Flexible Force-Vision Control for Surface Following using Multiple Cameras

Olsson, Tomas; Johansson, Rolf; Robertsson, Anders

Published in:
Proceedings. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004).

DOI:
10.1109/IROS.2004.1389450

2004

Proceedings of 2004 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 28 - October 2, 2004, Sendai, Japan

Flexible Force-Vision Control for Surface Following using Multiple Cameras

Tomas Olsson
Department of Automatic Control
Lund Inst. of Tech., Lund University
SE-221 00 Lund, Sweden
Tomas.Olsson@control.lth.se

Rolf Johansson
Department of Automatic Control
Lund Inst. of Tech., Lund University
SE-221 00 Lund, Sweden
Rolf.Johansson@control.lth.se

Anders Robertsson
Department of Automatic Control
Lund Inst. of Tech., Lund University
SE-221 00 Lund, Sweden
Anders.Robertsson@control.lth.se

Abstract— **A flexible method for six-degree-of-freedom combined vision/force control for interaction with a stiff uncalibrated environment is presented. An edge-based rigid-body tracker is used in an observer-based controller, and combined with a six-degree-of-freedom force- or impedance controller. The effect of error sources such as image space measurement noise and calibration errors are considered. Finally, the method is validated in simulations and a surface following experiment using an industrial robot.**

I. Introduction

Even in modern robot control systems, there are difficulties instructing robots how to deal with variations in their environment. In order to handle large deviations from the nominal setup, external sensing capabilities are crucial. In particular, force sensing capabilities can be useful when robots are required to interact with their environment. There has also been a growing interest in vision based control, since a lot of information can be obtained from visual data. With the increasing computational power available today, there is a potential for robust visual servoing systems that operate at camera frame rate. However, the nature and limited accuracy of vision based control makes it less suitable for controlling the interaction between a robot and a potentially stiff environment. Therefore, an interesting approach is to combine force control and visual servoing in a multi-sensor control system.

A. Visual control

Position-based visual servoing techniques require some type of pose estimation, since the feedback law is defined in the workspace, rather than directly in the image. Accurate and robust tracking and estimation of the position of rigid objects using measurements from one or several cameras has been an active research topic for many years. Many methods for rigid body tracking work by minimizing some measure of the image space error as a function of the unknown position and orientation parameters, using standard non-linear optimization methods, or Kalman filtering techniques [1], [2]. In [3] it was suggested that the output from the pose estimation should be used as input for the Kalman filter, in order to avoid the high computational complexity required when the output is a high-dimensional image-space vector. The position and orientation can be parameterized in different ways, such as roll-pitch-yaw angles [1], quaternions or dual quaternions [2]. There are also various ways to measure the image space error, the most common measurements are the positions of point features [1], lines, or point-to-contour errors [4], [5]. The point-to-contour method has a major advantage in that it does not require the exact matching of features, only the error in the normal direction at a number of points on a contour. This only requires a one-dimensional search for features (edges).

B. Force and impedance control

Impedance control aims to achieve a certain dynamical behavior of the end-effector position and orientation in response to external forces [6]. Using an inner motion control structure, the motion controller is made to track the pose of the so called *compliant frame*, denoted by Σ_c. The impedance relation is a relation on the form

$$\mathbf{M}_I \frac{d^2\mathbf{x}}{dt^2} + \mathbf{D}_I \frac{d\mathbf{x}}{dt} + \mathbf{K}_I \mathbf{x} = \mathbf{f} \qquad (1)$$

where $\mathbf{x}$ is the relative position of the compliant frame with respect to the reference frame Σ_d, $\mathbf{f}$ is the external force, and $\mathbf{M}_I$, $\mathbf{D}_I$ and $\mathbf{K}_I$ are positive definite matrices which can be interpreted as the effective mass, damping and stiffness, respectively.

C. Previous work

Over the last decade, some work on vision/force control has been presented. In [7] three different strategies are presented, traded control, hybrid control, and shared control. In [8] the use of vision/impedance control is proposed, and demonstrated in a peg-in-hole insertion experiment. The method presented in [9] uses Mason's task frame and a high level task description to determine how to use each sensor. An application of position-based force/vision control in flexible assembly is presented in [10], with a demonstration of the use of a single eye-in-hand camera for mating of moving parts. Other hybrid and adaptive techniques have been presented, for instance in [11], [12], [13].

Problem formulation

In this paper we demonstrate how to achieve high performance six-degree-of-freedom combined vision/force control for interaction with a stiff uncalibrated environment. It is shown how a process with linear dynamics in

0-7803-8463-6/04/$20.00 ©2004 IEEE

task space, together with a standard formulation for an edge-based rigid body tracker, can be used to design an observer with linear error dynamics. The effect of error sources such as image noise and geometrical calibration errors are considered and analyzed. Finally, experiments and simulations are used to validate the approach.

II. Controller for force/vision control

A. Modeling

We assume that we have M cameras placed in fixed locations, viewing a target object whose position and orientation with respect to some fixed (world) coordinate system should be estimated. The position and orientation is parameterized as $\mathbf{x}_p \in \mathbb{R}^n$ where typically $n = 6$. The image data is compressed into a vector $\mathbf{y} \in \mathbb{R}^N$, usually the image space coordinates of corners, edges and other features. If the geometry of the target is known, $\mathbf{x}_p$ and $\mathbf{y}$ are related by the projection equations of the cameras

$$\mathbf{y} = \mathbf{h}(\mathbf{x}_p) \tag{2}$$

which is usually a very complex non-linear function. The most commonly used camera model is the homogeneous form pinhole camera projection equation, which in our case becomes

$$\mathbf{y}_i = \mathbf{h}_i(\mathbf{x}_p) = \frac{1}{Z_i}\mathbf{K}_c\mathbf{T}_{cw}\mathbf{T}_{wo}(\mathbf{x}_p)\mathbf{X}_i \tag{3}$$

where $\mathbf{K}_c$ is a matrix of intrinsic camera parameters, $\mathbf{X}_i$ is the coordinates of the point in an object-centered coordinate system, Z_i is the depth of the point in the camera, and $\mathbf{T}_{cw}$ and $\mathbf{T}_{wo}(\mathbf{x}_p)$ are the homogeneous coordinate transformation matrices between the target object and the world coordinate system, and between the world coordinate system and the camera, respectively. The parameterization $\mathbf{x}_p$ of $\mathbf{T}_{wo}$ is the unknown position/orientation to be estimated. In the following, the camera position $\mathbf{T}_{cw}$ is assumed to be known. This is not a restriction in situations where only the relative pose of two tracked objects is to be controlled, since the position of the world coordinate system is arbitrary. Only the relative positions of the cameras need to be accurately calibrated, in order to be able to relate measurements from different cameras.

We assume that the task space dynamics of the motion controlled manipulator can be modeled as a linear system, which together with the nonlinear measurement equation give the Wiener-type model

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u} \\ \mathbf{y} = \mathbf{h}(\mathbf{x}_p) \end{cases} \tag{4}$$

where $\mathbf{u}$ is the input, and $\mathbf{x}$ is the state vector typically consisting of the position $\mathbf{x}_p$ and velocity of the end-effector in the task space, and possibly other states depending on the model of the dynamics. For relatively low bandwidth systems, such as vision based controllers, the approximation of the complex closed loop robot dynamics with a linear system of relatively low order is reasonable. The output $\mathbf{y}$ in Eq. (4) is the vector of image features obtained from the images, and $\mathbf{h}$ is given by Eq. (3), for each point.

For the pinhole camera, the task space position $\mathbf{x}_p$ could in general be obtained from a pose estimation as

$$\mathbf{x}_p = \mathbf{h}^{-1}(\mathbf{y}) \tag{5}$$

and used in a feedback control law in order to control the task space position [3]. The pose estimation is typically performed using some type of iterative least-squares optimization algorithm, using the previous position as a starting point for the iteration. However, near singular configurations, where the Jacobian of $\mathbf{h}$ loses rank, the pose estimation becomes very inaccurate [5]. An example of such a situation is when the relative depth of the object points is small, for instance when viewing a small object at a long distance from the camera. In such cases, a very accurate estimation of the depth Z_i of each point may be required in order to maintain stability [14].

B. Vision based observer

By exploiting the dynamic model in Eq. (4), we could obtain extra robustness and noise suppression. Since almost all real-time pose estimation algorithms work by updating an initial guess or prediction of the state, we use a state observer on the form

$$\frac{d\hat{\mathbf{x}}}{dt} = \mathbf{F}\hat{\mathbf{x}} + \mathbf{G}\mathbf{u} + \mathbf{K}\mathbf{J}^{\dagger}(\mathbf{y} - \mathbf{h}(\hat{\mathbf{x}})) \tag{6}$$

where $\mathbf{J}^{\dagger}$ is the pseudo inverse of the Jacobian $\mathbf{J} = d\mathbf{h}/d\mathbf{x}$ from the linearization of the measurement equation.

Eq. (3) can be differentiated with respect to $\mathbf{x}$ and linearized around $\hat{\mathbf{x}}$, and the equations for multiple feature points can be stacked to give the linearized equation

$$\Delta\mathbf{y} = \mathbf{y} - \mathbf{h}(\hat{\mathbf{x}}) \approx \mathbf{J}(\hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}}) \tag{7}$$

where $\mathbf{J}$ is the Jacobian of the projection equation, which can now be used directly in Eq. (6).

In the case of edge measurements, only the distance between the predicted and real edges in the normal direction of the contour is measurable. The corresponding equations are obtained by projecting the image space errors onto the normal as in [4], [5], giving us the alternative equations

$$\Delta\mathbf{y}_{(N)} = \mathbf{N}^T(\mathbf{y} - \mathbf{h}(\hat{\mathbf{x}})) \approx \mathbf{N}^T\mathbf{J}(\hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}}) = \mathbf{J}_{(N)}(\hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}}) \tag{8}$$

where $\mathbf{N}$ is a block diagonal $n \times n$ matrix, where the blocks are the edge normal directions at the n different measurement points along the contour (Fig. 1).

It is clear that in general the accuracy of the estimation will improve with the number of image measurements N. If we assume that the errors in the image measurements ε_y can be modeled as Gaussian, spatially uncorrelated white noise with variance σ^2, a useful approximation of the effective measurement error covariance $\varepsilon_x = \mathbf{J}^{\dagger}\varepsilon_y$ can be obtained as

$$\mathrm{E}[\varepsilon_x\varepsilon_x^T] = \mathrm{E}[\mathbf{J}^{\dagger}\varepsilon_y(\mathbf{J}^{\dagger}\varepsilon_y)^T] = (\mathbf{J}^T\mathbf{J})^{-1}\sigma^2 = (\sum_{i=1}^{M}\mathbf{J}_i^T\mathbf{J}_i)^{-1}\sigma^2 \tag{9}$$

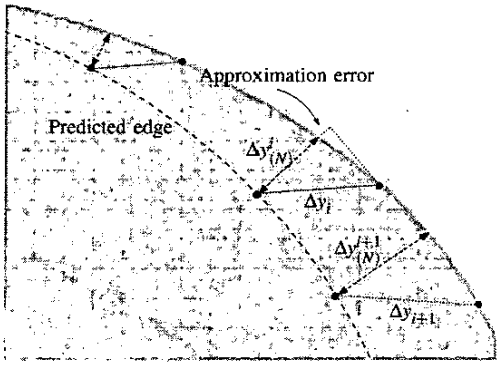

Fig. 1. Edge detection in the normal direction of the predicted edges.

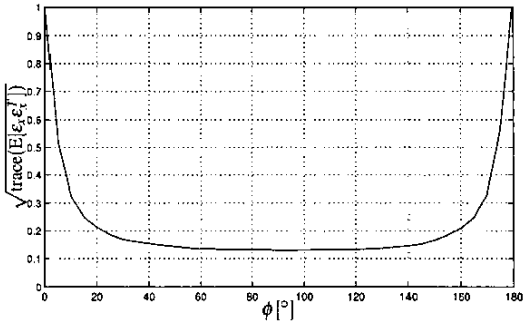

Fig. 2. The square root of the trace of $\mathrm{E}[\varepsilon_x \varepsilon_x^T]$ for three cameras with normalized intrinsic parameters, placed at an angle of ϕ.

where the Jacobian has been partitioned into the individual Jacobians for M cameras as $\mathbf{J}^T = [\mathbf{J}_1^T, \mathbf{J}_2^T, \cdots \mathbf{J}_M^T]^T$. If a large number of edge search points are distributed evenly along the visible edges of the object, we can approximate Eq. (9) with

$$\mathrm{E}[\varepsilon_x \varepsilon_x^T] \approx \sigma^2 \left(\sum_{i=1}^{M} N_i \Phi_i(\hat{\mathbf{x}}) \right)^{-1} \qquad (10)$$

where Φ_i is a positive semidefinite $n \times n$ matrix independent of N_i, which shows that the covariance of the measurement error is a direct function of the number, N_i, of edge detection points placed in each camera, giving the number of edge searches required to achieve a given measurement accuracy [15]. In addition, Eqs. (9) or (10) could be used as approximate covariance matrices in a (time-varying) Kalman filter. By performing the estimation update in this way, the tracking algorithm can be distributed over several processors, each responsible for the image processing and Jacobian creation and inversion for one camera. The measurements $\mathbf{J}_i^{\dagger} \Delta \mathbf{y}_i$ are combined in the central Kalman filter, using the covariance estimates for each camera.

The dependence of the covariance matrices on the Jacobian, calculated at $\hat{\mathbf{x}}$, is clear from Eq. (9). In general, the right singular vectors $\mathbf{V}$ of $\mathbf{J} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ will indicate the degrees of freedom with the largest error sensitivity. At distances significantly larger than the dimension of the object, this is usually translation in the Z-direction of the camera. By placing multiple cameras at different orientations, considerable improvements in the measurement covariance can be obtained. This advantage of a multi-camera setup is illustrated in Fig. 2, where the trace of the resulting covariance matrix is plotted as a function of the angle ϕ between the z-axes of three simulated cameras. The three cameras were placed at a distance 5L from the object, of dimensions $0.5L \times 0.5L \times 0.5L$, with 200 measured edge locations in each camera. In order to obtain the same measurement accuracy with a single camera, it would be necessary to either decrease σ by increasing the image quality and/or resolution, or to use cameras with longer focal length, thereby reducing the field of view.

C. Force and impedance control

The force controller implemented is a general impedance controller with inner motion control [6]. In the controller, the impedance equation is divided into *translational impedance* and *rotational impedance* as

$$\mathbf{M}_{I,t} \frac{d^2 \mathbf{t}_{dc}}{dt^2} + \mathbf{D}_{I,t} \frac{d \mathbf{t}_{dc}}{dt} + \mathbf{K}_{I,t} \mathbf{t}_{dc} = \mathbf{f} - \mathbf{f}_r \qquad (11)$$

$$\mathbf{M}_{I,r} \frac{d^2 \phi_{dc}}{dt^2} + \mathbf{D}_{I,r} \frac{d \phi_{dc}}{dt} + \mathbf{K}_{I,r} \phi_{dc} = \mathbf{T}^T(\phi_{dc})(\tau - \tau_r) \qquad (12)$$

where $\mathbf{f}$ and τ is the force and torque exerted by the environment on the end-effector, $\mathbf{f}_r$ and τ_r is an optional reference force/torque pair, $\mathbf{t}_{dc} = \mathbf{t}_c - \mathbf{t}_d$ is the relative translation between the reference frame and the compliant frame, and ϕ_{dc} is the Euler XYZ angles extracted from the rotation matrix $\mathbf{R}_{dc} = \mathbf{R}_c^T \mathbf{R}_d$, and $\mathbf{T}$ is a Jacobian matrix relating the angular velocity to the time derivative of ϕ_{dc}.

D. Combined vision/force controller

The block diagram for the system under vision/force control is shown in Fig. 3. The desired trajectory of the tool is defined relative to the target object, whose position is estimated from the image data. The visual feedback controller generates a reference position and velocities in order to follow the desired trajectory, based on the estimated relative position of the end-effector and the target. The force controller updates the position and velocity according to Eqs. (11) and (12), and the new references are sent to the built-in robot motion control.

We assume a decoupled dynamic model of a velocity controlled manipulator

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{v}_c \\ \Delta \mathbf{y} = \mathbf{h}(\mathbf{C}\mathbf{x}, \mathbf{C}\hat{\mathbf{x}}) \approx \mathbf{J}(\mathbf{C}\hat{\mathbf{x}})(\mathbf{C}\mathbf{x} - \mathbf{C}\hat{\mathbf{x}}) \end{cases} \qquad (13)$$

where $\mathbf{x} = \begin{bmatrix} \mathbf{x}_p^T & \dot{\mathbf{x}}_p^T \end{bmatrix}^T$ is the state vector, $\mathbf{v}_c$ the commanded velocity, $\Delta \mathbf{y}$ is the normal distances between the search points and the image edges, $\hat{\mathbf{x}}$ is the estimated state, and the system matrices are given by

$$\mathbf{F} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & -\omega \mathbf{I} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{0} \\ \omega \mathbf{I} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}. \qquad (14)$$

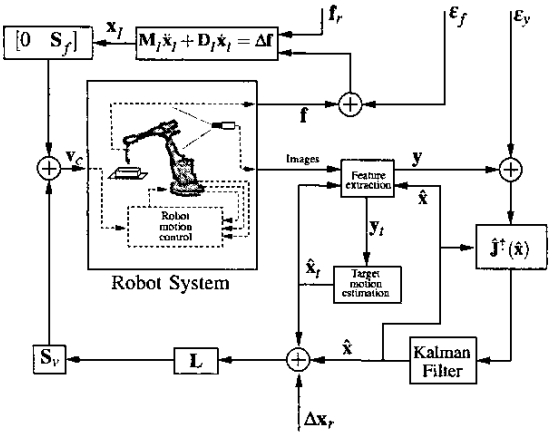

Fig. 3. Block diagram showing the structure of the combined force/vision control system. Inputs are the reference signals $\Delta\mathbf{x}_r$ and $\mathbf{f}_r$ and the output disturbances ε_y and ε_f.

By assumption $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ is small, and the approximation in Eq. (13) holds locally. A state observer on the form of Eq. (6) is given by

$$\frac{d\hat{\mathbf{x}}}{dt} = \mathbf{F}\hat{\mathbf{x}} + \hat{\mathbf{G}}\mathbf{v}_c + \mathbf{K}\hat{\mathbf{J}}^{\dagger}\Delta\mathbf{y} = \mathbf{F}\hat{\mathbf{x}} + \hat{\mathbf{G}}\mathbf{v}_c + \mathbf{K}\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}) \quad (15)$$

where the estimation $\hat{\mathbf{J}}$ is assumed to be equal to the true image Jacobian $\mathbf{J}$, and the difference $\Delta\mathbf{G} = \mathbf{G} - \hat{\mathbf{G}}$ models the calibration errors in the geometric model of the manipulator object frame. Using a force controller given by

$$\begin{aligned}\frac{d\mathbf{x}_I}{dt} &= \begin{bmatrix}\mathbf{0} & \mathbf{I} \\ \mathbf{0} & -\mathbf{M}_I^{-1}\mathbf{D}_I\end{bmatrix}\mathbf{x}_I + \begin{bmatrix}\mathbf{0} \\ \mathbf{M}_I^{-1}\end{bmatrix}\mathbf{S}_f(\mathbf{f} - \mathbf{f}_r) \\ &= \mathbf{F}_I\mathbf{x}_I + \mathbf{G}_I\mathbf{S}_f(\mathbf{f} - \mathbf{f}_r) \end{aligned} \quad (16)$$

together with the hybrid vision/force controller

$$\mathbf{v}_c = \mathbf{S}_v\mathbf{L}(\mathbf{x}_r - \hat{\mathbf{x}}) + \mathbf{S}_f\begin{bmatrix}\mathbf{0} & \mathbf{I}\end{bmatrix}\mathbf{x}_I, \quad (17)$$

where $\mathbf{S}_v$ and $\mathbf{S}_f$ are the hybrid *selection matrices*, and a contact model

$$\mathbf{f} = -\mathbf{k}_e\mathbf{C}\mathbf{x}, \quad (18)$$

we can write down the equations for the closed loop system as

$$\frac{d}{dt}\begin{bmatrix}\mathbf{x} \\ \tilde{\mathbf{x}} \\ \mathbf{x}_I\end{bmatrix} = \mathbf{F}_c\begin{bmatrix}\mathbf{x} \\ \tilde{\mathbf{x}} \\ \mathbf{x}_I\end{bmatrix} + \mathbf{G}_c\begin{bmatrix}\mathbf{x}_r \\ \mathbf{f}_r\end{bmatrix} \quad (19)$$

with

$$\mathbf{F}_c = \begin{bmatrix}\mathbf{F} - \mathbf{G}\mathbf{S}_v\mathbf{L} & \mathbf{G}\mathbf{S}_v\mathbf{L} & \mathbf{G}\begin{bmatrix}\mathbf{0} & \mathbf{S}_f\end{bmatrix} \\ -\Delta\mathbf{G}\mathbf{S}_v\mathbf{L} & \mathbf{F} - \mathbf{K}\mathbf{C} + \Delta\mathbf{G}\mathbf{S}_v\mathbf{L} & \Delta\mathbf{G}\begin{bmatrix}\mathbf{0} & \mathbf{S}_f\end{bmatrix} \\ -\mathbf{G}_I\mathbf{S}_f\mathbf{k}_e\mathbf{C} & \mathbf{0} & \mathbf{F}_I\end{bmatrix} \quad (20)$$

and

$$\mathbf{G}_c = \begin{bmatrix}\mathbf{G}\mathbf{S}_v\mathbf{L} & \mathbf{0} \\ \Delta\mathbf{G}\mathbf{S}_v\mathbf{L} & \mathbf{0} \\ \mathbf{0} & -\mathbf{G}_I\mathbf{S}_f\end{bmatrix} \quad (21)$$

E. Stability

When $\Delta\mathbf{G} = \mathbf{0}$ and $\hat{\mathbf{J}} = \mathbf{J}$, for small $\tilde{\mathbf{x}}$ the observer error is locally described by the stable system $\dot{\tilde{\mathbf{x}}} = (\mathbf{F} - \mathbf{K}\mathbf{C})\tilde{\mathbf{x}}$. In the force controlled directions, a stable and well-damped response in the measured contact force is obtained by proper tuning of $\mathbf{M}_I$ and $\mathbf{D}_I$. In practice, the possible choices of $\mathbf{M}_I$ and $\mathbf{D}_I$ are also limited by sensor noise, unmodeled dynamics, and resonances in the tool and workpiece.

In the case where the estimation of the Jacobian in Eq. (15) is not exact, the state observer (15) and the resulting closed loop system may become unstable. For a purely kinematic robot model $\dot{\mathbf{x}}_p = \mathbf{v}_c$ with $\mathbf{x} = \mathbf{x}_p$, $\mathbf{F} = \mathbf{0}$ and $\mathbf{G} = \mathbf{I}$, local stability of the observer with $\mathbf{K} = \mathbf{I}$ is guaranteed as long as the matrix $\hat{\mathbf{J}}^{\dagger}\mathbf{J}(\hat{\mathbf{x}})$ is positive definite. Near singularities, this is satisfied only if a very accurate estimation $\hat{\mathbf{J}}$ is available, and small errors in the intrinsic camera parameters or point depth distribution can cause the system to become unstable [14]. However, an observer for the dynamical system in Eq. (13) will need additional constraints on $\hat{\mathbf{J}}$ and $\mathbf{J}$ for stability, as stability can not be guaranteed even if $\hat{\mathbf{J}}^{\dagger}\mathbf{J}(\hat{\mathbf{x}})$ is positive definite.

F. Implementation

The vision controller and observer are designed as a stationary Linear Quadratic controller with Kalman filter, based on discretized version of the dynamic model in Eq. (13) sampled at 33 ms. The force controller in Eq. (16) is discretized at a sampling period of 4 ms. The force controller runs on a Power-PC G4 processor, connected to the internal robot motion controller over the PCI bus [16]. The image processing, and calculation and inversion of the image Jacobian runs on a separate 2 GHz Pentium 4, which communicates with the controller on the Power-PC using Ethernet.

The tracking algorithm running on the PC is summarized in Fig. 4. Two objects are tracked, the stationary target and the manipulator object, assumed to be rigidly attached to the robot hand. The tracker states are initialized by manually indicating features and using an approximate pose estimation algorithm. At each sample time all images are read from the cameras, the pre-calculated control signal is sent to the main controller using Ethernet, and measurement vectors and Jacobians are calculated. The total hybrid control signal is then read back from the Power-PC, and is used to calculate the state estimate and vision-based part of the control signal. A fast hidden-line removal technique based on BSP tree representations of the objects is used to predict locations of visible edges in the next set of images.

III. Results

A. Vision/force controlled surface following

By combining force control with visual feedback as described in Section II-D, we could achieve surface following that is independent on the workpiece calibration accuracy. Experiments with this scenario have been performed using an ABB Irb2400 industrial robot equipped with a rolling

1) Initialize state and data structures
2) Send pre-computed visual command velocity $\mathbf{u}_v$ to the main controller on the Power-PC
3) Capture images from each camera and perform image pre-processing
4) Search for image edges around the predicted edges of the manipulator and target, and build measurement vectors $\Delta\mathbf{y}$ and $\Delta\mathbf{y}_t$
5) For each edge measurement, build one row of the corresponding Jacobian $\mathbf{J}$ or $\mathbf{J}_t$
6) Read effective control signal from Power-PC, given by $\mathbf{v}_c := \mathbf{S}_v\mathbf{u}_v + \mathbf{S}_f \begin{bmatrix}\mathbf{0} & \mathbf{I}\end{bmatrix}\mathbf{x}_f$
7) Update the state estimate for the manipulator using the model $\hat{\mathbf{x}} := \mathbf{F}_d\hat{\mathbf{x}} + \hat{\mathbf{G}}_d\mathbf{v}_c + \mathbf{K}\mathbf{J}^\dagger\Delta\mathbf{y}$
8) Update estimated target position using one Gauss-Newton iteration $\hat{\mathbf{x}}_t := \hat{\mathbf{x}}_t + \mathbf{J}_t^\dagger\Delta\mathbf{y}_t$
9) Predict visible edges during the next sample, by performing hidden line removal based on the predicted positions $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}_t$
10) Calculate $\mathbf{x}_r = \hat{\mathbf{x}}_t + \Delta\mathbf{x}_r$, and pre-calculate the vision based part of the control signal $\mathbf{u}_v := \mathbf{L}(\mathbf{x}_r - \hat{\mathbf{x}})$
11) Wait for next sample time and repeat from Step 2)

Fig. 4. Algorithm for tracking and control of relative position.

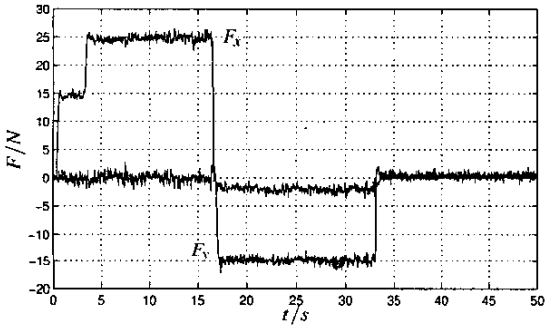

Fig. 6. Measured contact force during vision guided force control.

grinding tool, in contact with a metal box with dimensions $40 \times 40 \times 10$ cm. Experiments were first performed using two Sony digital cameras, and later repeated with an extra camera, using the resource allocation algorithm presented in [15], see Fig. 5.

The robot makes stable contact with the workpiece under vision guided impedance control, and when contact has been established the control switches to parallel vision/force control as described in Section II-D, while the tool moves across the surface at around 10 mm/s. The resulting force can be seen in Fig. 6. At time $t = 3$ s the force reference was changed from 15 N to 25 N in the x-direction of the tool. At time $t = 17$ s the tool reaches a corner, and the force reference changes to 15 N in the negative y-direction. The combined stiffness of the robot and surface was approximately 10 kN/m, and the translational controller parameters were chosen as $M = 0.1$, $D = 1.5$ and $K = 0$.

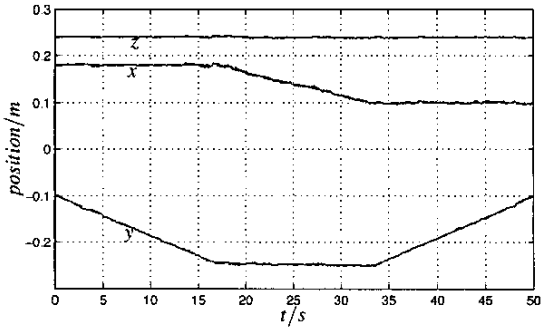

Fig. 7. Estimated tool translation (solid) and reference (dashed) during vision guided force control.

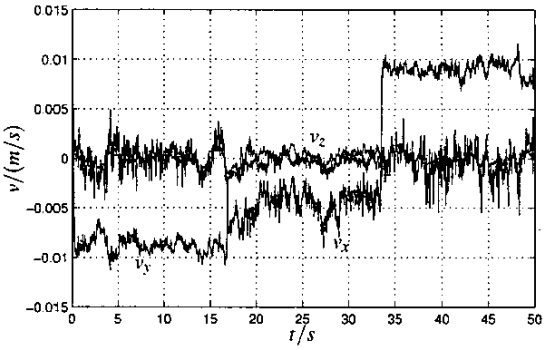

Fig. 8. Estimated tool velocity during vision guided force control.

Fig. 7 shows the estimated position of the tool with respect to the target frame during the same experiment. Fig. 8 shows the corresponding estimated velocities. A small control error in the force controlled directions is caused by the force control action, which makes the position deviate from the nominal trajectory.

Effect of calibration errors: In the presence of calibration errors $\Delta\mathbf{G}$ in Eqs. (19)–(21), the system properties will change. We assume that the error can be modeled as

$$\Delta\mathbf{G} = \mathbf{G} - \hat{\mathbf{G}} = \mathbf{G}(\mathbf{I} - \begin{bmatrix}\mathbf{R}_\Delta(\delta) & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_\Delta(\delta)\end{bmatrix}), \qquad (22)$$

where the rotation matrix $\mathbf{R}_\Delta(\delta)$ corresponds to an orientation error δ between the tracked frame and the actuated frame. In practice, the stability of the system is preserved for all reasonably small δ, but the servo properties of the system may degrade considerably. Particularly, in the common situation when the position trajectory is a ramp along the surface, large force errors may occur in the force controlled directions, due to the high surface stiffness.

We have simulated this effect in a typical scenario, where a hybrid controller with bandwidth 5 rad/s in the vision controlled direction has been used, together with an observer bandwidth of 10 rad/s. The force controller bandwidth was 15 rad/s, and the surface stiffness was 10 N/mm. Force control is applied in the x-direction, while the remaining degrees of freedom are vision controlled. The

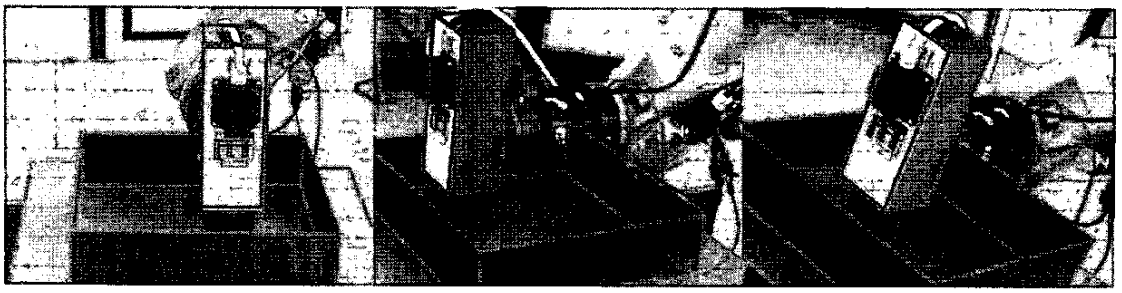

Fig. 5. Simultaneous tracking of tool and workpiece, using two Sony DFW-V300 and one Basler A602fc digital cameras.

calibration error $\mathbf{R}_\Delta(\delta)$ is given by a rotation of $\delta = 1°$ around the z-axis. $\mathbf{x}_r$ was given by a constant velocity of $\mathbf{v}_y = 10$ mm/s in the y-direction. The resulting stationary force error was 0.18 N, an error that scales approximately linearly with δ and $\mathbf{v}_y$.

IV. Discussion

The combination of force- and visual feedback is ideal for handling environments with geometric uncertainties on different scales, where the force controller is responsible for accurate control of the contact force, while the visual control handles of the overall guidance of the tool. Experiments show that the system is able to follow low speed trajectories with an accuracy of around 1 mm, while accurately controlling the contact force. The force control achieves tracking with rise times of under 0.2 s in stiff environments, so that the force control can quickly compensate for deviations from the nominal geometry. At higher speeds along the surface, calibration errors may cause large errors in the contact force, and the effects of geometrical deviations in the workpiece will increase.

By tracking multiple objects and controlling the relative position, we can theoretically achieve surface following with an accuracy that is independent of the calibration accuracy of the work cell. The price we pay is the use of external sensors such as cameras for position control. The robustness of camera sensing is still problematic, since phenomena such as occlusions, reflections, poor lighting or limited fields of view could cause signal loss, or degradation of measurement accuracy. Robustness increases considerably with the number of cameras [5], and using multiple low-cost cameras could be a cost effective solution for tasks in uncalibrated environments.

V. Conclusions

In this paper we have demonstrated how to achieve high performance six degree-of-freedom combined vision/force control for interaction with a stiff uncalibrated environment. A process with linear dynamics in task space, is used together with a standard edge-based rigid body tracker, which gives a locally stable observer with linear error dynamics. The effect of error sources such as image measurement noise and geometrical calibration errors are considered. Finally, experiments and simulations are used to validate the approach.

References

[1] V. Lippiello, B. Siciliano, and L. Villani, "Objects motion estimation via BSP tree modeling and Kalman filtering of stereo images," in *IEEE Int. Conf. on Robotics and Automation*, Washington D.C., 2002, pp. 2968–2973.

[2] T. Olsson, J. Bengtsson, A. Robertsson, and R. Johansson, "Visual position tracking using dual quaternions with hand-eye motion constraints," in *IEEE Int. Conf. on Robotics and Automation*, Taipei, Taiwan, Sept. 2003, pp. 3491–3496.

[3] P. Wunsch and G. Hirzinger, "Real-time visual tracking of 3d objects with dynamic handling of occlusion," in *IEEE Int. Conf. on Robotics and Automation*, Albuquerque, NM, USA, 1997, pp. 2868–2873.

[4] T. Drummond and R. Cipolla, "Real-time tracking of complex structures with on-line camera calibration," in *British Machine Vision Conf.*, 1999, vol. 2, pp. 574–583.

[5] F. Martin and R. Horaud, "Multiple camera tracking of rigid objects," *Int. Journal of Robotics Research*, vol. 21, no. 2, pp. 97–113, February 2002.

[6] B. Siciliano and L. Villani, *Robot Force Control*, Kluwer Academic Publishers, 1999.

[7] B.J. Nelson, J.D. Morrow, and P.K. Khosla, "Improved force control through visual servoing," in *Proc. American Control Conf.*, 1995, pp. 380–386.

[8] G. Morel, E. Malis, and S. Boudet, "Impedance based combination of visual and force control," in *Proceedings IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgium, May 1998, pp. 1743–1748.

[9] J. Baeten, H. Bruyninckx, and J. De Schutter, "Combining eye-in-hand visual servoing and force control in robotic tasks using the task frame," in *Proceedings IEEE Int. Conf. on Multisensor Fusion*, Taipei, Taiwan, August 1999, pp. 141–146.

[10] S. Jörg, J. Langwald, J. Stelter, G. Hirzinger, and C. Natale, "Flexible robot-assembly using a multi-sensory approach," in *IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, USA, 2000, pp. 3687–3694.

[11] D. Xiao, B. Ghosh, N. Xi, and T.J. Tarn, "Sensor-based hybrid position/force control of a robot manipulator in an uncalibrated environment," *IEEE Trans. on Control Systems Technology*, vol. 8, no. 4, pp. 635–645, July 2000.

[12] A. Pichler and M. Jägersand, "Uncalibrated hybrid force-vision manipulation," in *Proceedings IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2000, vol. 3, pp. 1866–1871.

[13] K. Hosoda, K. Igarashi, and M. Asada, "Adaptive hybrid visual servoing/force control in unknown environment," in *Proceedings of the 1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 1996, vol. 3, pp. 1097–1103.

[14] E. Malis and P. Rives, "Robustness of image-based visual servoing with respect to depth distribution errors," in *IEEE Int. Conf. on Robotics and Automation*, Taipei, Taiwan, 2003, pp. 1056–1061.

[15] D. Henriksson and T. Olsson, "Maximizing the use of computational resources in multi-camera feedback control," in *10th IEEE Real-Time and Embedded Technology and Applications Symposium RTAS04*, Toronto, Canada, May 2004, pp. 360–367.

[16] A. Blomdell *et al*, "Extending an industrial robot controller with a fast open sensor interface — implementation and applications.," *IEEE Robotics and Automation Magazine*, 2004, to appear.