

Uncalibrated Dynamic Visual Servoing

J. A. Piepmeier, G.V. McMurray, H. Lipkin

Abstract—A dynamic quasi-Newton method for uncalibrated, vision-guided robotic tracking control with fixed imaging is developed and demonstrated. This method does not require calibrated kinematic and camera models. Robotic control is achieved at each step through minimizing a nonlinear objective function by taking quasi-Newton steps and estimating the composite Jacobian at each step. The Jacobian is estimated using a dynamic recursive least squares algorithm. Experimental results demonstrate the validity of this approach.

Index Terms—Uncalibrated visual servoing, dynamic nonlinear least squares, Jacobian estimation.

I. INTRODUCTION

This article develops an uncalibrated, vision-guided, robotic control method with a fixed imaging system. The controller is a dynamic quasi-Newton method based on nonlinear least squares optimization methods. The system model is approximated using a dynamic Jacobian estimation scheme. Using this controller, the robot can be servoed to both static and moving targets, even with uncalibrated robot kinematics and camera models. The control method is completely independent of robot type, camera type, and camera location. In other words, it is independent of the system model.

By rejecting a model based paradigm, the control algorithm eliminates the necessity of extensive system modeling, system re-calibration, and ancillary hardware that constrains the workspace. In a manufacturing setting, these are nonvalue-added activities; reducing or eliminating the amount of time spent on them reduces the overall cost of a product.

Specifically, the contributions of this work are:

- Algorithms are explicitly derived to track a moving target using uncalibrated visual servoing with stationary cameras.
- A theoretical basis for uncalibrated vision-guided control algorithms is developed based on the direct minimization of Frobenius norms.
- The developed algorithms are experimentally verified for uncalibrated vision-guided robotic control and tracking.

Quasi-Newton methods implementing Jacobian estimation have been used with success to servo a robot to a static target. This work seeks to thoroughly develop the quasi-Newton control method for target tracking.

There are several instances in the literature where Jacobian estimation is used in visual servoing. Hosoda and Asada [1] and Hosoda, Igarashi, and Asada [2] have presented uncalibrated vision-guided robotic control for static targets using

fixed cameras.. The Jacobian estimation scheme utilized in [1] is similar to an exponentially weighted recursive least squares update equation for vectors. However, the vector format is applied to the Jacobian matrix estimation problem. In [3], Asada, Tanaka, and Hosoda extend the Jacobian estimation scheme to eye-in-hand stereo tracking of moving objects using static reference points to estimate target motion.

Jagersand [4], [5] takes the approach of a nonlinear least squares optimization method utilizing a trust region method and Broyden estimation. The moving target scenario is not addressed.

Previous work has shown the feasibility of uncalibrated visual servo control for stationary targets. However, a control law for the moving target scenario has not been rigorously developed.

II. AN UNCALIBRATED DYNAMIC VISUAL SERVOING METHOD

A *stationary* vision system is assumed that can sense sufficient end-effector and target features to locate both bodies in space. This renders the target features, $y^*(t)$, as functions of only time t and the end-effector features, $y(\theta)$, as functions of only the robot joint angles, $\theta \in R^n$. It is important to note that t and θ are independent variables since as time varies the joint angles can be held constant, and conversely at every given time the joint angles can take on any values. There are no assumptions yet about target tracking. To optimally track the target, a constraint relationship is imposed between θ and t so joint angles are selected as a function of time, $\theta(t) = g(y^*(t))$. This establishes an optimal end-effector trajectory $y(\theta(t))$ to follow the moving target. The constraint is established by minimizing the tracking error, $f \in R^m$, as seen in the image plane

$$f(\theta, t) = y(\theta) - y^*(t) \quad (1)$$

The combined transformations of forward kinematics and imaging geometry render $f(\theta, t)$ a highly nonlinear function. This multivariate optimization problem is solved at each increment by a dynamic quasi-Newton controller (Section II-A) with a dynamic Jacobian estimator (Section II-B).

A. A Dynamic Newton's Method

The imposed trajectory $\theta(t)$ that causes the end-effector to follow the target is established by minimizing the image error squared

$$F(\theta, t) = \frac{1}{2} f^T(\theta, t) f(\theta, t)$$

which can also be modified by a weighting matrix but is omitted for simplicity. The Taylor series expansion about (θ, t)

J. A. Piepmeier is with the Systems Engineering Department, U.S. Naval Academy, Annapolis, Maryland (email: piepmeie@usna.edu).

G. V. McMurray is with the Georgia Tech Research Institute, Atlanta, Georgia (email: gary.mcmurray@gtri.edu).

H. Lipkin is with the George Woodruff School of Mechanical Engineering, Georgia Institute of Technology (email: harvey.lipkin@me.gatech.edu).

is

$$F(\theta + h_\theta, t + h_t) = F(\theta, t) + F_\theta h_\theta + F_t h_t + \dots$$

where F_θ and F_t are partial derivatives and h_θ and h_t are increments of θ and t . For a fixed sampling period h_t , F is minimized by solving

$$\begin{aligned} 0 &= \frac{\partial F(\theta + h_\theta, t + h_t)}{\partial \theta} \\ &= F_\theta + F_{\theta\theta} h_\theta + F_{t\theta} h_t + O(h^2) \end{aligned} \quad (2)$$

where $O(h^2)$ indicates second order terms in h_t and h_θ . Dropping these terms, rearranging, expanding the partials, and adding θ gives what is referred to here as a dynamic Newton's method

$$\theta + h_\theta = \theta - (J^T J + S)^{-1} J^T (f + \frac{\partial f}{\partial t} h_t) \quad (3)$$

where J is the joint-to-image feature error composite Jacobian and

$$\begin{aligned} J &\equiv \frac{\partial f}{\partial \theta}, \quad S \equiv \frac{\partial J^T}{\partial \theta} f \\ F_\theta &= J^T f, \quad F_{\theta\theta} = J^T J + S, \quad F_{t\theta} = J^T \frac{\partial f}{\partial t} \end{aligned}$$

To compute the terms S and J analytically requires a calibrated system model. The term S is difficult to estimate, but as θ approaches the solution, it approaches zero and it is often dropped to give what is sometimes called a Gauss-Newton method. It can be shown that for a small enough time increment, h_t , the method is well-defined for all θ and converges linearly to a finite steady-state error [6]. When an estimated Jacobian, \hat{J} , is used the algorithm becomes a dynamic quasi-(Gauss-)Newton method such that at the k^{th} increment,

$$\theta_{k+1} = \theta_k - (\hat{J}_k^T \hat{J}_k)^{-1} \hat{J}_k^T (f_k + \frac{\partial f_k}{\partial t} h_t) \quad (4)$$

where $h_\theta = \theta_{k+1} - \theta_k$.

The qualifier “dynamic” specifically refers to the presence of the error velocity term $\frac{\partial f_k}{\partial t}$ which is used to linearly predict the error vector at the next time increment as $f_{k+1} \approx f_k + \frac{\partial f_k}{\partial t} h_t$, assuming the robot remains at its current position. Clearly more complex predictors are possible but the linear one has appealing simplicity. When the predictor is absent the method is referred to here as “static” so (4) becomes

$$\theta_{k+1} = \theta_k - (\hat{J}_k^T \hat{J}_k)^{-1} \hat{J}_k^T f_k \quad (5)$$

which is the basis for previous work by [1], [5], and [7]. The derivation is similar to the dynamic case if the error is only a function of the joint angles, $f(\theta)$.

The static and dynamic methods coincide in two distinct cases. First, if the time step h_t is made sufficiently small in relation to $\frac{\partial f_k}{\partial t}$ then it appears as if the error f_k is constant or static (sometimes this is called a quasi-static condition). This suggests that the dynamic method should be expected to track with slower required sampling rates and to track more demanding error functions compared to the static method. Second, since $\frac{\partial f}{\partial t} = \frac{\partial}{\partial t}(y(\theta) - y^*(t)) = -\frac{\partial y^*}{\partial t}$, then $\frac{\partial f}{\partial t}$ vanishes if the target is fixed, $y^* = 0$. Thus the predictor term attempts to compensate for target motion should result in better tracking than the static method for moving targets.

B. A Dynamic Quasi-Newton Method

As with Newton's method, the moving target scenario requires that the appropriate derivatives are included in estimating the Jacobian, \hat{J}_k . For static target servoing, Jagersand in [4] and [5] employs Broyden's method for Jacobian estimation. In this section, a dynamic Broyden's method is derived. Next, a similar Jacobian estimation scheme is developed that uses a recursive least squares algorithm and is more stable in the presence of noise. Finally, the steady state performance of a quasi-Newton method using the recursive least squares estimation is discussed.

1) *Derivation of a Dynamic Jacobian Estimation Scheme:* Several Jacobian estimation schemes using Broyden's method or a similar variant have been implemented as discussed in [1]-[5]. Analogous to the controller, the proposed dynamic Broyden update contains an additional term, $\frac{\partial f_k}{\partial t} h_t$. This dynamic Broyden's update is derived for a moving target scenario by extending the derivation of the static Broyden's update given in Dennis and Schnabel [8].

The affine model (a linear model that does not necessarily pass through the origin [8]) of the error function $f(\theta, t)$ is a first order Taylor series approximation denoted as $m(\theta, t)$. The model is updated at each iteration and for the k^{th} ,

$$m_k(\theta, t) = f_k + \hat{J}_k(\theta - \theta_k) + \frac{\partial f_k}{\partial t}(t - t_k) \quad (6)$$

Requiring that the k^{th} affine model correctly specifies the error at the $(k-1)^{th}$ increment, $m_k(\theta_{k-1}, t_{k-1}) = f_{k-1} = (f(\theta_{k-1}, t_{k-1}))$, yields the so-called secant equation

$$\hat{J}_k h_\theta + \frac{\partial f_k}{\partial t} h_t = \Delta f \quad (7)$$

where $\Delta f = f_k - f_{k-1}$. Broyden's method requires that (7) holds. Subtracting $\hat{J}_{k-1} h_\theta$ from each side, rearranging, and transposing gives,

$$h_\theta^T \Delta \hat{J}^T = (\Delta f - \frac{\partial f_k}{\partial t} h_t - \hat{J}_{k-1} h_\theta)^T \quad (8)$$

where $\Delta \hat{J} = \hat{J}_k - \hat{J}_{k-1}$. The Jacobian update $\Delta \hat{J}$ is selected to minimize the Frobenius norm $\|\Delta \hat{J}\|_F = (\sum (\Delta \hat{J}_{ij})^2)^{\frac{1}{2}}$ subject to the constraint (8) where $(\Delta \hat{J})_{ij}$ indexes $\Delta \hat{J}$. By stacking the elements into a vector and rewriting (8) accordingly, the problem is cast into a familiar form with a minimum norm solution. Unstacking the result gives the dynamic Broyden update (see [6], [9] for details),

$$\hat{J}_k = \hat{J}_{k-1} + \frac{(\Delta f - \hat{J}_{k-1} h_\theta - \frac{\partial f_k}{\partial t} h_t) h_\theta^T}{h_\theta^T h_\theta} \quad (9)$$

The qualifier “dynamic” specifically refers to the presence of the error velocity term $\frac{\partial f_k}{\partial t}$. If it is assumed that the affine model is only a function of the joint angles, $m(\theta)$, then the “static” Broyden update results,

$$\hat{J}_k = \hat{J}_{k-1} + \frac{(\Delta f - \hat{J}_{k-1} h_\theta) h_\theta^T}{h_\theta^T h_\theta} \quad (10)$$

The static and dynamic Broyden methods are entirely analogous to the static and dynamic Newton methods. They coincide when either the time increment is very small or the target is

stationary. The dynamic method should be expected to be more effective with slower sampling rates and should have better tracking of moving targets.

As a consequence of the secant condition the affine models are piecewise continuous so at the $(k-1)^{th}$ increment,

$$m_k(\theta_{k-1}, t_{k-1}) - m_{k-1}(\theta_{k-1}, t_{k-1}) = 0$$

which is easily verified using (6) for m_k and m_{k-1} at (θ_{k-1}, t_{k-1}) , and (7). The k^{th} affine model must contain the points f_{k-1} and f_k as well as satisfy the secant equation. However consider the case where noise is present in the system. The measurement f_k includes a nominal value of the error function \tilde{f}_k as well as system noise δ_{noise} so that $f_k = \tilde{f}_k + \delta_{noise}$. In the event of large values for δ_{noise} , requiring that the secant equation holds may result in an affine model m_k which introduces large errors in \hat{J}_k . Large errors in \hat{J}_k will result in poor tracking. Thus, the dynamic Broyden's method may perform poorly for noisy systems and can be improved by a recursive formulation that provides a filtering action.

2) *A Dynamic Jacobian Estimation: Recursive Least Squares:* Increased stability can be achieved using an exponentially weighted recursive least squares (RLS) algorithm [10] that minimizes a cost function G_k based on the change in the affine model

$$G_k = \sum_{i=1}^k \lambda^{k-i} \|m_k(\theta_{i-1}, t_{i-1}) - m_{i-1}(\theta_{i-1}, t_{i-1})\|^2 \quad (11)$$

The objective function in (11) is an exponentially weighted sum of the differences between the current affine model and past affine models where $0 < \lambda \leq 1$. By minimizing G_k the secant equation (7) no longer holds nor are the affine models piecewise continuous. Minimizing G_k is equivalent to minimizing the Frobenius norm of the term $(\Delta M)^T \Lambda (\Delta M)$ where ΔM is a $k \times m$ matrix whose i th column is $m_k(\theta_{i-1}, t_{i-1}) - m_{i-1}(\theta_{i-1}, t_{i-1})$ and Λ is a $k \times k$ diagonal matrix with λ^{k-i} at the i th diagonal element.

Following the approach taken in Brogan [9], it can be shown [6] that the desired recursive estimation scheme that minimizes (11) is given by,

$$\hat{J}_k = \hat{J}_{k-1} + \frac{\left((f_k - f_{k-1}) - \hat{J}_{k-1} h_\theta - \frac{\partial f_k}{\partial t} h_t\right) h_\theta^T P_{k-1}}{\lambda + h_\theta^T P_{k-1} h_\theta} \quad (12)$$

$$P_k = \frac{1}{\lambda} \left(P_{k-1} - \frac{P_{k-1} h_\theta h_\theta^T P_{k-1}}{\lambda + h_\theta^T P_{k-1} h_\theta} \right) \quad (13)$$

As before, the static version omits the term $\frac{\partial f_k}{\partial t} h_t$.

The form is similar to the dynamic Broyden's method in (9) with the exception of λ and P_k . The weighting parameter $0 < \lambda \leq 1$ can be tuned to average in more or less of the previous information where the effective number of terms is estimated by $\frac{1}{1-\lambda}$. As shown in [9], it is possible to use more complex weighting matrices in the formulation. However this has not been done here due to a lack of a specific physical justification and the desire for the simplest equations.

The RLS algorithm is often used for system identification [11]. Indeed, the formulation presented here identifies the system model (the Jacobian) for visually guided control. It is interesting to note that in the literature [9], [10] the RLS algorithm is typically used for the estimation of a vector; however, the algorithm can be shown [6] to extend to the estimation of a matrix. Equations (12) and (13) define a recursive update for estimating J_k .

A dynamic quasi-Newton method using the RLS Jacobian estimation scheme follows:

Algorithm 1: Given $f : R^n \rightarrow R^m$; $\theta_0, \theta_1 \in R^n$; $\hat{J}_0 \in R^{m \times n}$, $P_0 \in R^{n \times n}$, $\lambda \in (0, 1)$

Do for $k = 1, 2, \dots$

$$\Delta f = f_k - f_{k-1}$$

$$\frac{\partial f_k}{\partial t} h_t = -(y_k^* - y_{k-1}^*)$$

$$h_\theta = \theta_k - \theta_{k-1}$$

$$\hat{J}_k = \hat{J}_{k-1} + (\lambda + h_\theta^T P_{k-1} h_\theta)^{-1} \cdot (\Delta f - \hat{J}_{k-1} h_\theta - \frac{\partial f_k}{\partial t} h_t) h_\theta^T P_{k-1}$$

$$P_k = \frac{1}{\lambda} (P_{k-1} - (\lambda + h_\theta^T P_{k-1} h_\theta)^{-1} (P_{k-1} h_\theta h_\theta^T P_{k-1}))$$

$$\theta_{k+1} = \theta_k - (\hat{J}_k^T \hat{J}_k)^{-1} \hat{J}_k^T (f_k + \frac{\partial f_k}{\partial t} h_t)$$

End for

End

C. Convergence Analysis of Dynamic Quasi-Newton Method using RLS Jacobian Estimation

The previous section gives a clear rational for the formulation of the Jacobian estimation scheme as an RLS algorithm. In this section, the convergence properties using Algorithm 1 are addressed. Let θ_k^* represent the joint position for which the objective function is minimized. For the dynamic quasi-Newton method using RLS Jacobian estimation, it is shown that under certain assumptions, the norm of the joint error $\theta_k^* - \theta_k$ is bounded. This is accomplished using the steady state properties of the RLS algorithm given by (12) and (13) as discussed in [12].

It is readily verified that the dynamic quasi-Newton method (4) can be expressed as the equivalent form

$$h_{\theta_{k+1}}^* = \hat{J}_k^+ (f_k + \frac{\partial f_k}{\partial t} h_t + J_k^* h_{\theta_k}^* + E_k h_{\theta_k}^* + \hat{J}_k \Delta \theta^*) \quad (14)$$

by using the substitutions $h_{\theta_k}^* = \theta_k^* - \theta_k$, $\hat{J}_k^+ = (\hat{J}_k^T \hat{J}_k)^{-1} \hat{J}_k^T$, $J_k^* = J_k(\theta_k^*)$, $E_k = \hat{J}_k - J_k^*$, $\Delta \theta^* = \theta_{k+1}^* - \theta_k^*$ and the identity $\hat{J}_k^+ \hat{J}_k = I$. From the Taylor series expansion of θ_{k+1}^*

$$\frac{\partial \theta_k^*(t)}{\partial t} h_t = \theta_{k+1}^* - \theta_k^* + O(h_t^2)$$

and since $\frac{\partial f_k}{\partial t} h_t = -\frac{\partial y_k^*}{\partial t} h_t = -J_k^* \frac{\partial \theta_k^*(t)}{\partial t} h_t$ then

$$\frac{\partial f_k}{\partial t} h_t = -J_k^* \Delta \theta^* - J_k^* O(h_t^2) \quad (15)$$

so that with $f_k^* = f_k(\theta_k^*, t_k) = 0$ equation (14) becomes

$$h_{\theta_{k+1}}^* = \hat{J}_k^+ (f_k - f_k^* + J_k^* h_{\theta_k}^* + E_k h_{\theta_k}^* + E_k \Delta \theta^* - J_k^* O(h_t^2)) \quad (16)$$

If J^* is Lipschitz continuous with a Lipschitz constant γ , then taking the norm of (16) results in

$$\begin{aligned} \|h_{\theta_{k+1}}^*\| &\leq \|\hat{J}_k^+\| \left(\frac{\gamma}{2} \|h_{\theta_k}^*\| + \|E_k\| \right) \|h_{\theta_k}^*\| \\ &\quad + \|\hat{J}_k^+\| \|E_k\| \|\Delta\theta^*\| + c_1 O(h_t^2) \end{aligned}$$

where c_1 is some constant. Recall that $\hat{J}_k^+ = (\hat{J}_k^T \hat{J}_k)^{-1} \hat{J}_k^T$. Assuming that $\|(\hat{J}_k^T \hat{J}_k)^{-1}\| \leq \frac{c_2}{\kappa}$, where κ is the smallest eigenvalue of $\hat{J}_k^T \hat{J}_k$, and $\|\hat{J}_k^T\| \leq \alpha$ then

$$\begin{aligned} \|h_{\theta_{k+1}}^*\| &\leq \frac{c_2\alpha}{\kappa} \left(\frac{\gamma}{2} \|h_{\theta_k}^*\| + \|E_k\| \right) \|h_{\theta_k}^*\| \\ &\quad + \frac{c_2\alpha}{\kappa} \|E_k\| \|\Delta\theta^*\| + c_3 \|h_t^2\| \end{aligned} \quad (17)$$

Assume that $\|h_{\theta_k}^*\|$ and $\|E_k\|$ are bounded such that,

$$\frac{c_2\alpha}{\kappa} \left(\frac{\gamma}{2} \|h_{\theta_k}^*\| + \|E_k\| \right) \leq \alpha \quad (18)$$

$$\frac{c_2\alpha}{\kappa} \|E_k\| \|\Delta\theta^*\| + c_1 \|h_t^2\| \leq \beta \quad (19)$$

for some $0 < \alpha < 1$ and $\beta > 0$. Then, requiring that the initial error $\|h_{\theta_0}^*\| \leq \frac{\beta}{1-\alpha}$, assures that the sequence $h_{\theta_k}^*$ is linearly convergent and bounded for all k .

As discussed in [13] and [12], since the bound on $\|E_k\|$ is based on λ , system and measurement noise, and the amount of variation in J^* , it is difficult to rigorously characterize $\|E_k\|$ such that (18) holds for $\alpha < 1$. However, it is plausible that for non-pathological tracking problems, an upper bound on $\|E_k\|$ exists, and thus the quasi-Newton method is convergent and stable. Experimental results verify this assumption, and the reader is referred to [6] for further verification.

D. A Comparison of Uncalibrated Static and Dynamic Controllers and Jacobian Estimators

A convergence analysis for the static controller can be done in a similar manner to Section II-C. However, the requirement that $\|E_k\|$ is bounded may not necessarily be met by a static Jacobian estimator. Simulations performed in MATLAB demonstrate the improved tracking and stability of the dynamic controller with dynamic Jacobian RLS estimation (Algorithm 1) over the static controller with static Jacobian RLS estimation (Algorithm 1 without the $\frac{\partial y_k^*(t)}{\partial t}$ terms) which exhibits stability problems for faster moving targets.

The experimental workcell (see Section III) is simulated using Corke's Robotics Toolbox and Machine Vision Toolbox [14]. The first three joints (two revolute and one prismatic) of an AdeptOne robot servo the toolpoint to follow a moving target with one feature point on each tracked by two cameras. The target moves circularly in the x - y plane and sinusoidally in the z direction. Simulations were run for 25 seconds with a 100 ms sampling time at various target speeds. Uniform ± 0.5 pixel noise was added to the data. Twenty-five simulations were performed at each speed for each controller, and the average RMS tracking error in pixels is plotted for both cameras in Figure 1. At slower speeds, the visual servoing problem is quasi-static and the two controllers behave similarly as expected. As the target motion increases, the static controller's performance degrades with the tracking lagging substantially and often becoming wildly divergent. Clearly, the static controller is unsuitable for tracking moving targets.

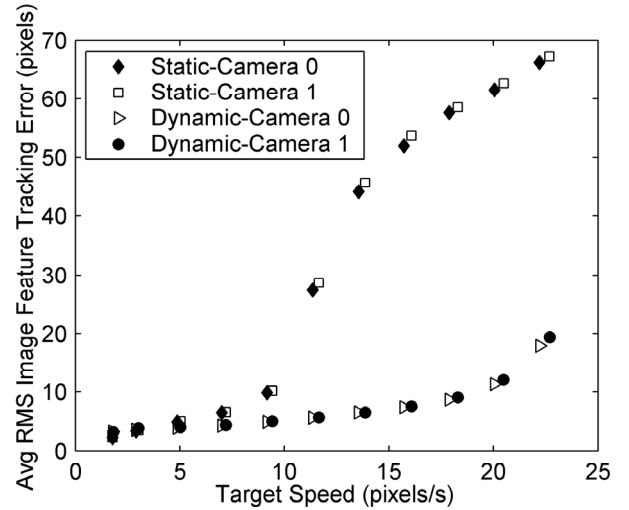


Fig. 1. A dynamic controller using Algorithm 1, and a static controller (Algorithm 1 without the $\frac{\partial y_k^*(t)}{\partial t}$ terms) are compared at increasing target speeds for a 3DOF tracking problem. The static controller causes the robot to lag behind the target motion and often goes unstable for higher target speeds. While tracking error increases at higher speeds, the dynamic controller remains stable.

III. EXPERIMENTAL SYSTEM

In [6] an initial verification of the dynamic quasi-Newton method and dynamic RLS Jacobian estimation is given in using a small reconfigurable RobixTM RCS-6 robot (in a 2-DOF planar configuration), a single Pulnix camera, a MuTech frame grabber and two personal computers. This system demonstrated stable and convergent tracking of a moving target.

The system used here employs stereo vision and the first three degrees-of-freedom on an AdeptOne robot. Image processing and pattern recognition is accomplished on a 400 MHz Pentium II running Windows98 utilizing the MVS-8000 software including the PatQuick tool from Cognex. The end effector is gripping a tool that holds a screw. The tool has been covered with a small pattern easily identifiable by the Cognex software. The target is comprised of a small paper pad with a pattern drawn on it moving on an industrial conveyor belt. The robot/vision system servos to the centroid of the pattern and tracks it on the belt.

Various hardware and software elements perform: image acquisition, image processing, algorithmic calculations, and the sending of control commands. During one control cycle, two images are captured; the second one is used to estimate the position of target and end effector at the end of the control cycle, because it is algorithmically important to sense the result of the most recent control signal before calculating the next update. Velocity commands are sent to the robot controller, creating a smooth motion.

The forgetting factor λ in (12) and (13) is used to tune the memory of the Jacobian estimation. A value of $\lambda = 0.5$ is used until convergence is achieved and then a value of $\lambda = 0.99$ is used. This creates a shorter memory while the system is learning the Jacobian and a longer memory during steady state tracking.

No model of target motion is used, the stereo vision system is completely uncalibrated, and none of the robot's kinematic parameters are used in the control algorithm. Due to hardware limitations, the update rate is a very slow 0.352s (2.8 Hz), yet the system is still able to exhibit convergence and steady state tracking of a moving target.

IV. EXPERIMENTAL RESULTS

The initial Jacobian is estimated using three initial moves by each of the three joints being controlled by the dynamic quasi-Newton method. The target is kept stationary initially due to a limited field of view. The target is then moved by a conveyor belt at 2.5cm/s which corresponds to an average motion of 26 pixels/s and 12 pixels/s for Cameras 0 and 1 respectively.

Figures 2-4 give experimental image feature data with two image captures per control cycle. Figure 2 shows the tracking error norm for each camera. The robot converges on the target smoothly, and the error converges to a nominal value. Convergence is achieved in 8 control cycles after which the tracking error is 8.6 pixels for Camera 0 and 10.0 pixels for Camera 1. Slight breaks in the tracking error indicate a failure of the vision system to identify target or robot features. In the event of insufficient data, no move is made.

Figure 3 and 4 show the image features for the end effector and the target point as seen by Cameras 0 and 1. Without known camera or robot models, the end-effector tracks the target point. The tracking does display a slight oscillatory nature. This could be the result of the memory inherent in the recursive least squares Jacobian estimation.

Figures 2-4 demonstrate that the algorithm is able to converge on and track a target moving on a conveyor belt. It is also interesting to examine the algorithm's response to an abrupt change in the target's motion. Figure 5 shows a detail of the RMS tracking error in pixels as the target moves back and forth in the camera's field of view. During this experiment, the target is moving at about 42 pixels/s and 25 pixels/s for Cameras 0 and 1 respectively. This corresponds to an average movement of about 15 and 9 pixels per control cycle for the respective cameras. The arrows in the figure denote the points where the target motion was changed by reversing the motion conveyor belt. As would be expected, the algorithm overshoots by 10-20 pixels, but regains tracking within roughly 3 control cycles.

The cameras are positioned such that one pixel corresponds to roughly to 1-2 mm. For the tracking of a steadily moving target shown in Figure 2, the average image errors are on the order of 10 or less pixels. This corresponds to a Cartesian error of 10-20 mm or less. There is a trade-off in selecting a camera position for this type of a system. If the camera is far away, an error of 5 pixels results in a greater error in Cartesian space than a 5 pixel error when the camera is closely positioned to the work area. However, the work volume becomes constrained as the field of view decreases.

V. CONCLUSIONS

This article has developed and demonstrated a dynamic quasi-Newton method for visual servo control of uncalibrated

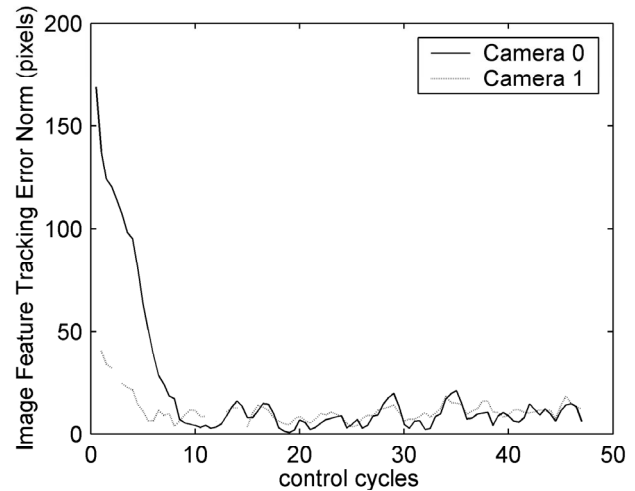


Fig. 2. Tracking error showing convergence after 8 control cycles (Two images are taken per control cycle.) After 8 control cycles, the tracking error is 8.6 pixels for Camera 0 and 10.0 pixels for Camera 1.

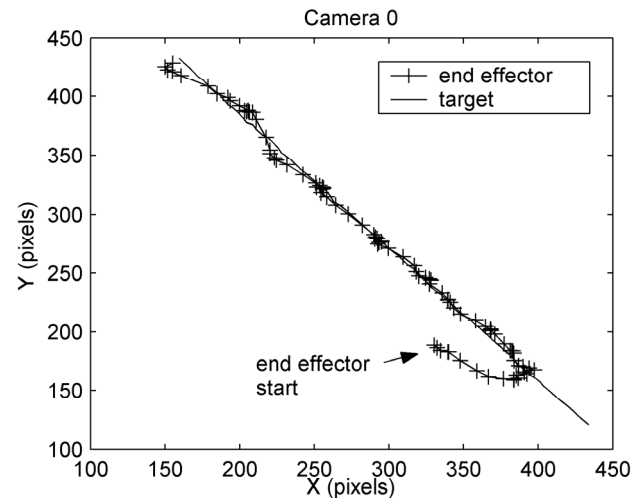


Fig. 3. The image features of the end effector and the target as seen by Camera 0 demonstrate the convergent tracking of a moving target. The average steady state tracking error is 10 pixels.

robotic systems with stationary imaging. Estimation of a Jacobian representing unknown imaging and robotic kinematic models is accomplished using a recursive least-squares algorithm.

While the methods in this work are closely related to the work in [1]- [5], the specific contributions are three-fold:

- 1) The moving target problem is explicitly addressed and algorithms are derived for this purpose. Simulations verify the inappropriateness of the static controller for the moving target problem.
- 2) A sound theoretical basis is established for uncalibrated vision-guided control algorithms.
- 3) The algorithms are experimentally verified, and model independent vision-guided robotic control and tracking are demonstrated within the bounds of hardware limitations.

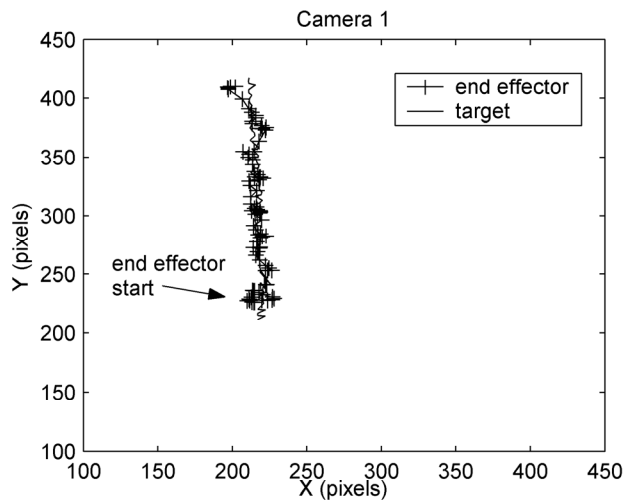


Fig. 4. The image features of the end effector and the target as seen by Camera 1 are shown. The average steady state tracking error is 8.6 pixels.

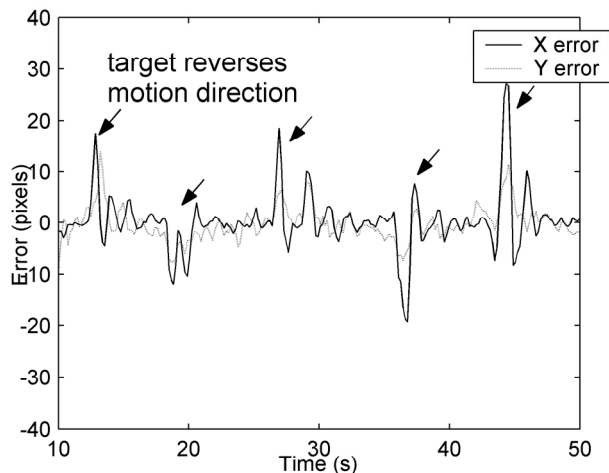


Fig. 5. Error in pixels for Camera 0 as robot tracks a target moving on a conveyor belt. The arrows denote the reversal of the conveyor belt motion. Note there is a 10-20 pixel error as the robot overshoots the target position. Tracking is regained within approximately three control cycles.

in *Proceedings of International Conference on Robotics and Automation*, Albuquerque, NM, April 1997, pp. 2874–80.

- [6] J. A. Piepmeyer, *A Dynamic Quasi-Newton Method for Model Independent Visual Servoing*, Ph.D. thesis, Georgia Institute of Technology, 1999.
- [7] J. A. Piepmeyer, G. V. McMurray, and H. Lipkin, “Tracking a moving target with model independent visual servoing: A predictive estimation approach,” in *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998, pp. 2652–2657.
- [8] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, New Jersey, 1983.
- [9] W. L. Brogan, *Modern Control Theory*, Prentice Hall, Englewood Cliffs, N.J., 1985.
- [10] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1991.
- [11] P. Eykhoff, *System Identification*, John Wiley and Sons, London, 1974.
- [12] L. Ljung L. Guo and P. Priouret, “Performance analysis of the forgetting factor rls algorithm,” *International Journal of Adaptive Control and Signal Processing*, vol. 7, pp. 525–537, 1993.
- [13] E. Eleftheriou and D. D. Falconer, “Tracking properties and steady-state performance of rls adaptive filter algorithms,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 5, pp. 1097–1109, October 1986.
- [14] P. I. Corke, “A robotics toolbox for MATLAB,” *IEEE Robotics and Automation Magazine*, vol. 3, no. 1, pp. 24–32, March 1996.

REFERENCES

- [1] K. Hosoda and M. Asada, “Versatile visual servoing without knowledge of true jacobian,” in *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, Munich, Germany, September 1994, pp. 186–193.
- [2] K. Hosoda, K. Igarashi, and M. Asada, “Adaptive hybrid control for visual servoing and force servoing in an unknown environment,” *IEEE Robotics and Automation Magazine*, vol. 5, no. 4, pp. 39–43, December 1998.
- [3] T. Tanaka M. Asada and K. Hosoda, “Visual tracking of unknown moving object by adaptive binocular visual servoing,” in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Taipei, Taiwan, August 1999, pp. 249–254.
- [4] M. Jagersand, “Visual servoing using trust region methods and estimation of the full coupled visual-motor Jacobian,” in *IASTED Applications of Robotics and Control*, 1996.
- [5] M. Jagersand, O. Fuentes, and R. Nelson, “Experimental evaluation of uncalibrated visual servoing for precision manipulation,”