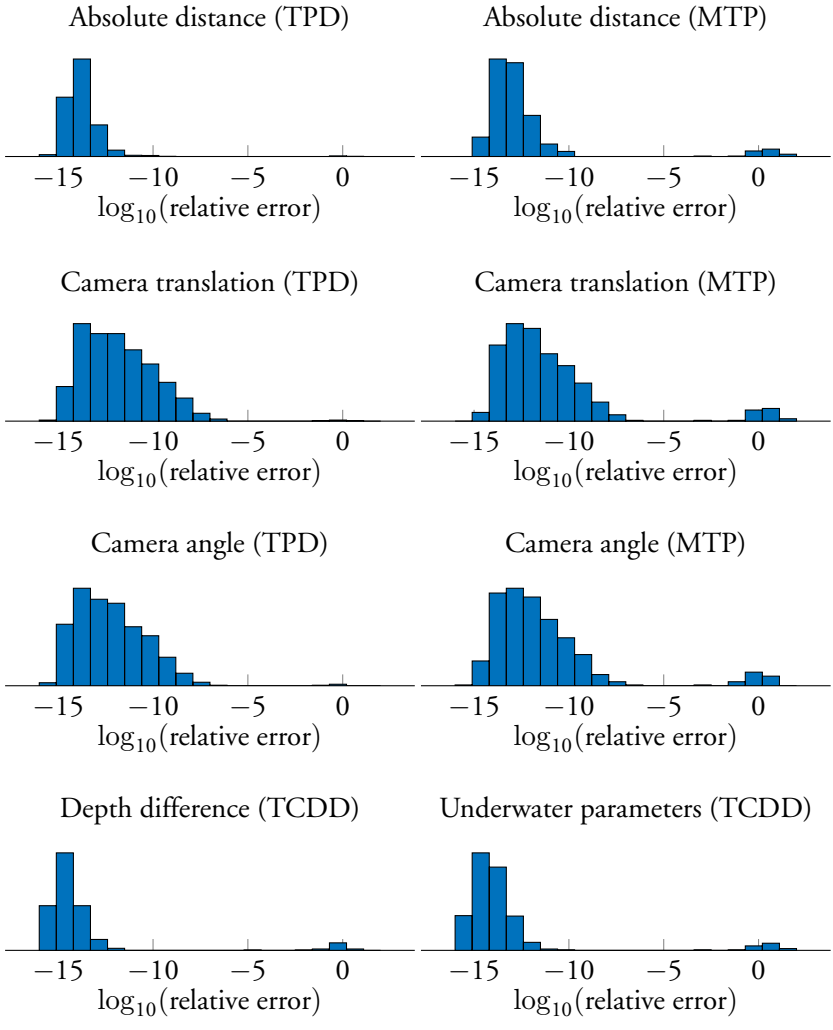


Figure X.5: Distribution of solver error relative to ground truth, computed over 10000 random problem instances. The top row shows the relative error in estimated distances for the Three Point Delta (TPD) algorithm and the Misty Three Point (MTP) algorithm. The second row shows the relative error in estimated camera direction for TPD and MTP. The third row shows the relative error in estimated camera translation for TPD and MTP. Lastly, the bottom row presents the relative errors in estimated depth difference (left) and underwater imaging parameters (right) performed by the Three Colors Depth Difference (TCDD) algorithm.

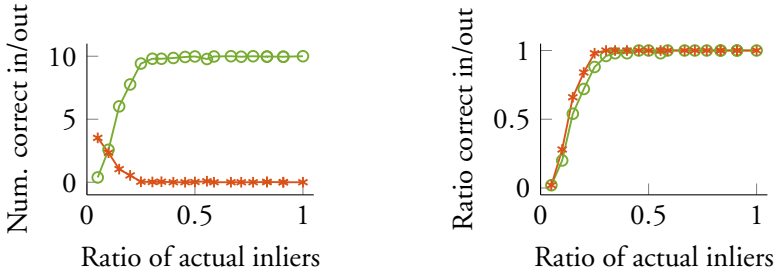


Figure X.6: The performance of inlier/outlier classification of the system given a varying number of outliers, while the number of inliers is fixed to 10. In the left figure, the number of classified inliers from the true inlier group (green) and the outlier group(red) is shown. In the right figure, the rate of classifying outliers as outliers (red) and inliers as inliers (green) is plotted. The test for each inlier ratio was repeated 50 times, each of which consisted of 1000 RANSAC iterations, and the plotted lines are the mean values over those tests.

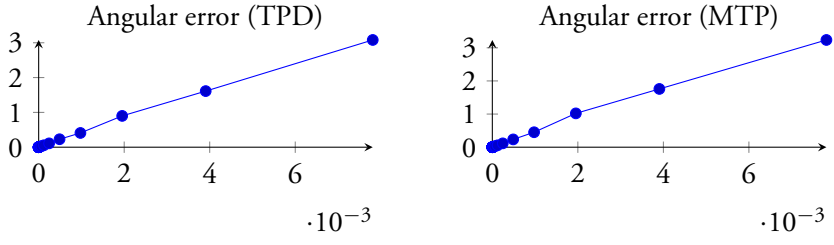


Figure X.7: Median of angular error in degrees as a function of noise variance. For each noise level x, 1000 random problem instances were generated, and normal distributed noise with zero mean and x^2 variance was added to the generated points.

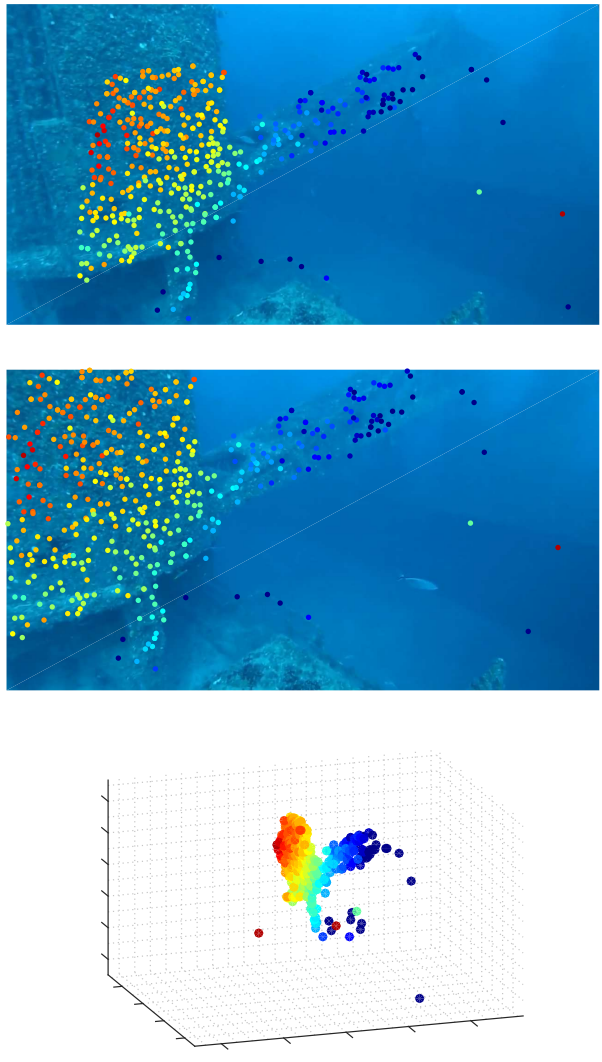


Figure X.8: A 3D-reconstruction of a scene estimated with the MTP. Detected point correspondences are colored by distance to the origin, where the closest points are colored red and the farthest away are blue. The top figure and the middle figure show the detected points plotted on the used images and the bottom figure shows the reconstructed 3D-points.

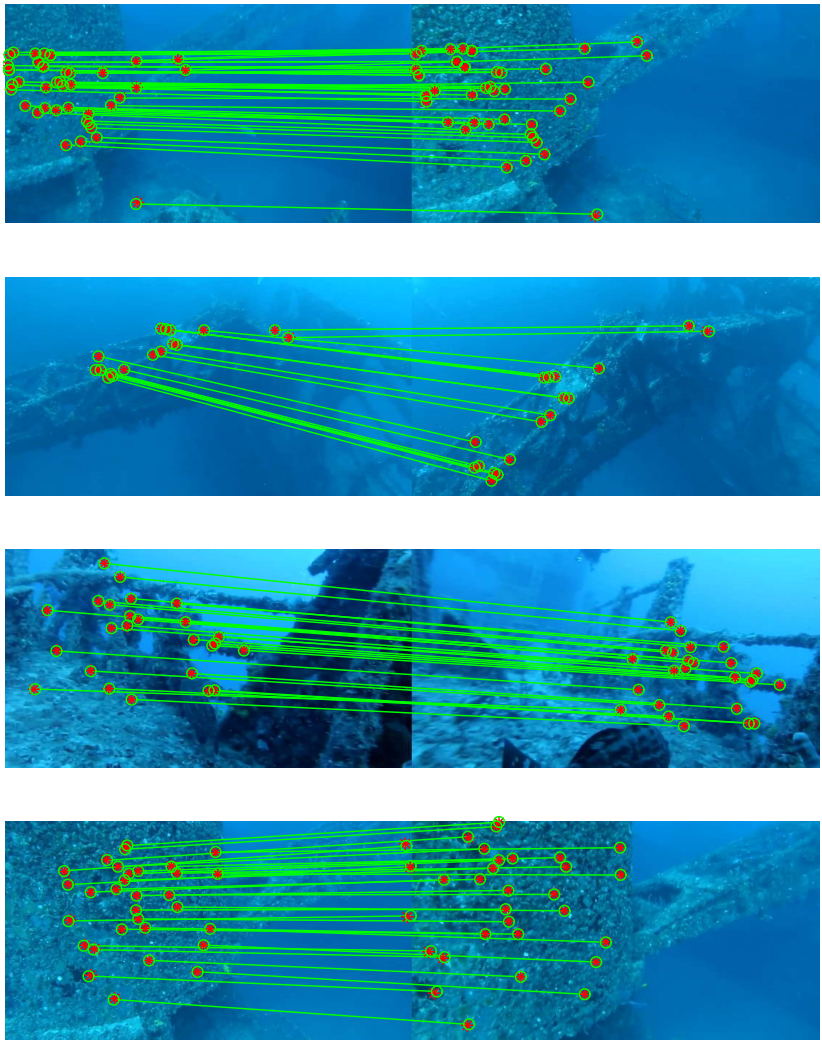


Figure X.9: Some examples of images and corresponding points found using the Misty Three Point algorithm. Reprojections of estimated scene points are given as red asterisks.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven Seitz, and Richard Szeliski. “Building rome in a day”. In: *Proc. Int. Conf. on Computer Vision*. 2009.

[2] Amit Agrawal, Srikumar Ramalingam, Yuichi Taguchi, and Visesh Chari. “A theory of multi-layer flat refractive geometry”. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 3346–3353.

[3] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. “Building Rome on a cloudless day”. In: *Proc. European Conf. on Computer Vision*. 2010.

[4] Sebastian Haner and Kalle Åström. “Absolute Pose for Cameras Under Flat Refractive Interfaces”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1428–1436.

[5] Robert M Haralick, Chung-nan Lee, K Ottenburg, and Michael Nölle. “Analysis and solutions of the three point perspective pose estimation problem”. In: *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR’91., IEEE Computer Society Conference on*. IEEE. 1991, pp. 592–598.

[6] Jared Heinly, Johannes Lutz Schönberger, Enrique Dunn, and Jan-Michael Frahm. “Reconstructing the World* in Six Days *(As Captured by the Yahoo 100 Million Image Dataset)”. In: *Proc. Conf. Computer Vision and Pattern Recognition*. 2015.

[7] Jules S Jaffe. “Computer modeling and the design of optimal underwater imaging systems”. In: *Oceanic Engineering, IEEE Journal of* 15.2 (1990), pp. 101–111.

[8] Anne Jordt. “Underwater 3D Reconstruction Based on Physical Models for Refraction and Underwater Light Propagation”. PhD thesis. Universitätsbibliothek Kiel, 2013.

[9] Anne Jordt-Sedlazeck and Reinhard Koch. “Refractive Structure-from-Motion on Underwater Images”. In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2013.

[10] Erickson Nascimento, Mario Campos, and Wagner Barros. “Stereo based structure recovery of underwater scenes from automatically restored images”. In: *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*. IEEE. 2009, pp. 330–337.

[11] Jose P Queiroz-Neto, Rodrigo Carceroni, Wagner Barros, and Mario Campos. “Underwater stereo”. In: *Computer Graphics and Image Processing, 2004. Proceedings. 17th Brazilian Symposium on*. IEEE. 2004, pp. 170–177.

[12] Yoav Y Schechner and Nir Karpel. “Clear underwater vision”. In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 1. IEEE. 2004, pp. I–536.

[13] Yoav Y Schechner and Nir Karpel. “Recovery of underwater visibility and structure by polarization analysis”. In: *IEEE Journal of Oceanic Engineering* 30.3 (2005), pp. 570–587.

[14] Anne Sedlazeck and Reinhard Koch. “Outdoor and Large-Scale Real-World Scene Analysis: 15th International Workshop on Theoretical Foundations of Computer Vision, Dagstuhl Castle, Germany, June 26 - July 1, 2011. Revised Selected Papers”. In: ed. by Frank Dellaert, Jan-Michael Frahm, Marc Pollefeys, Laura Leal-Taixé, and Bodo Rosenhahn. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. Chap. Perspective and Non-perspective Camera Models in Underwater Imaging – Overview and Error Analysis, pp. 212–242. ISBN: 978-3-642-34091-8. DOI: 10.1007/978-3-642-34091-8_10. URL: http://dx.doi.org/10.1007/978-3-642-34091-8_10.

[15] Anne Sedlazeck and Reinhard Koch. “Simulating deep sea underwater images using physical models for light attenuation, scattering, and refraction”. In: (2011).

[16] Noah Snavely, Steven Seitz, and Richard Szeliski. “Photo tourism: exploring photo collections in 3D”. In: (2006).

[17] Gale Young and Alston S Householder. “Discussion of a set of points in terms of their mutual distances”. In: *Psychometrika* 3.1 (1938), pp. 19–22.