2016 International Conference on Collaboration Technologies and Systems

Adaptive Kalman Filter Control Law for Visual Servoing

Matthew Marshall
Mechatronics Engineering Department
Kennesaw State University
Kennesaw, GA, USA
mqm@kennesaw.edu

Harvey Lipkin
School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA, USA
harvey.lipkin@me.gatech.edu

***Abstract*—We improve uncalibrated visual servoing for multiple cameras. This moves robots towards being more effective in unstructured environments. Our approach improves performance and reliability by prioritizing data from particular cameras in the system. Adaptive filtering is used to effect this in an uncalibrated situation. Rules are introduced for ensuring positive definite covariance estimates, with the resulting control law then compared to two prior methods. Performance metrics show that the new control law yields lower tracking error than previous approaches and the standard deviations of these metrics indicate that it provides a much more reliable system. Simulation results include improvements up to 33% in the performance metric for a static target and 63% in standard deviation of the performance metric for a moving target.**

***Keywords*—uncalibrated, visual servoing, adaptive Kalman filter, sensor fusion**

I. Introduction

A robot must posses information about its surroundings in order to safely and effectively work alongside people. Integrating visual sensing is one way to enhance robot capabilities. When done in a closed-loop form this control is referred to as visual servoing (VS). With recent increases in digital camera resolution and sensitivity as well as higher processing speed, VS offers ever-more potential for the placement of robots in unstructured environments. Applications include agricultural tasks, food processing, explosive ordinance disposal, and disaster-zone inspections. In VS systems, cameras can be either mounted on the robot end-effector (eye-in-hand) or at a distance such that the end-effector of the robot is in the field of view (eye-to-hand). The camera data is used to track either the robot end-effector, a target object for the robot to follow, or both by identifying a set of features in the images. The aim of VS is to actuate the robot so that the features $\boldsymbol{y}$ from the vision system match a desired set $\boldsymbol{y}^*$. In other words, the robot joint displacements $\boldsymbol{\theta}$ are controlled to minimize the error

$$\boldsymbol{f}(\boldsymbol{\theta}, t) \equiv \boldsymbol{y}(\boldsymbol{\theta}, t) - \boldsymbol{y}^*(\boldsymbol{\theta}, t). \tag{1}$$

This work was funded as an Independent Research and Development project at the Food Processing Technology Division of the Georgia Tech Research Institute.

This describes *image-based* visual servoing, as opposed to *position-based* VS, which utilizes features in Cartesian space. A different VS taxonomy is calibrated or uncalibrated. Calibration refers to kinematic and optical parameters, for example link lengths of the robot, location of the camera, or pitch of the sensor (pixels per mm). Uncalibrated visual servoing is able to control the robot without knowledge of these parameters and is the focus of this work because it offers the benefits of adding vision to a robotic system but does not require careful placement of the cameras.

There are two main components in a visual-servoing system: Jacobian estimation (whether calibrated or uncalibrated) and the control law. The Jacobian relates robot velocity in joint space to the velocity of the end-effector in the image space as $J \equiv \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{\theta}}$. Most VS literature focuses on the first component, Jacobian estimation, and this is especially true for uncalibrated VS.

The algorithm in Piepmeier et al. [20] is used herein for the Jacobian estimation. It has been shown that the population-based method for Jacobian estimation presented by Bonkovic et al. [5] tracks a moving target better than the recursive approach of Piepmeier, though the latter performs better for stationary targets and yields smoother robot motion [7]. A reason for eschewing the population-based method, especially for multi-camera visual servoing, is that population length increases with the number of images and this leads to computational requirements comparable to the cycle time of the robotic system.

A control law combines J with the positions (possibly also velocities and accelerations) of the robot and target as measured by the robot joint encoders and by machine vision to compute a robot action that best reduces $\boldsymbol{f}$. The most common is the Gauss-Newton (GN) algorithm detailed in §II-A. The Kalman filter (KF) control law of [13] is introduced to improve upon traditional uncalibrated visual-servoing control laws such as GN. This recent approach assumes that the process- and measurement-noise statistics are known and that they are constant. The present work addresses the shortcomings of these assumptions and

978-1-5090-2300-4/16 $31.00 © 2016 IEEE
DOI 10.1109/CTS.2016.72

provides an alternative that uses adaptive Kalman filtering.

This work shows that using an adaptive Kalman filter (AKF) as part of the VS control law provides a system with better performance and more consistency than a conventional control law (see §V). In multi-camera visual servoing it's possible for some cameras to have higher signal-to-noise ratios than others due to, for example, lighting variations or different angles between the robot motion and the cameras' optical axes. Using AKF provides a mechanism by which the controller can base the action mostly on high SNR camera data in contrast to other control laws that give equal weight to all cameras.

Various adaptation techniques are reviewed in §III. Also, rules are introduced to ensure the Kalman filter stability. Simulations comparing the AKF with a conventional control law are described in §IV with results presented in §V. Conclusions and direction for future work are provided in §VI. First, a review of the two visual servoing control laws to be compared is presented in §II.

II. Control Law

This section reviews two visual servoing control laws: the ubiquitous Gauss-Newton method and a Kalman filter approach.

A. Gauss-Newton control law

Visual servoing systems are often over constrained where the number of image features is greater than the robot degrees of freedom, and so in general (due to camera noise) there is no robot position that makes the error zero. Therefore, the computation of the desired joint positions is routinely treated as a minimization problem, usually of the error-squared objective function G,

$$G(\boldsymbol{\theta}, t) = \boldsymbol{f}^{\mathsf{T}}(\boldsymbol{\theta}, t) W \boldsymbol{f}(\boldsymbol{\theta}, t),$$

where the weighting matrix W typically is the identity matrix, I, and $\boldsymbol{f}$ is the image-plane error defined in (1).

The robot position that solves the minimization problem at a given time t_k is denoted $\boldsymbol{\theta}^*$ and can be estimated via Newton's method, yielding what has been called the dynamic Newton's method [19]

$$\widehat{\boldsymbol{\phi}}_k = \left(J_k^{\mathsf{T}} J_k + S_k\right)^{-1} \left(J_k^{\mathsf{T}} \boldsymbol{f}_k - J_k^{\mathsf{T}} \frac{\partial \boldsymbol{y}^*(t)}{\partial t} h_t\right),$$

where h_t is the time step, $S_k = \frac{\partial J_k^{\mathsf{T}}}{\partial \boldsymbol{\theta}} \boldsymbol{f}_k$, and

$$\widehat{\boldsymbol{\phi}} \equiv \boldsymbol{\theta}_k - \widehat{\boldsymbol{\theta}_k^*} \tag{2}$$

is the estimated joint-space error vector between the current robot position and the one that minimizes $\boldsymbol{f}^{\mathsf{T}} \boldsymbol{f}$, that is, the goal robot position $\boldsymbol{\theta}^*$. The term $\frac{\partial \boldsymbol{y}^*(t)}{\partial t} h_t$ facilitates servoing to moving targets. Leaving it out gives the more standard Newton's equation for visual servoing. The S term is both difficult to compute and small when near the target so it is often left out. These two omissions yield the Gauss-Newton method:

$$\widehat{\boldsymbol{\phi}}_k = \left(J_k^{\mathsf{T}} J_k\right)^{-1} J_k^{\mathsf{T}} \boldsymbol{f}_k.$$

Gauss-Newton is the most popular control law in visual servoing. For example, Deng and Jägersand [6] use it to compare three different Jacobian estimation methods. It therefore serves as the benchmark for the Kalman filter control law described below.

B. Kalman filter visual servoing control law

The Kalman equations recursively solve weighted least squares for a system with the following process and observation models,

$$\boldsymbol{x}_k = F_{k-1} \boldsymbol{x}_{k-1} + B_k \boldsymbol{u}_k + \boldsymbol{w}_k, \quad \boldsymbol{w}_k \sim N(\boldsymbol{0}, Q_k) \tag{3}$$

$$\boldsymbol{z}_k = H_k \boldsymbol{x}_k + \boldsymbol{v}_k, \quad \boldsymbol{v}_k \sim N(\boldsymbol{0}, R_k) \tag{4}$$

where $N(\boldsymbol{\mu}, \Sigma)$ refers to Gaussian noise with expected value $\boldsymbol{\mu}$ and covariance Σ. At time k, the Kalman filter provides the estimate $\hat{\boldsymbol{x}}_{k|k}$ of the state and $P_{k|k}$, its error covariance matrix. The state being estimated by the KF control law is based on the difference between the current and goal robot positions, that is, $\widehat{\boldsymbol{\phi}}$ from (2).This joint-space error generally is a function of both robot position ($\boldsymbol{\theta}$) and time.

The measurement $\boldsymbol{z}$ for an image-based visual servoing system is the image-plane error $\boldsymbol{f}$. It is necessary to relate the measurements to the state in order to use the Kalman filter. The measurement model uses the following approximation to (1)

$$\boldsymbol{f}_k \equiv \boldsymbol{y}_k - \boldsymbol{y}_k^* \approx J_k(\boldsymbol{\theta}_k - \boldsymbol{\theta}_k^*) = J_k \boldsymbol{\phi}_k, \tag{5}$$

where J_k is the composite Jacobian at time k mapping robot velocity in joint space to the velocity of the end-effector in the image space. Due to non-linear robot kinematics the approximation (5) generally is valid only in the area around $\boldsymbol{\theta}_k$.

Here the system of (3) and (4) is formulated for VS in two ways depending on the order of the state space representation and whether input $\boldsymbol{u}$ is included. For clarity the state and measurement variables $\boldsymbol{x}$ and $\boldsymbol{z}$ are replaced with $\boldsymbol{\phi}$ (or $\begin{bmatrix} \boldsymbol{\phi}^{\mathsf{T}} & \dot{\boldsymbol{\phi}}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$) and $\boldsymbol{f}$.

a) Zeroth order with input: The process equation

$$\boldsymbol{\phi}_k = \boldsymbol{\phi}_{k-1} + \boldsymbol{h}_{\theta,k} + \boldsymbol{w}_k$$

predicts a stationary target position $\boldsymbol{\theta}^*$ (here $\boldsymbol{h}_{\theta,k} \equiv \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-1}$). Using (5), the measurement equation is

$$\boldsymbol{f}_k = J_k \boldsymbol{\phi}_k + \boldsymbol{v}_k.$$

b) First order without input: The process equation

$$\begin{bmatrix} \boldsymbol{\phi}_k \\ \dot{\boldsymbol{\phi}}_k \end{bmatrix} = \begin{bmatrix} I & h_t I \\ 0 & I \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_{k-1} \\ \dot{\boldsymbol{\phi}}_{k-1} \end{bmatrix} + \boldsymbol{w}_k$$

predicts a constant joint-space error velocity $\dot{\boldsymbol{\phi}}$. The measurement equation is

$$\boldsymbol{f}_k = [J_k \; 0] \begin{bmatrix} \boldsymbol{\phi}_k \\ \dot{\boldsymbol{\phi}}_k \end{bmatrix} + \boldsymbol{v}_k.$$

Additional formulations are possible but the *zeroth order with input* and *first order without input* are the most reasonable [12].

C. Control action

The current joint-space error estimate $\widehat{\boldsymbol{\phi}}_k$ generated by the control law is used to set the robot joint offset command as

$$\Delta\boldsymbol{\theta}_{k+1} = \begin{cases} -\widehat{\boldsymbol{\phi}}_{k|k}, & \text{if} \quad \left\| \widehat{\boldsymbol{\phi}}_{k|k} \right\| < \mu \\ -\mu \dfrac{\widehat{\boldsymbol{\phi}}_{k|k}}{\left\| \widehat{\boldsymbol{\phi}}_{k|k} \right\|}, & \text{otherwise} \end{cases}$$

where μ is the maximum allowable joint offset norm, which is necessary to keep the robot motion confined to one VS control period.

As long as μ is within the robot's dynamic limits then the control law described in §II-B and §II-C is shown to be stable under the assumption that the robot starts in a full-rank configuration and passes through at least one more such pose along the servoed path [13].

III. Adaptive Filtering

The Kalman equations (not provided in §II-B) constitute a linear, unbiased, and minimum error variance algorithm; they provide an optimal estimate in the sense that the standard deviation of the error probability density is minimized. To achieve optimality requires complete knowledge of the process and measurement noise statistics, but in practice determining these can be a struggle. Values for the variance-covariance matrices are often based on knowledge of system parameters and on experience. Furthermore, using constant noise covariance matrices is problematic for time-varying error sources.

Adaptive Kalman filtering has been developed to adjust the noise covariance matrices as measurements become available. Research has been active in this field for more than forty years and numerous techniques exist. Mehra [15] describes the following four categories for adaptive filtering methods: Bayesian, maximum likelihood, correlation, and covariance matching. Neither Bayesian nor maximum likelihood methods are favored in practice due to their sometimes excessive computational requirements. Discussions of correlation and covariance-matching approaches follow.

A. Correlation Approaches

Two correlation approaches to adaptive Kalman filtering are discussed here. The classical approach is presented by Mehra [14]. A more recent method comes from Odelson et al. [18] and is called ALS for *autocovariance least-squares*. Different terms, "autocorrelation" and "autocovariance," are used in the two works, but they refer to the same quantity: "the expectation of the data with some lagged version of itself" [18]. Mehra develops two different correlation methods, one based on the measurements $\boldsymbol{z}_k$ and the other on the innovation

$$\boldsymbol{\nu}_k \equiv \boldsymbol{z}_k - H_k \hat{\boldsymbol{x}}_{k|k-1}. \tag{6}$$

The innovation method is more efficient because innovations are less correlated than measurements and is the preferred approach [14] of these two.

In both the classic approach and the ALS method the fundamental quantity is the j-th lag autocovariance of the innovation: $C_{j,k} \equiv \mathrm{E}\left[\boldsymbol{\nu}_k \boldsymbol{\nu}_{k-j}^{\mathsf{T}}\right]$. This expectation is estimated as

$$\widehat{C}_{j,k} = \frac{1}{N-j} \sum_{i=0}^{N-j-1} \boldsymbol{\nu}_{k-i} \boldsymbol{\nu}_{k-i-j}^{\mathsf{T}}, \tag{7}$$

where N is the sample size.

The classic approach, and those based on it, take three steps to estimate Q and R. The first step uses (7) in a least squares problem to estimate PH^{T}. The second calculates

$$\widehat{R} = \widehat{C}_k - H\left(\widehat{PH^{\mathsf{T}}}\right) \tag{8}$$

and the final step determines Q by employing the estimates of PH^{T} and R. Neethling and Young [17] point out that this results in statistical estimates with large variances. One cause, according to Odelson et al. [18], is splitting the estimation of Q and R into two stages. They improve upon the above method with the ALS technique.

ALS is a single-step procedure and, for a reported example, results in estimates with at least an order of magnitude less

variance than the classical approach. It is possible, though, that the estimates of Q and R may not be positive semidefinite. Since such estimates are without physical meaning, the authors add constraints to guarantee positive semidefinite covariances. Åkesson et al. [1] introduce a specialized algorithm to solve this semidefinite programming problem. The new autocovariance least-squares method is promising but relatively untested compared to the number of covariance-matching implementations.

B. Covariance Matching Approaches

Covariance-matching techniques are the most widely used and most intuitive of the four categories referred to by Mehra. These methods work to adjust either Q or R (or both) so that estimated covariances are consistent with the theoretical values. The estimated covariance of the innovations is

$$\widehat{C}_{\nu,k} = \frac{1}{N}\sum_{i=0}^{N-1} \boldsymbol{\nu}_{k-i}\boldsymbol{\nu}_{k-i}^{\mathsf{T}}. \tag{9}$$

The theoretical covariance of the innovations can be obtained by applying the *law of covariance propagation* to the innovation (6). This law states that if $\boldsymbol{Y} = B\boldsymbol{X} + \boldsymbol{l}$ and $\boldsymbol{l}$ is known then the covariances of $\boldsymbol{Y}$ and $\boldsymbol{X}$ are related by $\Sigma_Y = B\,\Sigma_X B^{\mathsf{T}}$. For the case where the vector $\boldsymbol{l}$ is subject to zero-mean noise (independent of $\boldsymbol{X}$) its covariance is included as $\Sigma_Y = B\,\Sigma_X B^{\mathsf{T}} + \Sigma_l$. Applying this to the innovation $\left(\boldsymbol{\nu}_k = \boldsymbol{z}_k - H_k\hat{\boldsymbol{x}}_{k|k-1}\right)$ yields

$$\begin{aligned} E[\boldsymbol{\nu}_k\boldsymbol{\nu}_k^{\mathsf{T}}] &= (-H_k)\Sigma_{x_{k|k-1}}(-H_k)^{\mathsf{T}} + \Sigma_{z_k} \\ &= H_k P_{k|k-1}H_k^{\mathsf{T}} + R_k. \end{aligned} \tag{10}$$

Note that (10) is only approximate since for a sub-optimal filter (one with inaccurate Q and R) $P_{k|k-1}$ is not a true representation of the error covariance of the state estimate.

A covariance matching technique using the innovation yields an estimate of the measurement noise covariance by combining (9) with (10) as

$$\widehat{C}_{\nu,k} = \frac{1}{N}\sum_{i=0}^{N-1} \boldsymbol{\nu}_{k-i}\boldsymbol{\nu}_{k-i}^{\mathsf{T}} \approx E[\boldsymbol{\nu}_k\boldsymbol{\nu}_k^{\mathsf{T}}] = H_k P_{k|k-1}H_k^{\mathsf{T}} + R_k \tag{11}$$

and subtracting HPH^{T} from both sides yields the estimate for R at time k

$$\widehat{R}_{\nu,k} = \frac{1}{N}\sum_{i=0}^{N-1} \boldsymbol{\nu}_{k-i}\boldsymbol{\nu}_{k-i}^{\mathsf{T}} - H_k P_{k|k-1}H_k^{\mathsf{T}}. \tag{12}$$

This differs from (8) in that there is no lag between data samples in the innovation covariance estimate and P comes from the Kalman filter. The value of N is chosen empirically to provide balance between statistical significance (large N) and reactiveness (small N). One consolation about the need to manually tune this parameter is given by Almagbile et al. [2], who report little effect on adaptive filtering performance with changes in the window size.

Myers and Tapley [16] introduce an approach to estimate both Q and R at every iteration, whereas the above method estimates only R. It is noted by Blanchet et al. [4] that using this as a basis to estimate both covariance matrices might give poor performance, "since it is not easy to distinguish between errors in Q and R." Furthermore, the best results with the covariance matching approach are obtained for the case in which Q is known but R is unknown [15].

Covariance matching is used as the basis for many fuzzy logic systems that tune Kalman filter noise covariances (for example, [11], [10], [9]). The shapes of the membership functions in these systems are typically designed by using knowledge of the system [21], [22]. Since one goal of uncalibrated visual servoing is to allow operation with little foreknowledge of the system (cameras, robot, etc.) then the this approach to adaptive filtering is not suitable for uncalibrated visual servoing.

C. Adaptive Measurement Noise Covariance in Uncalibrated Visual Servoing

This section describes how adaptation of the Kalman noise covariance matrices is treated via covariance matching for the present research. Since it is difficult to differentiate between effects of errors in Q and errors in R, one of these matrices is assumed to be known. There are several reasons to choose Q as the assumed covariance. Mehra states that better results are obtained for this case than for the assumed-R-adaptive-Q case. Additionally, one of the research goals is VS improvement by prioritizing data from accurate cameras over noisy cameras. Therefore, in this work Q is assumed known and R is updated adaptively. Q is set to a diagonal matrix βI, where β can be treated as a tuning parameter.

The biggest concern [3] about covariance matching approaches is that they do not necessarily guarantee positive semidefinite estimates of Q and R. It is common in covariance matching techniques to use ad-hoc methods for guaranteeing positive semidefinite estimates. Myers and Tapley advocate always resetting the diagonal elements of $\widehat{R}$ (and $\widehat{Q}$) to the absolute values of their estimates. While this is an instinctual step, consideration of (11) and (12) reveals it to be unsound. According to those equations $\widehat{R}_{\nu,k}$ is negative when the estimated innovation covariance $\widehat{C}_{\boldsymbol{\nu},k}$ is less than the contribution made by the covariance of the prediction error $H_k P_{k|k-1}H_k^{\mathsf{T}}$ to the theoretical innovation covariance. The theoretical value is bigger than the estimate

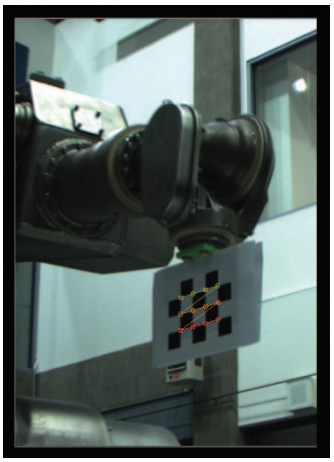

Fig. 1. Image feature extraction: of the twelve checkerboard vertices pictured, the outer four constitute $\boldsymbol{y}$ (goal coordinates not shown).

even if $R = 0$; using $R = \left|\widehat{R}_{\nu,k}\right|$ only exacerbates the problem.

The adaptation technique for the present research is based on (12). Two additional steps are made to ensure a positive semidefinite estimate: (a) $\widehat{R}$ is diagonalized by setting the off-diagonal terms to zero to improve filter stability [8] while yielding comparable accuracy as retaining the off-diagonal elements [4], and (b) a diagonal element is set equal to that of $\widehat{R}_{\nu,k}$ from (12) only if it is positive, otherwise it reverts to the initial value, R_0.

The result is the following update equation for the estimate of R,

$$\widehat{R}_k(i,j) = \begin{cases} \delta_{i,j}\widehat{R}_{\nu,k}(i,j), & \text{if} \quad \widehat{R}_{\nu,k}(i,j) > 0 \\ R_0(i,j), & \text{otherwise} \end{cases}$$

where $R_0 = \kappa I$. Like β for Q, κ can be treated as a tuning parameter.

IV. Simulation

Three stationary cameras observe the six-axis manipulator so the results presented here are for the eye-to-hand case. Figure 1 shows a checkerboard pattern mounted to the robot tool flange. The image-plane coordinates of the four outermost detected vertices make up the image feature vector of the i-th camera

$$\boldsymbol{y}^{(i)} = [u_1^{(i)}\ v_1^{(i)}\ u_2^{(i)}\ v_2^{(i)}\ u_3^{(i)}\ v_3^{(i)}\ u_4^{(i)}\ v_4^{(i)}]^\mathsf{T},$$

which is stacked with the other camera data to form the image feature vector as $\boldsymbol{y} = [\boldsymbol{y}^{(1)}\ \boldsymbol{y}^{(2)}\ \boldsymbol{y}^{(3)}]^\mathsf{T}$.

Simulated features from the vision system are obtained using a pinhole camera model with

$$\begin{bmatrix} u_p \\ v_p \end{bmatrix} = \frac{\lambda}{z_c} \begin{bmatrix} x_c \\ y_c \end{bmatrix},$$

where λ is the focal length and (u_p, v_p) are the image-plane coordinates of a point located at $[x_c \quad y_c \quad z_c]^\mathsf{T}$ relative to the camera. White noise is added to u_p and v_p in order to more closely approximate real conditions.

Two camera-parameter sets are employed, simulating a high-cost camera and a low-cost camera. The high-cost camera model has a 10 mm focal length and a maximum additive noise magnitude of 0.5 pixels. A low-cost camera likely has a wide-angle lens and a smaller sensor and is modeled as $\lambda = 3.95$ mm with 2 pixels of noise.

Simulations use heterogeneous camera combinations to represent unstructured environments, where conditions such as lighting vary from camera to camera. All possible combinations of high- and low-cost cameras for a three-camera system are used. That is, $C_H + C_L = C = 3$ with $C_H = 0 \ldots 3$, where C_H is the number of high-cost cameras used and C_L is the number of low-cost cameras.

Two different tasks are performed. In one the VS system servos to a stationary target and in the other it tracks a moving one. For static-target trials the maximum joint command norm, μ, is one degree. Static-target trials are concluded when the image-plane error norm in each of the C cameras is less than some threshold value ϵ, or $\left\|\boldsymbol{f}^{(i)}\right\| < \epsilon$ for $i = 1 \ldots C$. For moving-target trials the translational speed is 100 mm/sec and μ is a much larger eight degrees in order to simulate the lab's physical system that suffers from USB throughput limitations.

V. Results

Two Kalman filter (KF) formulations are tested: the zeroth-order with input and the first-order without input. The noise covariance matrices are $Q = 5I$ and $R_0 = I$ since this combination has the best performance of six combinations tested by Marshall [13]. The KF formulations are employed both with adaptation and with fixed $R = R_0$. Gauss-Newton is also tested, which results in five different controllers tested.

For each camera scenario one hundred trials are run, each with a different random-number-generator seed that is used for position and orientation of each camera, plus additive noise in robot joint position and pixel coordinates. When comparing controllers, the same noise seed is used. For example, Trial 10 of the $C_H = 3$, static-target case using zeroth order AKF has the same added camera noise as Trial 10 of the $C_H = 3$, static-target case using GN. Ramifications

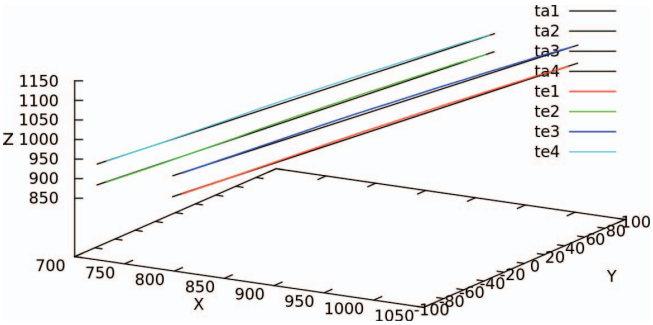

Fig. 2. Exemplar static-target trial paths for checkerboard corners: ideal (ta1–ta4) and servoed (te1–te4) using first-order-no-input AKF with, two high-cost cameras and one low-cost camera ($C_H = 2$ and $C_L = 1$); units are in mm

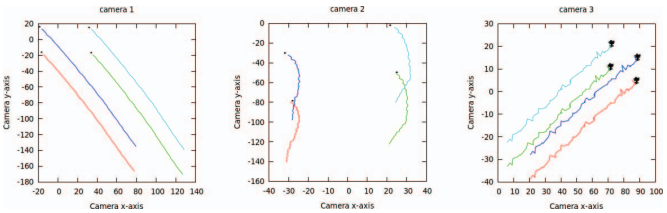

Fig. 3. Image-plane data from three cameras for $C_H = 2$ ("camera 1" and "camera 2") and $C_L = 1$ ("camera 3") static-target trial using first-order-no-input AKF; units are in pixels

include identically positioned cameras for both trials and identical target pixel coordinates.

The robot path for a static-target trial using KF is shown in Figures 2 (3-D coordinates) and 3 (image-plane coordinates). The straight lines in Figure 2 represent the ideal (shortest) checkerboard path between the start and end positions. The starting point is at the bottom left in the figure. The final offset between target and servoed checkerboard positions is a function of the threshold value ϵ. The lines in Figure 3 trace the servoed image-plane path in the three cameras and the dots at the ends are the target image coordinates; the relative difference in signal-to-noise ratios between high- and low-cost camera models can be seen by comparing the data from "camera 1" and "camera 3" here. The random-number seed for this trial results in 113 steps for convergence with GN and just 52 with AKF. Such improvement over the conventional control law is repeatable (see Figure 4, which is described below).

The performance metric for static-target servoing is the average number of iterations to convergence

$$\bar{S} = \frac{1}{T}\sum_{a=1}^{T} S_a,$$

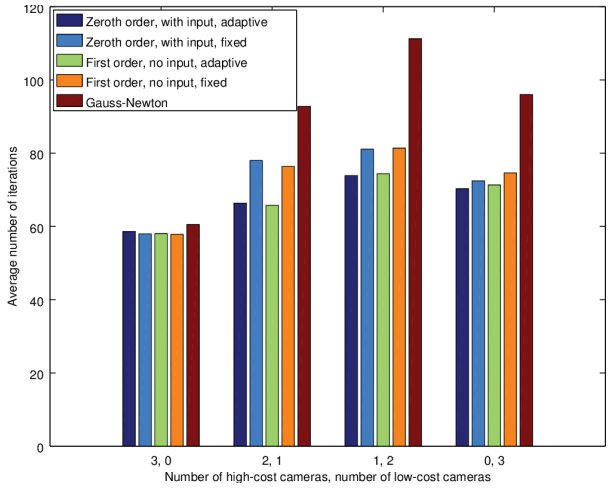

Fig. 4. Static target: $\bar{S}$ using combinations of high- and low-cost cameras with five different controllers

where a is the trial number, S_a is the number of iterations to convergence for the a-th trial, and T is the total number of trials; here $T = 100$. Convergence means the image-plane error norm is less than ϵ in all $C = 3$ cameras. For these trials the threshold is $\epsilon = 10$ pixels.

Figure 4 shows $\bar{S}$ for the five controllers tested in four different three-camera scenarios, each taken over one hundred trials. $\bar{S}$ is plotted along the vertical axes, the different camera scenarios are listed along the horizontal, and the legend identifies the control laws. Three observations from this chart are (1) zeroth- and first-order KF have similar performance; (2) KF outperforms GN; and (3) adaptation provides further gains. For example, the trials with two high- and one low-cost camera show that on average GN takes ninety-three iterations to reach the target, zeroth- and first-order non-adaptive KF take about seventy-seven, and the two AKFs require just sixty-six. Regarding this performance metric, the Kalman filter control laws outperform Gauss-Newton, with the adaptive versions doing better than the non-adaptive KF controllers.

For moving-target servoing, the performance metric is the Cartesian error norm for the checkerboard corners, averaged over all $T \times S$ iterations, where T is the number of trials run and S is the number of iterations in a given trial. That is,

$$\bar{e}_{\text{check},T,S} = \frac{1}{T \times S}\sum_{a=1}^{T}\sum_{j=1}^{S} \left\| \boldsymbol{p}_{\text{check},a,j} - \boldsymbol{p}^{*}_{\text{check},j} \right\|, \quad (13)$$

where $\boldsymbol{p}_{\text{check},a,j} = \begin{bmatrix}\boldsymbol{p}_{1,a,j} & \boldsymbol{p}_{2,a,j} & \boldsymbol{p}_{3,a,j} & \boldsymbol{p}_{4,a,j}\end{bmatrix}^{\mathsf{T}}$ comprises the $\boldsymbol{p}_{i,a,j}$ Cartesian position vectors to the the $i = 1 \ldots 4$ corners of the checkerboard at time step j of

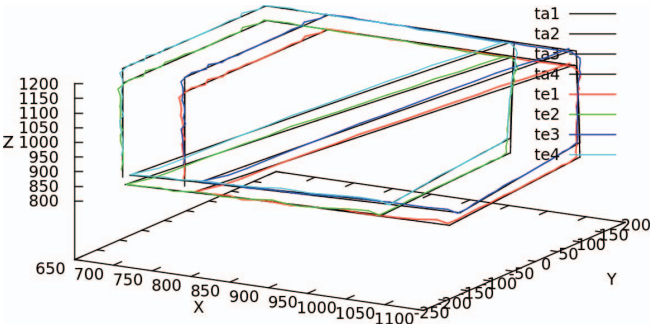

Fig. 5. Exemplar moving-target trial paths for checkerboard corners: ideal (ta1–ta4) and servoed (te1–te4) using first-order-no-input AKF, one high-cost camera and two low-cost cameras ($C_H = 1$ and $C_L = 2$); units are in mm

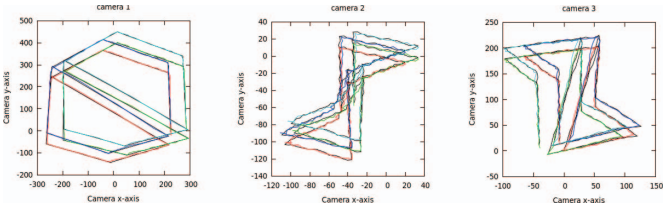

Fig. 6. Image-plane data from three cameras for $C_H = 1$ ("camera 1") and $C_L = 2$ ("camera 2" and "camera 3") moving-target trial using first-order-no-input AKF; units are in pixels

trial a. Note that in (13) neither $\boldsymbol{p}^*$ nor S has an a subscript because neither the Cartesian coordinates of the target path nor the number of iterations change between moving-target trials.

Figures 5 (3-D coordinates) and 6 (image-plane coordinates) show the checkerboard paths—ideal and servoed—for a moving-target trial using AKF. As a reference point the average error norm for the checkerboard corners in this trial is $\bar{e}_{\text{check}} = 9.37$ mm.

Figure 7 shows $\bar{e}_{\text{check},T,S}$ for the five controllers tested in four different three-camera scenarios, each taken over one hundred trials. It is apparent in this chart that the trends observed in Figure 4 are weaker for the moving-target case, though KF still outperforms GN and adaptation still provides gains. This could be merely due to the different maximum-joint-command-norm (μ) values used in the two scenarios.

These average-performance metrics indicate that KF and AKF outperform Gauss-Newton, but the standard deviations of these metrics shows this even more. Figures 8 and 9 show the standard deviations corresponding to Figures 4 and 7. It is evident that KF and especially AKF reduce outliers.

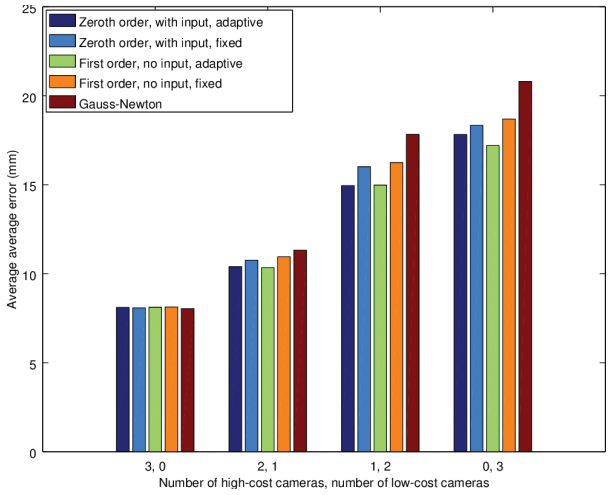

Fig. 7. Moving target: $\bar{e}_{\text{check},T,S}$ using combinations of high- and low-cost cameras with five different controllers

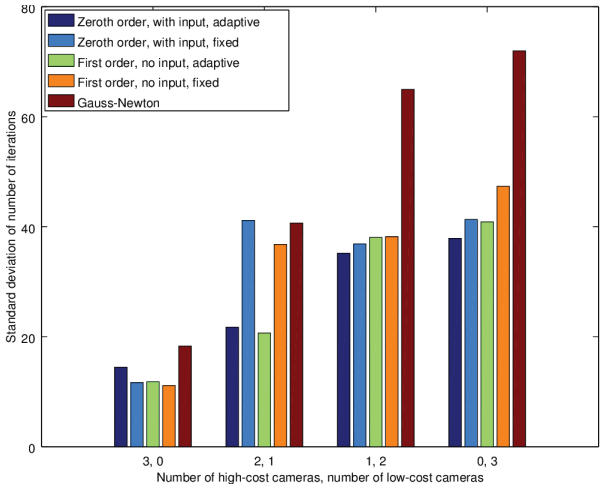

Fig. 8. Static target: standard deviation in $\bar{S}$

Consider, for example, using three noisy cameras when servoing to a moving target (Figure 9); in this situation the performance-metric variation for first-order-no-input AKF is doubled when using its non-adaptive counterpart, and tripled with Gauss-Newton. The average error norms for the controllers in this situation are within twenty percent of each other but some of the random situations (that is, camera placement and noise values) led to such erratic behavior from the non-adaptive controllers that the standard deviation for Gauss-Newton is over three hundred percent of that for AKF.

VI. Conclusion and Future Work

A covariance-matching approach for adaptation in the Kalman filter based visual servoing control law is presented.

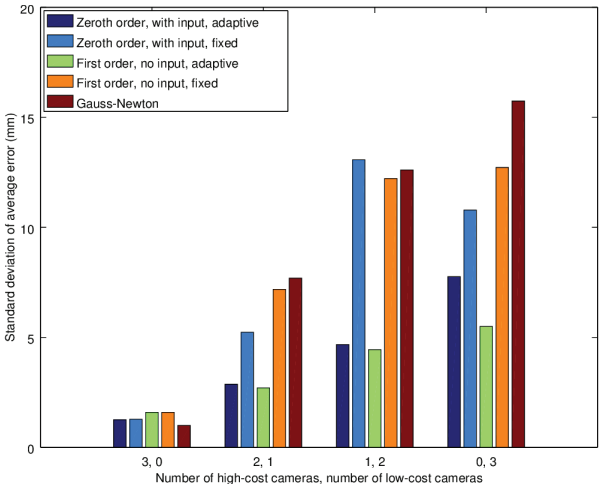

Fig. 9. Moving target: standard deviation in $\bar{e}_{\text{check},T,S}$

This method uses ad-hoc rules for ensuring positive semidefinite covariance estimates and it improves servoing performance with both stationary and moving targets. Average performance metrics are better across the board with AKF but reductions in standard deviations are even more dramatic. This means that the roboticist can reap the benefits of camera guidance without the burdens of careful placement and calibration.

Since adaptation adds so much benefit to uncalibrated visual servoing, future work should focus on an improved technique. The ALS technique is attractive because it estimates both process and measurement covariance matrices.

REFERENCES

[1] B.Akesson, J.,Jrgensen, N.Poulsen, and S.Jorgensen, "A generalized autocovariance least-squares method for kalman filter tuning," Journal of Process Control, vol. 18, no. 78, pp. 769–779, 2008.

[2] A.Almagbile, J.Wang, and W.Ding, "Evaluating the performances of adaptive kalman filter methods in gps/ins integration," Journal of Global Positioning Systems, vol. 9, pp. 33–40, 2010.

[3] V.Bavdekar, A.Deshpande, and S.Patwardhan, "Identification of process and measurement noise covariance for state and parameter estimation using extended kalman filter," Journal of Process Control, vol. 21, no. 4, pp. 585–601, 2011.

[4] I.Blanchet, C.Frankignoul, and M.Cane, "A comparison of adaptive Kalman filters for a tropical pacific ocean model," Monthly Weather Review, vol. 125, pp. 40–58, 1997.

[5] M.Bonkovic, A.Hace, and K.Jezernik, "Population-based uncalibrated visual servoing," Mechatronics, IEEE/ASME Transactions on, vol. 13, pp. 393–397, June 2008.

[6] Z.Deng and M.Jagersand, "Evaluation of model independent image-based visual servoing," Computer and Robot Vision, 2004. Proceedings. First Canadian Conference on, pp. 138–144, 17-19, 2004.

[7] M.Hao and Z.Sun, "A universal state-space approach to uncalibrated model-free visual servoing," Mechatronics, IEEE/ASME Transactions on, vol. PP, no. 99, pp. 1–14, 2011.

[8] C.Hide, T.Moore, and M.Smith, "Adaptive kalman filtering algorithms for integrating gps and low cost ins," Position Location and Navigation Symposium, pp. 227–233, April 2004.

[9] Y.Ip, A.Rad, Y.Wong, Y.Liu, and X.Ren, "A localization algorithm for autonomous mobile robots via a fuzzy tuned extended kalman filter," Advanced Robotics, vol. 24, no. 1/2, pp. 179–206, 2010.

[10] D.Jwo and F.Chang, "A fuzzy adaptive fading kalman filter for gps navigation," Advanced Intelligent Computing Theories and Applications, vol. 4681 of Lecture Notes in Computer Science, pp. 820–831, Springer Berlin/Heidelberg, 2007.

[11] X.Lv and X.Huang, "Fuzzy adaptive kalman filtering based estimation of image jacobian for uncalibrated visual servoing," Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, pp. 2167–2172, October 2006.

[12] M.Marshall, Multi-Camera Uncalibrated Visual Servoing. PhD thesis, Georgia Institute of Technology, 2013.

[13] M.Marshall and Lipkin, H."Kalman filter visual servoing control law," Mechatronics and Automation (ICMA), 2014 IEEE International Conference on, pp. 527–534, 3-6 August 2014.

[14] R.Mehra, "On the identification of variances and adaptive kalman filtering," Automatic Control, IEEE Transactions on, vol. 15, pp. 175–184, April 1970.

[15] R.Mehra, "Approaches to adaptive filtering," Automatic Control, IEEE Transactions on, vol.17, no.5, pp.693–698, October 1972.

[16] K.Myers and B.Tapley, "Adaptive sequential estimation with unknown noise statistics," Automatic Control, IEEE Transactions on, vol. 21, pp. 520–523, August 1976.

[17] C.Neethling and P.Young, "Comments on identification of optimum filter steady-state gain for systems with unknown noise covariances," Automatic Control, IEEE Transactions on, vol. 19, no. 5, pp. 623–625, 1974.

[18] B.Odelson, M.Rajamani, J.Rawlings, "A new autocovariance least-squares method for estimating noise covariances," Automatica, Volume 42, Issue 2, pp. 303–308, February 2006.

[19] J.Piepmeier, "Experimental results for uncalibrated eye-in-hand visual servoing," IEEE Southeastern Symposium on System Theory, Morgantown, WV, pp. 335–339, 2003.

[20] J.Piepmeier, G.McMurray, and H.Lipkin, "Uncalibrated dynamic visual servoing," IEEE Transactions on Robotics and Automation, vol. 20, pp. 143–147, February 2004.

[21] T.Takagi and M.Sugeno, "Fuzzy identification of systems and its applications to modeling and control," Systems, Man and Cybernetics, IEEE Transactions on, vol. SMC-15, pp. 116–132, January–February 1985.

[22] S.Zhang and X.Wei, "Fuzzy adaptive kalman filtering for dr/gps," Machine Learning and Cybernetics, 2003 International Conference on, vol. 5, pp. 2634–2637, November 2003.