Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Measurement Errors in Visual Servoing

Ville Kyrki[1], Danica Kragic and Henrik I Christensen
Centre for Autonomous Systems, Royal Institute of Technology, Stockholm, Sweden
Email: kyrki@lut.fi, {danik, hic}@nada.kth.se

***Abstract*— In recent years, a number of hybrid visual servoing control algorithms have been proposed and evaluated. For some time now, it has been clear that classical control approaches — image and position based — have some inherent problems. Hybrid approaches try to combine them to overcome these problems. However, most of the proposed approaches concentrate on the design of the control law, neglecting the issue of errors resulting from the sensory system.**

This paper addresses the issue of measurement errors in visual servoing. The particular contribution is the analysis of the propagation of image error through pose estimation and visual servoing control law. We have chosen to investigate the properties of the vision system and their effect to the performance of the control system. Two approaches are evaluated: i) position, and ii) 2 1/2 D visual servoing. We believe that our evaluation offers a tool to build and analyze hybrid control systems based on, for example, switching [1] or partitioning [2].

I. Introduction

The performance issues of visual servoing methods in the presence of errors have received a considerable amount of attention. Particularly, the effect of camera calibration errors has been studied, e.g., [3]. Also, the convergence properties of the systems are known for most cases. While the convergence of the system is an essential property, it does not reveal knowledge about the possible trajectory and its uncertainty.

The procedures of camera calibration have improved enormously over the last decade. However, even perfect calibration does not overcome the restriction of the image resolution. The discretization error causes an uncertainty in the control. This paper proposes the use of error propagation in the analysis and comparison of different types of visual servoing methods, i.e., position-based and hybrid.

In Figure 1, a general model of visual servoing is presented. It divides the system into three parts: pose estimation, servoing strategy, and control strategy. This model can be used with most position-based and hybrid approaches. It is based on eye-in-hand configuration, and the objective of servoing is defined as bringing the camera to a desired pose with respect to the target. The pose estimation part may compute the full 3-D pose of the target, or it may use homography- or epipolar-based techniques to infer the pose. The choice of servoing strategy determines the error function of the system and thus has most effect on the trajectory while the control strategy affects convergence properties.

To compare the servoing methods based on the effect of errors in the image, we need a common reference. We use the control output of a Cartesian controller as the reference,

[1]V. Kyrki was supported by a grant from the Academy of Finland.

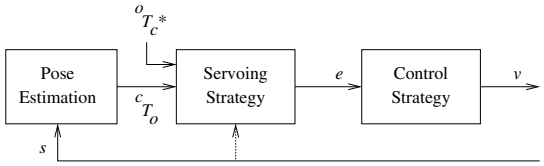

Fig. 1. System model

since it seems reasonable to study the sensitivity of the system by propagating the errors in the image measurements to the control output. The division of the servoing model into subsystems allows us not only to compare the behavior of complete systems but also to compare the components of the system.

We will use the error analysis to compare position based visual servoing and the hybrid approach termed 2.5D visual servoing proposed by Malis et al. [2]. The rest of the paper is organized as follows: We begin by motivating the research by surveying related work in Section II. In Section III we present the pose estimation algorithm and analyze its error propagation. Sections IV and V present the position-based and hybrid servoing and their analyses. The analytical results are verified by experiments in Section VI, which also discusses the merits of the different approaches. Finally, in Section VII, we present a summary and conclusions.

II. Related work

The work in this paper is closely related to the analysis of pose estimation algorithms. Pose estimation using a 2-D projection and a 3-D model is a widely studied problem in computer vision, and several approaches exist to solve the problem. Many of the approaches are, however, iterative, which is a disadvantage in the context of visual servoing, where the "real-time" is a requirement. There are a few closed form solutions for point feature based pose estimation [4]–[6]. We have used the algorithm by Fiore [5], but any of these algorithms could be used. While in the structure-from-motion there are analyses of sensitivity based on linear error propagation, according to the authors knowledge no corresponding analyses have been published for the pose estimation. Haralick et al. have demonstrated empirically that pose estimation breaks down when the noise exceeds a certain threshold [7]. Ansar et al. have presented sensitivity analysis but their results are upper bounds for the error derived from matrix perturbation theory [6]. Their experiments also reveal

that the bounds are highly conservative and are not thus well suited for comparing different systems.

The error characteristics of visual servoing are usually investigated from either of the two different points of view: the stability of the closed-loop system, or the steady-state error [8]. It is known that the convergence of position-based visual servoing (PBVS) is sometimes inhibited by the loss of stability in pose estimation [9]. 2.5D servoing does not seem to suffer from this problem [10], unless the partial pose estimation becomes unstable. Deng [8] has proposed use of the steady-state error as a measure of sensitivity of visual servoing. However, if long trajectories are allowed, it is important to know about the sensitivity of the system along the trajectory to, for example, predict the set of possible trajectories in the presence of errors. Another approach is to consider the outliers in the image data. Comport et al. [11] have proposed a scheme to increase the robustness by embedding the outlier processing into the control law. Outlier rejection can also be performed in the image processing step [12].

Recently, Gans et al. [1] have proposed switching between position- and image-based servoing. A possible application for error modeling would be to use it to make the switching decisions, which is currently an unsolved problem.

III. Pose estimation

In this section, the pose estimation algorithm of Fiore [5] is first briefly described with some modifications to facilitate the analysis. This is followed by the analysis of error propagation.

A. Estimation algorithm

The exterior orientation problem seeks the similarity transform consisting of translation $\mathbf{t}$ and rotation $\mathbf{R}$ that brings a set of known 3D feature points $\mathbf{a}_i$ into alignment with a set of corresponding image plane points (x_i, y_i). Without loss of generality, we can assume unit focal length of the camera. Then, translation and rotation are the one that best satisfy the set of equations

$$l_i \begin{bmatrix} x_i & y_i & 1 \end{bmatrix}^T = s\mathbf{R}(\mathbf{a}_i + \mathbf{t}), \;\; i = 1, \ldots, N \qquad (1)$$

where l_i are the projective parameters, s is a scale factor, and N is the number of feature points.

Now, the parameters l_i are first solved. Let us define the data matrix $\mathbf{P}$ for 3D points as

$$\mathbf{P} = \begin{pmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_N \\ 1 & \cdots & 1 \end{pmatrix}. \qquad (2)$$

Then, we can find the $N \times N - 4$ weight matrix $\mathbf{W}$ which satisfies

$$\mathbf{PW} = \mathbf{0} \qquad (3)$$

from the singular value decomposition (SVD) of $\mathbf{P}$ as the matrix of the $N-4$ right singular vectors of $\mathbf{P}$ corresponding to its null space. Defining the vector of projective parameters $\mathbf{l} = [l_1, \ldots, l_N]$, it must satisfy $\mathbf{l} = \mathbf{P}^T\boldsymbol{\alpha}$ for a 4×1 unknown vector $\boldsymbol{\alpha}$. Then, $\boldsymbol{\alpha}$ can be found as the solution to the set of homogeneous linear equations

$$\mathbf{C}\boldsymbol{\alpha} \equiv \begin{bmatrix} w_{1,1}\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} & \cdots & w_{N,1}\begin{pmatrix} x_N \\ y_N \end{pmatrix} \\ \vdots & & \vdots \\ w_{1,N-4}\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} & \cdots & w_{N,N-4}\begin{pmatrix} x_N \\ y_N \end{pmatrix} \end{bmatrix} \mathbf{l} = \mathbf{0}. \qquad (4)$$

The solution is found as the eigenvector corresponding to the smallest eigenvalue of $\mathbf{C}^T\mathbf{C}$. This in turn gives the set of projective parameters.

Now, we only need to recover the absolute orientation with scaling. With l_i known, we can write Eq. 1 as

$$\mathbf{b}_i = s\mathbf{R}(\mathbf{a}_i + \mathbf{t}), \;\; i = 1, \ldots, N \qquad (5)$$

where $\mathbf{b}_i = [l_i x_i, l_i y_i, l_i]^T$. The unknown scale parameter s can be solved by centering the two point sets $\mathbf{a}_i$ and $\mathbf{b}_i$, and calculating the ratio between the lengths of the centered vectors.

Next, we want to find the rotation matrix that minimizes the sum of the square errors between the centered point sets, that is, $\sum_{i=1}^{N} \|\tilde{\mathbf{b}}_i - s\mathbf{R}\tilde{\mathbf{a}}_i\|_2^2$, where $\tilde{\mathbf{a}}_i$ and $\tilde{\mathbf{b}}_i$ are the centered points. This, so called Orthogonal Procrustes problem, can be solved using SVD as suggested by Fiore, but we choose to solve the rotation using unit quaternions as presented by Weng et al. in [13]. The solution involves another eigenvalue decomposition for the solution of a set of homogeneous linear equations. Finally, the translation is found from $\mathbf{t} = s^{-1}\mathbf{R}^T\mathbf{b}_0 - \mathbf{a}_0$ where $\mathbf{a}_0$ and $\mathbf{b}_0$ are the point set centroids.

B. Error analysis

The error analysis in this paper is based on first-order error propagation [14]. The goal of this analysis is to find the covariance of the pose estimate with respect to the variances of image plane coordinates. While errors can be also analyzed by finding worst case error bounds, these can result in overly conservative bounds that are suitable only for small errors. In practice, the possible redundancy of data in pose estimation (i.e. having more features than necessary) allows finding stable solutions also in the presence of noise. In this paper, it is assumed that the errors in the pose estimate result from the errors in the image coordinates of features. Their sources include spatial quantization, feature detection, and camera distortion. However, we assume that there is no systematic calibration error and thus the errors in the image can be modeled as zero-mean random variables. It is further assumed that the errors between points are uncorrelated.

Let $\mathbf{x}$ be the vector of image coordinates of features such that $\mathbf{x} = (x_1, \ldots, x_N, y_1, \ldots, y_N)^T$. We can formulate the error analysis problem as finding the matrices $\mathbf{D_t}$ and $\mathbf{D_R}$ such that $\boldsymbol{\delta_t} = \mathbf{D_t}\boldsymbol{\delta_x}$ and $\boldsymbol{\delta_R} = \mathbf{D_R}\boldsymbol{\delta_x}$ are linear error estimates in $\mathbf{t}$ and $\mathbf{R}$ with respect to errors in $\mathbf{x}$. It is evident that $\mathbf{D_t}$ and $\mathbf{D_R}$ depend on the values of both $\mathbf{x}$ and $\mathbf{a}_i$s. Note that matrix $\mathbf{R}$ must be represented as a vector $\mathbf{r}$ by concatenating the columns of the matrix into a single vector. Thus, $\boldsymbol{\delta_R}$ is the error in this vector. For vectors, let $\boldsymbol{\Gamma}$ denote the covariance matrix, e.g., $\boldsymbol{\Gamma}_\mathbf{x} = E[\boldsymbol{\delta_x}\boldsymbol{\delta_x}^T]$.

The error is now propagated through the pose estimation algorithm. First, it can be seen that $\mathbf{W}$ in Eq. 3 depends only on matrix $\mathbf{P}$ where there is no associated uncertainty. Now, the uncertainty in matrix $\mathbf{C}$ (Eq. 4) can be found by finding the matrix $\mathbf{G_C}$ that represents the transform from $\mathbf{x}$ to $\mathbf{c}$ (and $\boldsymbol{\delta}_\mathbf{c} = \mathbf{G_C}\boldsymbol{\delta}_\mathbf{x}$), the vector representation of matrix $\mathbf{C}$. This operation is linear so no approximations are needed. The matrix is easily found to be

$$\mathbf{G_C} = \begin{pmatrix} \mathbf{Q_1} & \mathbf{0} & \mathbf{Q_2} & \mathbf{0} & \mathbf{Q_3} & \mathbf{0} & \mathbf{Q_4} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_1} & \mathbf{0} & \mathbf{Q_2} & \mathbf{0} & \mathbf{Q_3} & \mathbf{0} & \mathbf{Q_4} \end{pmatrix}^T \tag{6}$$

where

$$\mathbf{Q_i} = \begin{pmatrix} W_{1,1}P_{i,1} & \cdots & W_{1,N-4}P_{i,1} \\ \vdots & & \vdots \\ W_{N,1}P_{i,N} & \cdots & W_{N,N-4}P_{i,N} \end{pmatrix}.$$

Next, the linear estimate for the error in $\mathbf{C}^T\mathbf{C}$ is found. Denoting the error matrix corresponding to vector $\boldsymbol{\delta}_\mathbf{C}$ by $\boldsymbol{\Delta}_\mathbf{C}$, the linear estimate is

$$\boldsymbol{\Delta}_{\mathbf{C}^T\mathbf{C}} \approx \mathbf{C}^T\boldsymbol{\Delta}_\mathbf{C} + \boldsymbol{\Delta}_\mathbf{C}{}^T\mathbf{C}. \tag{7}$$

Using the vector notation, this can be written as

$$\boldsymbol{\delta}_{\mathbf{C}^T\mathbf{C}} \approx \mathbf{G}_{\mathbf{C}^T\mathbf{C}}\boldsymbol{\delta}_\mathbf{C} = \mathbf{G}_{\mathbf{C}^T\mathbf{C}}\mathbf{G}_\mathbf{C}\boldsymbol{\delta}_\mathbf{x} = \mathbf{D}_{\mathbf{C}^T\mathbf{C}}\boldsymbol{\delta}_\mathbf{x} \tag{8}$$

where $\mathbf{G}_{\mathbf{C}^T\mathbf{C}}$ can be determined using Eq. 7.

To propagate the error through the eigenvalue decomposition, we use the result presented by Weng et al. in [13]. The linear error term in $\boldsymbol{\alpha}$, the smallest eigenvector of $\mathbf{C}^T\boldsymbol{C}$, is given by

$$\begin{aligned} \boldsymbol{\delta}_{\boldsymbol{\alpha}} &\approx \mathbf{H}\boldsymbol{\Delta}\mathbf{H}^T\boldsymbol{\Delta}_{\mathbf{C}^T\mathbf{C}}\boldsymbol{\alpha} \\ &= \mathbf{H}\boldsymbol{\Delta}\mathbf{H}^T[\alpha_1 I_4 \quad \alpha_2 I_4 \quad \alpha_3 I_4 \quad \alpha_4 I_4]\boldsymbol{\delta}_{\mathbf{C}^T\mathbf{C}} \\ &= \mathbf{G}_{\boldsymbol{\alpha}}\boldsymbol{\delta}_{\mathbf{C}^T\mathbf{C}} = \mathbf{G}_{\boldsymbol{\alpha}}\mathbf{D}_{\mathbf{C}^T\mathbf{C}}\boldsymbol{\delta}_\mathbf{x} = \mathbf{D}_{\boldsymbol{\alpha}}\boldsymbol{\delta}_\mathbf{x} \end{aligned} \tag{9}$$

where $\mathbf{H}$ is the matrix of eigenvectors of $\mathbf{C^TC}$ and $\boldsymbol{\Delta}$ is given in terms of the eigenvalues λ_i as

$$\boldsymbol{\Delta} = \text{diag}\left\{0, (\lambda_1 - \lambda_2)^{-1}, (\lambda_1 - \lambda_3)^{-1}, (\lambda_1 - \lambda_4)^{-1}\right\}.$$

As the projective parameters depend linearly on $\boldsymbol{\alpha}$, we can find the associated error as $\boldsymbol{\delta}_\mathbf{l} = \mathbf{P}^T\boldsymbol{\delta}_{\boldsymbol{\alpha}} \approx \mathbf{P}^T\mathbf{D}_{\boldsymbol{\alpha}}\boldsymbol{\delta}_\mathbf{x} = \mathbf{D}_\mathbf{l}\boldsymbol{\delta}_\mathbf{x}$.

Now we continue to propagate the errors to $\mathbf{b}_i$. Let $\boldsymbol{\delta}_\mathbf{B} = [\delta_{l_1x_1}, \cdots, \delta_{l_Nx_N}, \delta_{l_1y_1}, \cdots, \delta_{l_Ny_N}, \delta_{l_1}, \cdots, \delta_{l_N}]$. The linear approximation for the error is

$$\begin{aligned} \boldsymbol{\delta}_\mathbf{B} &\approx \begin{pmatrix} \text{diag}(\mathbf{1}) & \mathbf{0} & \text{diag}(\mathbf{x_{1\ldots N}}) \\ \mathbf{0} & \text{diag}(\mathbf{1}) & \text{diag}(\mathbf{y_{1\ldots N}}) \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{\delta}_\mathbf{x} \\ \boldsymbol{\delta}_\mathbf{l} \end{pmatrix} \\ &= \begin{pmatrix} \text{diag}(\mathbf{1})+\text{diag}(\mathbf{x_{1\ldots N}})\mathbf{D_l} & \mathbf{0} \\ \mathbf{0} & \text{diag}(\mathbf{1})+\text{diag}(\mathbf{y_{1\ldots N}})\mathbf{D_l} \\ \mathbf{D_l} & \end{pmatrix} \boldsymbol{\delta}_\mathbf{x} \\ &= \mathbf{D_B}\boldsymbol{\delta}_\mathbf{x} \end{aligned} \tag{10}$$

In the following, we'll skip the details on linear steps of the error propagation to keep the discussion as brief as possible while still stating each approximation during the nonlinear steps. Centering the set of vectors $\mathbf{b}_i$ does not involve nonlinear operations so no approximations need to be done to find the error in $\tilde{\mathbf{b}_i}$. Then, $\boldsymbol{\delta}_{\tilde{\mathbf{B}}} \approx \mathbf{D}_{\tilde{\mathbf{B}}}\boldsymbol{\delta}_\mathbf{x}$. In calculating the scale, $\delta_s \approx \mathbf{G_s}\boldsymbol{\delta}_{\tilde{\mathbf{B}}}$ where

$$\mathbf{G_s} = \frac{1}{\sum_i \|\tilde{\mathbf{a}}_i\|^2}\left(\frac{\tilde{b}_{1,1}\|\mathbf{a}_1\|}{\|\mathbf{b}_1\|} \cdots \frac{\tilde{b}_{N,1}\|\mathbf{a}_N\|}{\|\mathbf{b}_N\|}\right) \tag{11}$$

As stated before, the rotation matrix is now estimated using unit quaternions. This encompasses another case of determining the eigenvector corresponding to the smallest eigenvalue of a matrix $\mathbf{E}$, which is a linear combination of previously known variables and thus its error $\boldsymbol{\delta}_\mathbf{E}$ can be represented as a matrix product $\boldsymbol{\delta}_\mathbf{E} = \mathbf{G_E}[\boldsymbol{\delta}_{\tilde{\mathbf{B}}}^T, \delta_s]^T$. The error can now be propagated in a similar fashion as shown above for vector $\boldsymbol{\alpha}$. As a result, we get the unit quaternion $\mathbf{q}$ that represents the rotation and its error with respect to the errors in input $\boldsymbol{\delta}_\mathbf{q} \approx \mathbf{D_q}\boldsymbol{\delta}_\mathbf{x}$. We can estimate the first order perturbation of $\mathbf{R}$ as $\boldsymbol{\delta}_\mathbf{R} \approx \mathbf{G_R}\boldsymbol{\delta}_\mathbf{q} = \mathbf{D_R}\boldsymbol{\delta}_\mathbf{x}$. The error in the translation can be finally estimated from the first-order Taylor expansion as

$$\begin{aligned} \boldsymbol{\delta}_\mathbf{t} &\approx \frac{1}{s}(\mathbf{R}^T\boldsymbol{\delta}_{\mathbf{b}_0} + \boldsymbol{\Delta}_\mathbf{R}{}^T\mathbf{b}_0) - \frac{1}{s^2}\delta_s\mathbf{R}^T\mathbf{b}_0 \\ &= \mathbf{G_t}\boldsymbol{\delta}_\mathbf{E} = \mathbf{D_t}\boldsymbol{\delta}_\mathbf{x} \end{aligned} \tag{12}$$

In summary, we have expressed the perturbations in the pose estimate as a linear transformation of the perturbations in the input image. This allows us also to write the covariance matrices of the pose parameters as

$$\boldsymbol{\Gamma_R} = \mathbf{D_R}\boldsymbol{\Gamma_x}\mathbf{D_R}{}^T \qquad \boldsymbol{\Gamma_t} = \mathbf{D_t}\boldsymbol{\Gamma_x}\mathbf{D_t}{}^T. \tag{13}$$

The following two sections outline two visual servoing methods and relate the uncertainty in the pose estimate presented in this section to the uncertainty in the control.

IV. Position Based Visual Servoing

In position-based visual servoing (PBVS), the task function is defined in terms of the pose-vector from the current to the desired position, which can be expressed as the transformation ${}^cT_{c^*}$. The input image is usually used to estimate the camera to object transformation ${}^cT_o$ which can be composed with the object to desired pose transformation ${}^oT_{c^*}$ to find the relation from the current to the desired pose. By decomposing the transformation matrices into translation and rotation, this can be expressed as

$$\begin{aligned} {}^cT_{c^*} &= {}^cT_o\,{}^oT_{c^*} = \begin{pmatrix} {}^c\mathbf{R}_o & {}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{pmatrix}\begin{pmatrix} {}^o\mathbf{R}_{c^*} & {}^o\mathbf{t}_{c^*} \\ \mathbf{0} & 1 \end{pmatrix} \\ &= \begin{pmatrix} {}^c\mathbf{R}_o\,{}^o\mathbf{R}_{c^*} & {}^c\mathbf{R}_o\,{}^o\mathbf{t}_{c^*}+{}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{pmatrix} = \begin{pmatrix} {}^c\mathbf{R}_{c^*} & {}^c\mathbf{t}_{c^*} \\ \mathbf{0} & 1 \end{pmatrix} \end{aligned} \tag{14}$$

The task function for position is then the vector ${}^c\mathbf{t}_{c^*}$, and for the orientation the rotation matrix can be decomposed into axis of rotation $\mathbf{u}$ and angle θ, which can be multiplied to attain the task function $\mathbf{u}\theta$.

Starting from the result of the analysis of pose estimation, we first inspect the camera to object transformation. The rotation matrix $\mathbf{R}$ in the image formation model in Eq. 1 is the desired rotation from the camera to object frames ${}^c\mathbf{R}_o$. The translation, however, is given in the object instead of the camera frame. Thus, we need to rotate the translation vector to correspond to camera frame axes, and find the uncertainty for this rotated vector using the uncertainties in both the rotation

matrix and the translation vector. The uncertainty can thus be expressed as $\boldsymbol{\delta}_{c\mathbf{t}_o} \approx \mathbf{G}_{c\mathbf{t}_o}[\mathbf{D}_\mathbf{R}{}^T, \mathbf{D}_\mathbf{t}{}^T]^T\boldsymbol{\delta}_\mathbf{x} = \mathbf{D}_{c\mathbf{t}_o}\boldsymbol{\delta}_\mathbf{x}$.

Assuming that there is no uncertainty associated with the desired position, the error in the rotation from the current to desired pose can be approximated as $\boldsymbol{\Delta}_{c\mathbf{R}_{c^*}} \approx \boldsymbol{\Delta}_{c\mathbf{R}_o}{}^o\mathbf{R}_{c^*}$ which can be expressed as $\boldsymbol{\delta}_{c\mathbf{R}_{c^*}} \approx \mathbf{G}_{c\mathbf{R}_{c^*}}\boldsymbol{\delta}_{c\mathbf{R}_o}$. For the translation, the corresponding errors can be written $\boldsymbol{\delta}_{c\mathbf{t}_{c^*}} \approx \boldsymbol{\Delta}_{c\mathbf{R}_o}{}^o\mathbf{t}_{c^*} + \boldsymbol{\delta}_{c\mathbf{t}_o} = \mathbf{G}_{c\mathbf{t}_{c^*}}[\boldsymbol{\delta}_{c\mathbf{R}_o}^T, \boldsymbol{\delta}_{c\mathbf{t}_o}^T]^T = \mathbf{D}_{c\mathbf{t}_{c^*}}\boldsymbol{\delta}_\mathbf{x}$. Now, what remains is to transform the rotation matrix into control vector for rotation. We use the $\mathbf{u}\theta$ form and estimate the errors as $\boldsymbol{\delta}_{\mathbf{u}\theta} \approx \mathbf{G}_{\mathbf{u}\theta}\boldsymbol{\delta}_{c\mathbf{R}_{c^*}} = \mathbf{D}_{\mathbf{u}\theta}\boldsymbol{\delta}_\mathbf{x}$. Assuming that a proportional control is used, the error in the control vector $\mathbf{v}$ is finally estimated as

$$\boldsymbol{\delta}_\mathbf{v} = -\lambda\begin{pmatrix}\boldsymbol{\delta}_{c\mathbf{t}_{c^*}}\\ \boldsymbol{\delta}_{\mathbf{u}\theta}\end{pmatrix} \approx \begin{pmatrix}-\lambda\mathbf{D}_{c\mathbf{t}_{c^*}}\\ -\lambda\mathbf{D}_{\mathbf{u}\theta}\end{pmatrix}\delta_x = \mathbf{D}_\mathbf{v}\boldsymbol{\delta}_\mathbf{x}. \tag{15}$$

This allows us also to approximate the covariance matrix of the control error from $\boldsymbol{\Gamma}_\mathbf{v} = E[\boldsymbol{\delta}_\mathbf{v}\boldsymbol{\delta}_\mathbf{v}^T] \approx \mathbf{D}_\mathbf{v}\boldsymbol{\delta}_\mathbf{x}\mathbf{D}_\mathbf{v}{}^T$.

V. Hybrid Visual Servoing

The hybrid visual servoing approach, called 2.5D servoing, was originally presented as a means to avoid the target leaving the field of view of the camera (a PBVS problem), and to perform servoing without a 3D model of the target [2]. It is based on partial pose estimation using a scaled Euclidean reconstruction with a homography decomposition. However, it can be also used with full pose estimation.

We now briefly present the 2.5D servoing with full pose estimation used in our work. The control scheme is based on controlling the orientation using the estimated 3-D rotation between the current and desired poses and driving the vector $\mathbf{u}\theta$ to zero just as in position-based visual servoing. The position in turn is controlled using a single point feature that is driven towards its desired location in both image coordinates and depth. Thus, the visibility of the feature during the servoing sequence is guaranteed. The task vector can be defined as

$$\mathbf{e} = [x - x^*, y - y^*, \log(Z/Z^*), \theta\mathbf{u}^T]^T \tag{16}$$

where x and y are the position of the control point in image, Z is its depth, and asterisks denote the desired values. The motion control law is then

$$\mathbf{v} = -\lambda\begin{pmatrix}\mathbf{L}_v^{-1} & -\mathbf{L}_v^{-1}\mathbf{L}_{v\omega}\\ \mathbf{0} & \mathbf{I}\end{pmatrix}\mathbf{e} \tag{17}$$

where

$$\mathbf{L}_v^{-1} = \begin{pmatrix}-Z & 0 & -xZ\\ 0 & -Z & -yZ\\ 0 & 0 & -Z\end{pmatrix} \tag{18}$$

and

$$\mathbf{L}_{v\omega} = \begin{pmatrix}xy & -(1+x^2) & y\\ 1+y^2 & -xy & -x\\ -y & x & 0\end{pmatrix}. \tag{19}$$

In our framework (Fig. 1), the rotation $\mathbf{u}\theta$ and the depth Z are calculated using the pose estimation while x and y result directly from image measurements. Z can thus be written as

$$[X, Y, Z]^T = {}^cT_o[\mathbf{a}^T, 1]^T. \tag{20}$$

The sensitivity for the rotation is identical to that presented in the previous section. However, we desire to estimate the error in the whole control vector to recognize correlations between the errors in different variables. The error in the depth can be approximated in terms of the errors on estimated rotation and translation as

$$\delta_Z = \mathbf{G}_Z[\mathbf{D}_\mathbf{R}{}^T, \mathbf{D}_\mathbf{t}{}^T]^T\boldsymbol{\delta}_\mathbf{x} \tag{21}$$

where $\mathbf{G}_Z$ can be determined from (20). Now, the uncertainty in the control output $\mathbf{v}$ can be approximated as $\boldsymbol{\delta}_\mathbf{v} = \mathbf{G}_\mathbf{v}[\delta_x, \delta_y, \delta_Z, \boldsymbol{\delta}_\mathbf{u}^T, \delta_\theta]^T = \mathbf{D}_\mathbf{v}\boldsymbol{\delta}_\mathbf{x}$ where $\mathbf{G}_\mathbf{v}$ can be determined from (16)–(19). Then, the covariance of the control is approximately $\boldsymbol{\Gamma}_\mathbf{v} \approx \mathbf{D}_\mathbf{v}\boldsymbol{\delta}_\mathbf{x}\mathbf{D}_\mathbf{v}{}^T$.

VI. Experimental evaluation

In this section, we present the experiments performed to validate the presented error analysis and to compare position based and hybrid visual servoing. We begin by considering the pose estimation algorithm, then consider the visual servoing approaches separately, and conclude by discussing the relative properties of the approaches.

A. Pose estimation

Figure 2 shows the validity region of the error estimation. The deviation of the translation with respect to image error is presented on the left in Fig. 2, while the deviation in the rotation angle is on the right. The breakdown point of the error estimation is when the deviation in the image coordinates is approximately $\sigma \approx 10^{-2.5}$. Naturally, the point depends on the feature point configuration. The 6-feature target and its point deviations used in the experiment is shown on the left in Fig. 3. Four of the feature points lie on a plane while two are displaced by a small amount. It should be noted that the breakdown point of the error estimation coincides with the breakdown point of the pose estimation, that is, the error estimation becomes invalid when the pose estimation algorithm starts to break down. An obvious restriction of the linear error estimation is its inability to predict the breakdown point as it is primarily a higher order phenomenon.

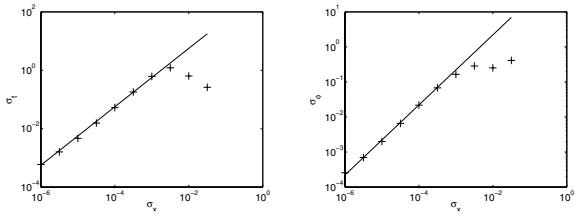

Fig. 2. Measured and predicted deviations in pose estimates with respect to image error: (left) translation; (right) rotation angle.

Another experiment was performed, which verified some known properties of pose estimation accuracy. Particularly, assuming constant deviation in the image coordinates, the error in the coordinate-axes parallel to the image plane is linearly dependent on the distance of the object from the image while the error in the axis parallel to the optical axis

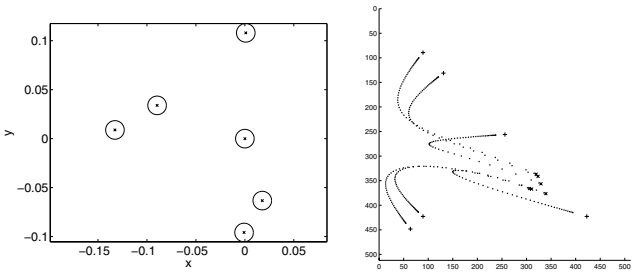

Fig. 3. (left) Deviation of image points for $\sigma = 10^{-2.5}$; (right) Image plane trajectories for PBVS.

depends on the square of the distance (a known phenomenon in stereo vision). The result also predicted full correlation between the translation in x and rotation around y, as well as vice versa. This translation-rotation-ambiguity is another well known phenomenon.

B. Position based servoing

The following experiments assume a servoing task where the camera is initially rotated around all axes and positioned relatively far away from the goal position (around ten times the desired distance). This allows us to see the effect of the distance to the servoing and also investigate the rotation around different axes. The target is the same as presented in the previous section. The trajectories of the features in the image plane are shown on right in Fig. 3.

The validity of the analysis was verified by displacing the feature locations using a known error distribution and measuring the deviation of the control output. In Fig. 4, the predicted and measured deviations in the translational velocities in y (dotted line) and z (dashed line) are shown, as well as in the rotational velocities around the same axes. The figure shows that the measured and predicted deviations agree very well which seems to indicate that the theoretic analysis is valid.

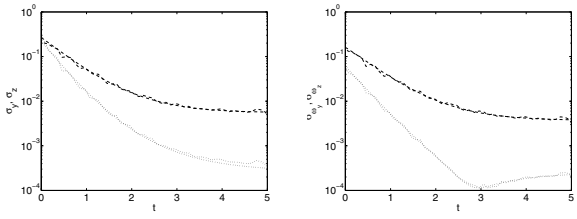

Fig. 4. Measured and predicted deviations in PBVS control output: (left) translation; (right) rotation.

Figures 5 and 6 demonstrate the error behavior of position-based servoing. The results are presented in both world and camera frames because the world frame is the most natural way to inspect the error in terms of the Cartesian controller while the camera frame reveals information about the directional nature of the error. In the figures, the solid line corresponds to x-axis translation and rotation, the dotted line to y-axis, and the dashed line to z-axis.

Top row of Fig. 5 shows the negative exponential velocity of the Cartesian control in PBVS. The absolute deviations in the control are presented in the middle of Fig. 5, and relative (deviation divided by the control output) in the bottom. Fig. 6 presents the behavior in camera frame. It can be seen that the control in the direction of the camera optical axis is much more reliable in terms of image errors, as is also the rotation around the optical axis (coinciding with the world y-axis in the goal position). There seems to be little difference in the control in the axes perpendicular to the optical axis near the goal position, but initially when the object is not yet aligned to the image plane, there is some difference in the accuracy. The maximum shown in bottom left subfigure of Fig. 6 is caused by the zero-crossing of the corresponding control (note that this zero-crossing occurs only in the camera frame, not in the world frame where PBVS guarantees a trajectory along a straight line). Another observation to make is that the relative error has a minimum along the trajectory, when the distance to the target is already quite small, but the target is still not too precisely aligned. After this minimum, the relative error continues to increase to a level that would make the servoing impossible if the error would exist in practice.

The relative errors can be used to assess the validity of the servoing in the direction of a certain axis so that when the relative error becomes dominant (say, more than third of the control), the control in that axis can begin to diverge. The target is almost perfectly aligned in z-coordinate of the world frame, so the relative error in z-translation is very high throughout the motion which seems to suggest that there is no reason to control that axis.

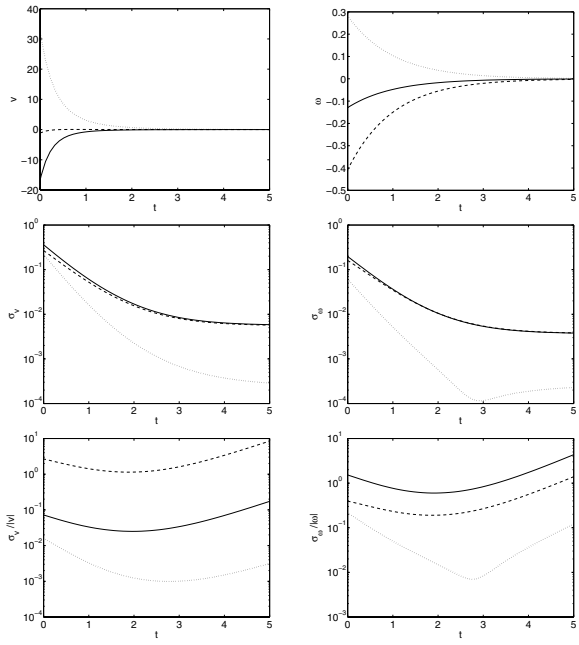

Fig. 5. PBVS behavior in world frame: (top row) Velocity; (middle row) Absolute errors; (bottom row) Relative errors; (left column) translation; (right column) rotation.

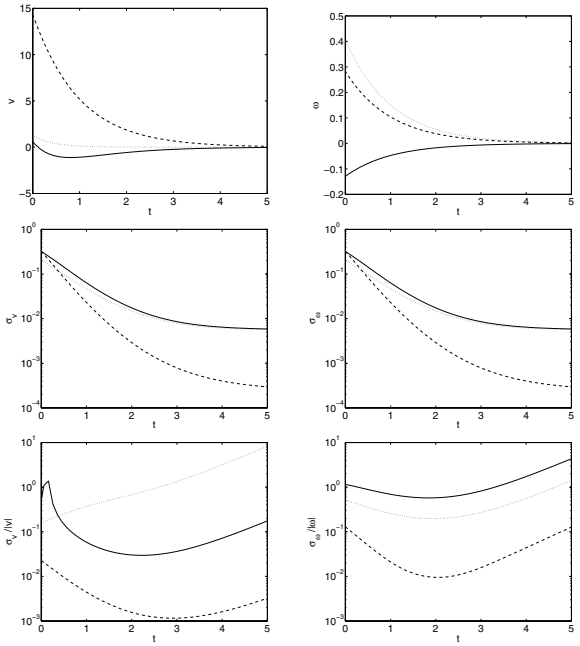

Fig. 6. PBVS behavior in camera frame: (top row) Velocity; (middle row) Absolute errors; (bottom row) Relative errors; (left column) translation; (right column) rotation.

C. Hybrid visual servoing

The same control task which was used with PBVS was also used with the hybrid approach. The results of the analysis were verified by an experiment, which can be seen in Fig. 7. The predicted deviations seem to follow the measurements well, which suggests that the analysis is valid.

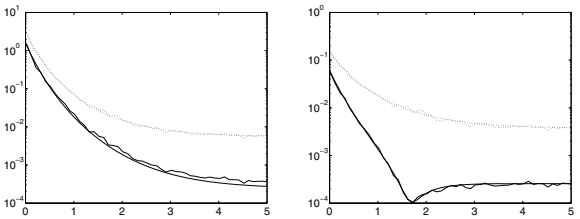

Fig. 7. Measured and predicted deviations in HYBVS control output: (left) translation; (right) rotation.

The error behavior of HYBVS can be seen in Fig. 8 for the world frame and in Fig. 9 for the camera frame. In some respects, the behavior is similar to PBVS. Most importantly, the errors in the translation along the optical axis and in the rotation around it are considerably smaller than for the axes parallel to the image plane. The behavior in rotation resembles that of PBVS, but it is important to note that they are not identical, as the systems have a different trajectory. It is easy to notice that HYBVS has a faster control in the depth (Figs. 5 and 8), and this seems to be the reason for it to attain the constant error region of z-axis rotation sooner (Figs. 5 and 8). The relative errors show another characteristic of HYBVS, the occurrence of zero-crossings in the Cartesian control. This can be seen easily from the strong peak of the relative error in translation (Fig. 5). The relative errors also suggest the regions where the control is likely to diverge due to the errors in pose estimation. For HYBVS it seems that translation can be controlled at least to some degree in x and y and rotation in y and z. Now, the translation-rotation ambiguity can be again seen as the z-axis translation corresponds to x-axis rotation in the world frame.

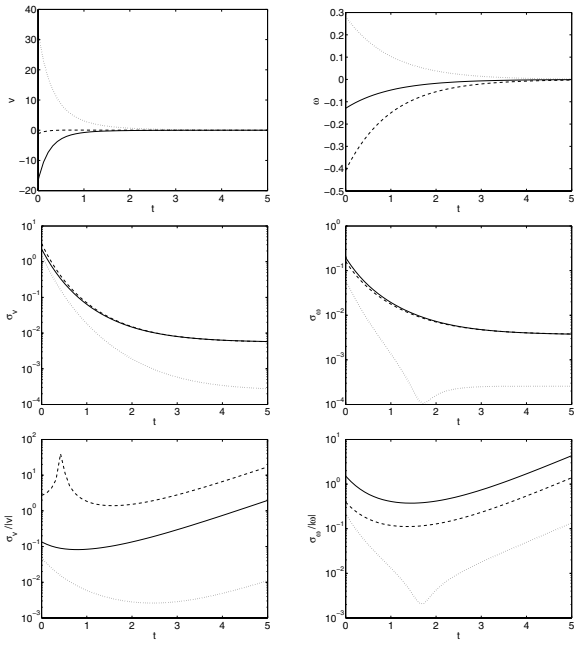

Fig. 8. HYBVS behavior in world frame: (top row) Velocity; (middle row) Absolute errors; (bottom row) Relative errors; (left column) translation; (right column) rotation.

D. Discussion

A common reference trajectory needs to be defined in order to compare PBVS and HYBVS uncertainties with respect to time. Fig. 10 presents the estimated errors of HYBVS control when the camera is moved along the trajectory generated using PBVS. Thus, the camera location with respect to time corresponds to Figs. 5 and 6. The errors are presented in the world frame. The absolute and relative errors for translation in Fig. 10 correspond to Fig. 5. For the rotation part, the errors are not shown, as they would be identical to the PBVS case since the rotation control is identical. PBVS has clearly smaller absolute errors than HYBVS in the beginning. A possible explanation for this is that the methods follow a different trajectory. Another issue is the ability of PBVS to use all feature points for the pose estimation, while HYBVS uses only a single control point to control the trajectory parallel to the image plane. The relative errors have some similarities, particularly the order of the axes is the same. For the translation along y (which is closest to the optical axis, and which has the longest initial distance), the relative error

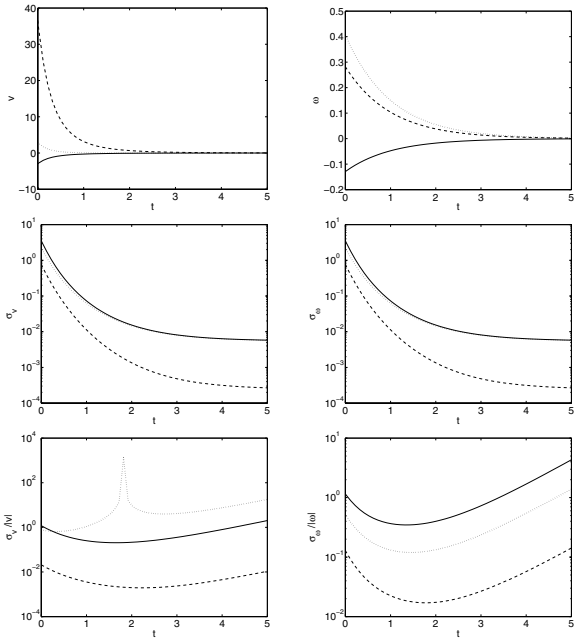

Fig. 9. HYBVS behavior in camera frame: (top row) Velocity; (middle row) Absolute errors; (bottom row) Relative errors; (left column) translation; (right column) rotation.

amplitudes seem to be comparable. For the x-axis, which has some initial error, PBVS is initially less prone to errors, while later in the trajectory the errors become comparable. For the z-axis, HYBVS has slightly smaller error, but it is unlikely that either can be used for efficient control, as the error is large. In addition, the reason that HYBVS has smaller relative error is that it initially controls the axis away from the point of convergence as was seen in the existence of the zero-crossing discussed earlier.

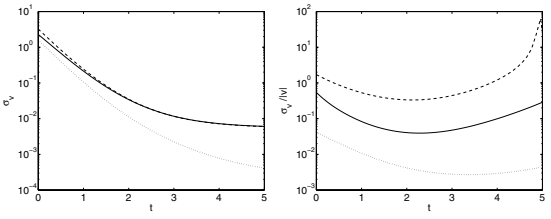

Fig. 10. HYBVS translation errors on PBVS trajectory: (left) absolute; (right) relative.

VII. Summary and Conclusion

In this paper, we have analyzed the effect of measurement errors in visual servoing. The particular contribution of this paper is the propagation of image error through pose estimation and visual servoing control law. In particular, we have investigated the properties of the vision system and their effect to the performance of the control system. Two approaches have been evaluated: i) position, and ii) 2 1/2D visual servoing. We believe that our evaluation offers a valid tool to design hybrid control systems based on, for example, switching [1] or partitioning [2].

Our future work will investigate the following questions: Can we use this measure of uncertainty to control only viable degrees of freedom? For example, to first control the robot to a more reasonable distance from an initially distant pose and then, when close to target, control the more difficult degrees of freedom. We also want to propagate the error through the pure image based visual servoing control law and compare this to the results presented here. The last question we want to answer is: Can we use this type of evaluation to find favorable feature configurations so to obtain optimal or stable behavior, especially in the case of image based visual servoing, [15].

References

[1] N. R. Gans and S. A. Hutchinson, "An asymptotically stable switched system visual controller for eye in hand robots," in *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, Oct. 2003.

[2] E. Malis, F. Chaumette, and S. Boudet, "2-1/2-d visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, Apr. 1999.

[3] B. Espiau, "Effect of camera calibration errors on visual servoing in robotics," in *3rd International Symposium on Experimental Robotics*, Kyoto, Japan, Oct. 1993.

[4] L. Quan and Z. Lan, "Linear n-point camera pose determination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 8, pp. 774–780, 1999.

[5] P. D. Fiore, "Efficient linear solution of exterior orientation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 140–148, Feb. 2001.

[6] A. Ansar and K. Daniilidis, "Linear pose estimation from points or lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 578–589, May 2003.

[7] R. M. Haralick, H. Joo, C.-N. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim, "Pose estimation from corresponding point data," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1426–1446, 1989.

[8] L. Deng, W. J. Wilson, and F. Janabi-Sharifi, "Characteristics of robot visual servoing methods and target model estimation," in *Proceedings of the 2002 IEEE International Symposium on Intelligent Control*, Vancouver, Canada, Oct 27–30 2002, pp. 684–689.

[9] F. Chaumette, "Potential problems of stability and convergence in image-based nad position-based visual servoing," in *The Confluence of Vision and Control*, ser. Lecture Notes in Control and Information Sciences, no. 237. Springer-Verlag, 1998, pp. 66–78.

[10] E. Malis and F. Chaumette, "Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods," *IEEE Transactions on Robotics and Automation*, 2002.

[11] A. Comport, M. Pressigout, Éric Marchand, and F. Chaumette, "A visual servoing control law that is robust to image outliers," in *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, Oct. 2003, pp. 492–497.

[12] D. Kragic and H. I. Christensen, "Cue integration for visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 1, pp. 18–27, Feb. 2001.

[13] J. Weng, T. S. Huang, and N. Ahuja, "Motion and structure from two perspective views: Algorithms, error analysis, and error estimation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 11, no. 5, pp. 451–476, May 1989.

[14] R. M. Haralick, "Propagating covariance in computer vision," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 10, no. 5, 1996.

[15] E. Malis and P. Rives, "Robustness of image-based visual servoing with respect to depth distribution errors," *IEEE International Conference on Robotics and Automation*, pp. 1056–1061, Sept. 2003.