2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Depth Invariant Visual Servoing

Peter A. Karasev, Miguel Moises Serrano, Patricio A. Vela, and Allen Tannenbaum

Abstract— **This paper studies the problem of achieving consistent performance for visual servoing. Given the nonlinearities introduced by the camera projection equations in monocular visual servoing systems, many control algorithms experience non-uniform performance bounds. The variable performance bounds arise from depth dependence in the error rates. In order to guarantee depth invariant performance bounds, the depth nonlinearity must be cancelled, however estimating distance along the optical axis is problematic when faced with an object with unknown geometry. By tracking a planar visual feature on a given target, and measuring the area of the planar feature, a distance invariant input to state stable visual servoing controller is derived. Two approaches are given for achieving the visual tracking. Both of these approaches avoid the need to maintain long-term tracks of individual feature points. Realistic image uncertainty is captured in experimental tests that control the camera motion in a 3D renderer using the observed image data for feedback.**

I. Introduction

This paper examines the geometry associated to planar, or nearly planar, object tracking in order to derive a depth-invariant visual servoing controller. Visual-servo control is a *closed-loop* visual control strategy that uses the image stream dynamics directly to perform a control task. While it is highly desirable to tightly integrate the visual processing and feedback control strategy, many robotic systems typically decouple the two steps.

Ignoring the feedback strategy and considering the visual processing loop only leads to *open-loop* visual tracking. Visual tracking is an estimation procedure that generates a spatial description of the target within the image. This description is typically either a point or a closed region in the image. Visual tracking is inherently nonlinear, and thus necessitates design procedures from nonlinear control when closing the loop. Common nonlinear strategies can be found in [12], [13]. Recently these nonlinear design techniques have been applied to adaptive camera calibration [10] and motion trajectory tracking [23].

Unlike open-loop vision-based estimation, the use of underlying computer vision techniques directly in closed-loop design is less well-understood. Typical systems in hardware consider the vision estimation algorithms as black-box measurement sources. Image formation, rigid body motion, and projective geometry are now well-understood [15], [18], as are the significant difficulties in computing solutions to the related *inverse* problems. A number of papers in the control field work towards the goal of linking control-theoretic design to an understanding of geometric image formation, with a particular focus on homography decomposition and depth estimation from correspondence of projected points [2], [5], [6], [8], [9], [14], [17]. Feedback terms used in these works require as input a stream of solutions to inverse problems in computer vision, such as reliable feature-correspondence and homography-decomposition for rigid pose estimation. Accurate, numerically stable geometric estimation requires nonlinear optimization and outlier-rejection schemes [15], [22].

Contribution. In [11], a depth-invariant visual servoing strategy was derived by assuming the existence of a variable focal length monocular camera. Augmenting the system dynamics to include the rate of change of the focal length for area stabilization neutralized the nonlinear effects of the camera projection equations. Inspired by this research, but seeking to remove the constraint of a variable focal-length system, we propose a time-varying visual servoing strategy that properly accounts for the depth nonlinearity to arrive at a depth-invariant feedback strategy. Furthermore, the time-varying gain is directly obtained through one of two known visual tracking strategies from the literature, segmentation-based [19] or template-based visual trackers [3], [20]. This approach contrasts with the existing literature which commonly utilizes tracked feature points or a tracked centroid. Often the feature points are associated to points on known 3D structure. Feature points may be unreliable in the long term (depending on the visual conditions). Segmentation-based or template-based methods rely on larger-scale imaging information, and are therefore more robust to image variation.

Organization. The remainder of this paper is organized as follows. In Section II, we describe the system and geometric quantities of interest for visual tracking. Control methodology is given in Section III, with the assumption that the geometric quantities of interest are measurable. Section IV describes the direct measurement of the main geometric quantity, the target area, required for depth-invariant visual servoing. The

978-1-61284-799-3/11/$26.00 ©2011 IEEE

resulting closed-loop system is tested in a 3D virtual environment simulation in Section V. Finally, Section VI concludes the paper.

II. Problem Formulation

The visual servoing task under consideration specifies that a target with unknown motion be image-centered when viewed by a moving monocular camera equipped with pan-tilt actuators. A pinhole camera model, is utilized for the image formation process. Due to the visual tracking algorithms utilized, it is presumed that a template image $\mathbb{I}_0$ of the target is available and that it corresponds to a planar region viewed head-on (e.g., with the plane's normal parallel to the optical axis). Planarity can be relaxed if the depth variation is minimal compared to the ratio between the focal length and the distance to the target. A control policy is sought such that future images, $\mathbb{I}(t)$, capture the target as close to the image center as possible.

The remainder of this section examines the geometry associated to the problem formulation and presents equations needed to derive the desired feedback strategy.

A. Rigid Camera Motion

A rigid body transformation can be represented by a rotation $R \in SO(3)$ and a Euclidean translation $T \in \mathbb{R}^3$. The set of all such configurations can be described in homogeneous matrix form as:

$$SE(3) \doteq \left\{ \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} | R \in SO(3), T \in \mathbb{R}^3 \right\} , \quad (1)$$

and is called the special Euclidean group in three dimensions. It is a subgroup of the four dimensional general linear group $GL(4)$. Let us denote a particular element in $SE(3)$ by g_C^W. The group $SE(3)$ also describes transformations of reference frames for points. The element g_C^W maps a homogeneous coordinate point q^C in the camera frame to q^W in the world frame. A camera-centric, or body, coordinate formulation of the dynamics using $g_W^C = (g_C^W)^{-1}$ is obtained from the relationship

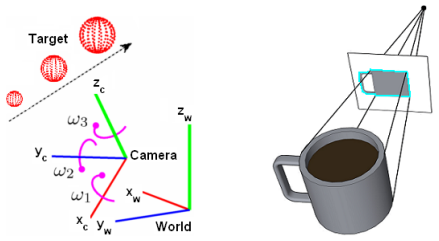
$$q^C = g_W^C q^W = \left(g_C^W\right)^{-1} q^W . \quad (2)$$

The time derivative of the transformation g_C^W satisfies

$$\left(g_C^W\right)^{-1} \dot{g}_C^W = \begin{pmatrix} \left(R_C^W\right)^T \dot{R}_C^W & -\left(R_C^W\right)^T \dot{T}_C^W \\ 0 & 0 \end{pmatrix} = \zeta_C^C(t) ,$$

where $\zeta_C^C(t)$ is the body velocity as it describes motion of the camera with respect to its own moving coordinate frame via a skew-symmetric block matrix $\hat{\omega}(t)$ and a translational vector $\nu(t)$, which is zero if the camera origin is fixed,

$$\hat{\omega} \doteq \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} , \ \zeta_C^C(t) \doteq \begin{pmatrix} \hat{\omega}(t) & \nu(t) \\ 0 & 0 \end{pmatrix} \quad (3)$$



(a) System (b) Measurement

Fig. 1. The system consists of a moving pan-tilt camera observing a moving target via a monocular, projective camera.

The time derivative of the observed point q^C is:

$$\begin{aligned} \dot{q}^C &= -\left(g_C^W\right)^{-1} \dot{g}_C^W \left(g_C^W\right)^{-1} q^W + \left(g_C^W\right)^{-1} \dot{q}^W \\ &= -\zeta_C^C(t) q^C + g_W^C \dot{q}^W \end{aligned} \quad (4)$$

where $\dot{q}^W$ is unknown target motion in the world frame creating input disturbance $(v_x^C, v_y^C, v_z^C, 0)^T = \left(g_C^W\right)^{-1} \dot{q}^W$. The matrix $\hat{\omega}$ satisfies $\dot{R}_C^W = R_C^W \hat{\omega}$, a nonlinear ordinary differential equation that describes the camera-frame angular motion as per Fig. 1(a).

B. Imaged Point Dynamics

The coordinates of a point $q^C(t)$ on the target's surface in 3D is not directly observable for a monocular camera. Measurements are generated by a projection and scaling [15], whereby depth information is lost. Setting $q^C = (x, y, z, 1)^T$ and using the model of a pinhole camera with fixed image size and isotropic scaling gives:

$$x_c = f\frac{x}{z}, \quad y_c = f\frac{y}{z}, \qquad \Pi(f, q^C) \doteq \begin{pmatrix} x_c \\ y_c \end{pmatrix} . \quad (5)$$

For a fixed focal length, f, the dynamics of the image point in the image coordinate system is

$$\begin{aligned} \frac{d}{dt}\Pi(q^C) &= \mathbf{D}\Pi(q^C)\dot{q^C} \\ &= \begin{pmatrix} \frac{f}{z} & 0 & -\frac{fx}{z^2} & 0 \\ 0 & \frac{f}{z} & -\frac{f}{z^2} & 0 \end{pmatrix} \left(-\zeta_C^C q^C + g_W^C \dot{q}^W\right) , \end{aligned} \quad (6)$$

This is a product of the measurement's *interaction matrix* [1] with a twist-induced motion. Assuming rotation-

only camera motion, the resulting image-coordinate dynamics are

$$\begin{pmatrix}\dot{x_c}\\ \dot{y_c}\end{pmatrix}=\begin{pmatrix} y_c\omega_3 - \omega_2 f + \frac{\omega_1 x_c y_c - \omega_2 x_c^2}{f} + \frac{f v_x^C - x_c v_z^C}{z} \\ -x_c\omega_3 + \omega_1 f - \frac{\omega_2 x_c y_c - \omega_1 y_c^2}{f} + \frac{f v_y^C - y_c v_z^C}{z}\end{pmatrix}. \tag{7}$$

C. Depth Dependent Area of a Planar Target.

Consider a target of finite extent that is planar, and whose normal to the plane aligns with the camera's optical axis (the $\hat{z}^C$-axis). The target, when imaged by the camera will have a depth varying area. For a template view of the target from a known distance z_0,

$$A_0 = \int_{\mathcal{S}_0} dx_c dy_c = \int_{\mathcal{A}} \frac{f^2}{z_0^2} dxdy = \frac{f^2}{z_0^2}\int_{\mathcal{A}} dxdy, \tag{8}$$

where $\mathcal{S}_0$ is the segmented region of the target, and $\mathcal{A}$ gives the collection of coplanar coordinates associated to the target in 3D. A later view of the same target at the distance $z(t)$ has the area

$$A(t) = \int_{\mathcal{S}(t)} dx_c dy_c = \int_{\mathcal{A}(t)} \frac{f^2}{z^2} dxdy = \frac{z_0^2}{z^2(t)} A_0. \tag{9}$$

Thus, the ratio of the two areas is equivalent to the ratio of the two depths squared,

$$\alpha(t) \doteq A(t)/A_0 = z_0^2/z^2(t). \tag{10}$$

Kowledge of the area ratio provides an inversely proportional estimate of the target depth. This information can be strategically used for visual servoing[1].

III. Depth-Invariant Visual Servoing

This section derives a Lyapunov-based controller for visual servoing. With ω_1 and ω_2 as control inputs and $\omega_3 = 0$,

$$\begin{aligned}\dot{x_c} &= -\omega_2 f + \frac{\omega_1 x_c y_c - \omega_2 x_c^2}{f} + \frac{f v_x^C - x_c v_z^C}{z} \\ \dot{y_c} &= \omega_1 f - \frac{\omega_2 x_c y_c - \omega_1 y_c^2}{f} + \frac{f v_y^C - y_c v_z^C}{z}\end{aligned}. \tag{11}$$

Proposition III.1 *Given the area ratio $\alpha(t)$ between the current imaged target and a template version viewed at a fixed distance is available, then the control strategy Eq. 12-Eq. 14:*

$$\omega_1(t) = -\frac{K(t) y_c}{f} \tag{12}$$

$$\omega_2(t) = \frac{K(t) x_c}{f} \tag{13}$$

$$K(t) = K_0 \alpha^{\frac{1}{2}}(t) \tag{14}$$

renders the system (11) GES under zero target velocity. As a consequence (11) is input-to-state stable when there is bounded velocity target motion. Furthermore, the region of stability is invariant to depth for known bounds on target velocities.

Proof: Consider the candidate Lyapunov function

$$V(x) = \frac{1}{2}(x_c^2 + y_c^2) \tag{15}$$

Using the previously computed $\dot{x}_c, \dot{y}_c$, the associated time-derivative for V is

$$\begin{aligned}\dot{V} &= x_c\dot{x}_c + y_c\dot{y}_c \\ &= x_c y_c\omega_3 - \omega_2 f x_c - x_c y_c\omega_3 + \omega_1 y_c f \\ &\quad + \frac{\omega_1 x_c^2 y_c - \omega_2 x_c^3}{f} + \frac{f x_c v_x^C - x_c^2 v_z^C}{z} \\ &\quad - \frac{\omega_2 x_c y_c^2 - \omega_1 y_c^3}{f} + \frac{f y_c v_y^C - y_c^2 v_z^C}{z}\end{aligned} \tag{16}$$

The control input terms ω_1 and ω_2 are the tilt and pan angular rates. The relative target motion in the camera frame v^C is unknown. Let $\omega_1 = -K y_c/f$ and $\omega_2 = K x_c/f$, where K is to be refined later. The time derivative becomes,

$$\begin{aligned}\dot{V} &= -K\left(x_c^2 + y_c^2\right) - Kf^{-2}\left(x_c^2 + y_c^2\right)^2 \\ &\quad + \frac{f}{z}\left(x_c v_x^C + y_c v_y^C\right) - \frac{v_z^C}{z}\left(x_c^2 + y_c^2\right)\end{aligned} \tag{17}$$

with a negative-definite part (the first two terms) and a disturbance part (the last two terms). When $v^C(t) = 0$, then

$$\dot{V} = -K\left(x_c^2 + y_c^2\right) - Kf^{-2}\left(x_c^2 + y_c^2\right)^2, \tag{18}$$

and the system (11) is clearly GES and Lipschitz. When $v^C(t) \neq 0$, then the system (11) is bounded and Lipschitz when z is bounded from below[2], f is bounded from above, and $v^C(t)$ is bounded and Lipschitz. Since these conditions are met and the image domain is finite, the closed-loop system is Lipschitz and GES when unforced by the disturbance. Under disturbances, it is ISS [21].

Note that in (17), the disturbance scales according to the distance of the target along the optical axis. This

[1]The focal length, f, cancels in the ratio. Thus the focal length of the camera does not need to be known precisely. In fact, all scale ambiguity is cancelled in this ratio. The ratio still holds if the plane is rotated and has non-trivial projection onto the image plane

[2]$z(t) > \underline{z} > 0$ holds when the target is in the camera's field of view.

simple visual servoing law will have variable performance depending on the target depth. The attracting invariant set radius will vary with z. It is clear that the invariant set radius can be made independent of z by simply scaling the gain by $z^{-1}(t)$, e.g., define $K(t) \doteq K_0 \alpha^{\frac{1}{2}}(t)$. The variable gain has a beneficial effect on the time-derivative of V. Since $\alpha^{\frac{1}{2}} = z_0/z(t)$,

$$\dot{V} = \frac{1}{z}\Big[-K_0 z_0 \left(x_c^2 + y_c^2\right) - K_0 z_0 f^{-2} \left(x_c^2 + y_c^2\right)^2 + f\left(x_c v_x^C + y_c v_y^C\right) - v_z^C \left(x_c^2 + y_c^2\right)\Big], \tag{19}$$

meaning that the depth is not a consideration when seeking to dominate the disturbance for estimating the invariant set radius. ■

Since the $\alpha(t)$ term is a function of the directly measured area, using it as a modifier on control gains does *not* entail the added complexity and potential convergence problems of an observer estimating $z(t)$.

IV. Area Rectification

Derivations in Section II-C assumed that the normal to the plane was parallel to the optical axis. When this is not the case, then the area of the target will be warped by the perspective transformation. As compared to the head-on view, pure rotation of the target relative to the template shrinks the apparent area in the image. If, however, the image template could be rectified so as to provide the head-on view, then the area computation would be correct. This section describes how to rectify the imaged object by inverting out the estimated rotation (or by applying the estimated translation). Compensation requires computation of the determinant of the transformation's Jacobian.

A. Effect of Target Rigid Body Displacement

From an initial camera view of the object (*template* image) to a new camera view of the object, four reference frames exist as per Figure Fig. 2. The first pair belongs to the camera and the object associated to the initial view, and the second pair describes the new frames of the camera and object configurations. The camera frames are denoted C_0 and C_1 while the object frames are denoted P_0 and P_1. Assume the initial camera view is such that $g_{P_0}^{C_0}$ consists of no rotation, $R_0 = \mathbb{1}$ and translation along the optical axis only, $d_0 = (0, 0, z_0)^T$, with the plane normal parallel to the optical axis. This view provides a head-on view of the planar target. All points on the planar target are described by the coordinates $q^{P_0} = (x, y, 0)^T$. Projections of points on

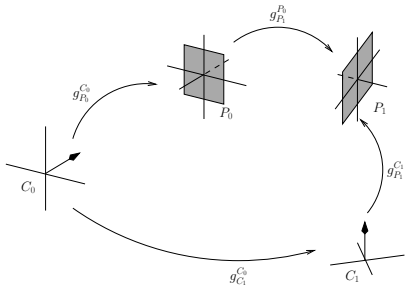


Fig. 2. Depiction of frames for template view and new view.

the planar target to image coordinates are

$$r^0 = \Pi\left(g_{P_0}^{C_0} q^{P_0}\right) = \gamma \begin{pmatrix} x \\ y \end{pmatrix} \tag{20}$$

$$r^1 = \Pi\left(g_{P_1}^{C_1} q^{P_1}\right) = \Pi\left(g_{P_1}^{C_1} g_{P_0}^{P_1} q^{P_0}\right) = \Pi\left(g_{P_0}^{C_1} q^{P_0}\right), \tag{21}$$

where γ is a scalar factor depending on f and z_0. The mapping taking the coordinates r^0 to r^1 has a nice structure, whose determinant provides clues regarding the true area. The following propostion is well known:

Proposition IV.1 *Under the presumption that the translation $T = (t_x, t_y, t_z)^T$ and rotation R define the element $g_{P_0}^{C_1}$, and points on the planar object are given by $(x, y, 0)$ in the object frame, the perspective transformation becomes*

$$P(x, y) = \left(f\frac{R_{11}x + R_{12}y + t_x}{R_{31}x + R_{32}y + t_z}, f\frac{R_{21}x + R_{22}y + t_y}{R_{31}x + R_{32}y + t_z}\right) \tag{22}$$

with associated Jacobian determinant

$$\det(DP) = f^2(R_{31}x + R_{32}y + t_z)^{-3} \cdot \Big((R_{11}R_{22} - R_{12}R_{21})t_z + (R_{12}R_{31} - R_{11}R_{32})t_y + (R_{21}R_{32}t_x - R_{22}R_{31})t_x\Big). \tag{23}$$

Derivation:: The template is a planar patch in 3D; all of the z-coordinates on the patch in the target frame are zero. In terms of the planar patch x and y coordinates, the visualized image coordinate are as per Equation (21). Expanding out the transformation of coordinates and the projection equation,

$$r^1 = \Pi\left(g_{P_0}^{C_1} q^{P_0}\right) = \Pi\left(\begin{pmatrix} R_{11}x + R_{12}y + t_x \\ R_{21}x + R_{22}y + t_y \\ R_{31}x + R_{32}y + t_z \\ 1 \end{pmatrix}\right) = P(x, y) \tag{24}$$

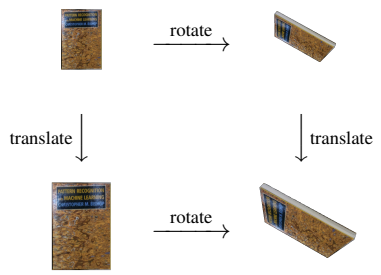


Fig. 3. Commutative diagram: template view is translated and rotated and projected to the 2D image plane.

The determinant naturally follows when computing the differential of P. Furthermore, for the template setup defined, the above perspective transformation and determinant also applies to the coordinate warp from r^0 to r^1; the translation will be scaled relative to that of $g_{P_0}^{C_1}$.

Area Integral Compensation:: Generation of the area ratio will require one of two potential image pairings, both of which are depicted in Fig. 3. There exists a transformation g whose perspective transform P maps from the template view (top left) to the current view (bottom right). One may consider the area ratio between the template and a translated version, or between the current segmentation and a rotated version of the template (area ratios occur between a lower element and the corresponding upper element in the diagram).

Let the image domain be given by Ω with the target region given by $\mathcal{S} \subset \Omega$, and let the head-on image of the target lead to Ω_0 and $\mathcal{S}_0$. Through a direct application of the change-of-variable theorem and the Jacobian (23), the area of the target in domain Ω_0 is

$$A(t) = \int_{\mathcal{S}} dx_c dy_c = \int_{\mathcal{S}_0} det(P(x_c, y_c) dx_c dy_c. \quad (25)$$

To obtain the area ratios, define P_T/P_R to be the perspective transformation arising from only applying the translation/rotation part of g. The measurement α results from either one of the following computations

$$\alpha(t) = \frac{\int_{\mathcal{S}} dx_c dy_c}{\int_{\mathcal{S}_0} det(P_R(x_c, y_c)) dx_c dy_c} \quad (26)$$

or

$$\alpha(t) = \frac{\int_{\mathcal{S}_0} det(P_T(x_c, y_c)) dx_c dy_c}{\int_{\mathcal{S}_0} dx_c dy_c}. \quad (27)$$

It is important to note that the camera focal length and the target distance in the template view need not be known precisely. The area ratio cancels out any scale ambiguity introduced by errors in both values. The controller derived in Section III need only use one of the ratios to achieve invariance with respect to target rotation. Which to use will depend on the estimation algorithm utilized, as each algorithm has differing sensitivities in R and T. The next two sections describe two different methods for estimating g, which is decomposed into $R = \exp(\hat{\omega})$ and T for improved numerics.

B. Segmentation-Based Estimation

One method to achieve tracking of an object is to extract the contour boundary of the target over time, known as *segmentation-based tracking*. Segmentations result in an *indicator function* $\phi : \Omega \to \mathbb{R}$ evaluating to unity for target points and zero for background points. Segmentation is achieved by minimizing a function encoding for the error in the current segmentation and for violation of any prior information. Once the segmentations are computed, the following optimization solves for the parametric warp $P : \Omega_0 \subset \mathbb{R}^2 \to \Omega \subset \mathbb{R}^2$ that aligns a template indicator function $\phi_0 : \Omega_0 \to \mathbb{R}$ (obtained from segmenting the template image) to the one currently observed $\phi_1 : \Omega_1 \to \mathbb{R}$ (obtained from segmenting the current image). Define the following cost function:

$$E(\hat{\omega}, \mathbf{T}) = \frac{1}{2} \int_{\Omega_1} \left(\phi_1 - \phi_0 \circ P^{-1}\right)^2 \, dudv \,, \quad (28)$$

which uses the inverse map since the forward map sends Ω_0 to Ω_1.

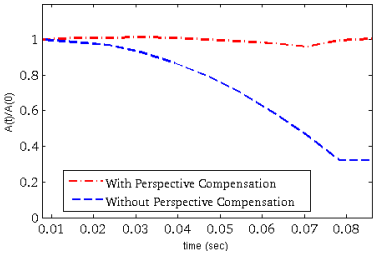
C. Template-Based Estimation

Template-based tracking utilizes the template image to generate a reference image patch $\mathbb{T} : \Omega_0 \to \mathbb{R}$ that must be warped under P to align with a patch within the new image $\mathbb{I}(t)$. The template patch $\mathbb{T}$ is defined over a (strict) sub-domain of the image domain, $\Omega_{\mathbb{T}} \subset \Omega_0$. The target in the current image is located by optimizing:

$$\min_{\omega, T} ||\mathbb{T} - \mathbb{I}(t) \circ P||^2 \,, \quad (29)$$

The objective function seeks to match the pixel intensities of the template to the pixel intensities of the image. Note that the implementation is achieved by comparing the template pixel intensities at the coordinates $(x_c, y_c)^T \subset \Omega_{\mathbb{T}}$ to the image pixel intensities at the perspective transformed coordinates $(x_c', y_c')^T = P(x_c, y_c)$. The collection of pixel intensities for both defines two matrices of pixel intensity values. The norm applied is the Frobenius norm of the difference between the two matrices (one of which is g-dependent).

(a) Pure rotation: images



(b) Rotation in-place: area ratio

Fig. 4. Under pure rotation of the target, direct computations of area fluctuate significantly. The compensated area integral keeps the ratio A/A_0 close to unity.

D. A Note on the Methods

The field of visual tracking in computer vision has a history of studying both template-based and segmentation-based tracking [24]. There are a large variety of tracking algorithms optimized for both of these techniques [4], [7], [16]. By casting the visual servoing framework to rely on these methods, we can leverage the extensive literature associated to robust, high performance tracking algorithms for them.

V. Simulation Results

This section first demonstrates the effect of compensating for area loss through incorporation of the determinant of the perspective transformation Jacobian, then demonstrates how the area compensating feedback control law improves closed-loop performance when faced with unknown target motion in the camera frame.

A simulation was carried out utilizing a planar region undergoing pure rotations only (constant translational displacement). The area of the target in the image plane was computed and compared to the area of the target as computed via (27). The compensated area remains fairly constant with some fluctuation due to estimation errors. The estimation errors arise from the pixel quantization effects associated to images, continuously changing measurements are not possible. The simulation provides confidence that the image-based estimates are sufficiently accurate for computation of the area integral.

To verify depth-invariance in closed-loop operation, the camera is placed at a fixed location in space while the target moves along a sinusoidal trajectory that approaches then recedes from the camera as depicted in Fig. 5. The tracking results are depicted in Fig. 6. Note that the standard visual servoing strategy has depth-varying performance while the proposed strategy has a consistent error bound. Snapshots of the simulated visual servoing task can be found in Fig. 7. The changing depth and orientation can be clearly seen. A planar ground surface is also depicted in the background to help visualize the camera pan and tilt movements during the visual servoing task.

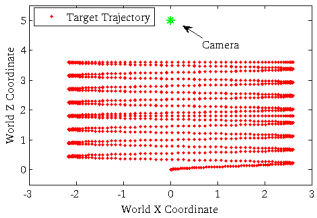


Fig. 5. Test Sequence: target approaches then recedes from camera.

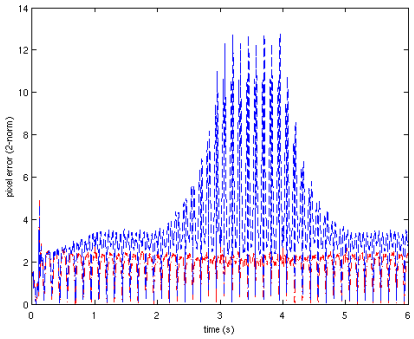


Fig. 6. Performance comparison: using the $K = \alpha^{1/2} K_0$ law (red) gives a range-independent bound on the image centroid errors for given velocity bounds. The tighter bound is particularly noticeable when the target approaches the camera.

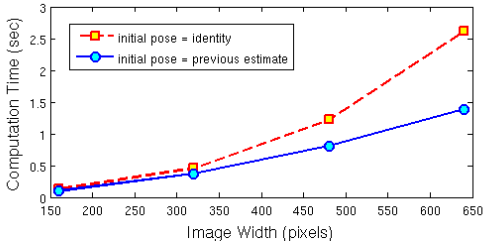


Fig. 8. Average execution time for region-based posed estimation, with identity and previous frame's estimate as initialization. Images used are all of $4:3$ aspect ratio, with widths $640, 480, 320$, and 160.

It is natural to ask whether the formulation of region-based pose estimation is necessarily too slow due to computational burden, despite the benefits in accuracy. Average execution time for representative image sizes is

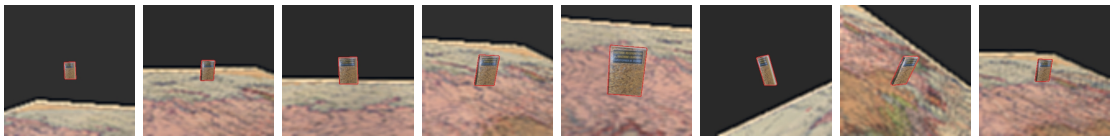


Fig. 7. Snapshots of target tracking. A video clip is available at http://www.youtube.com/watch?v=ssoixPyYaPQ.

shown in Fig. 8. During the iterative process, quadratic approximations $\hat{E}$ to the objective function $E(\omega, T)$ are periodically recomputed; performing the necessary evaluations of $E(\cdot)$ to create $\hat{E}$ in parallel dramatically speeds up the solver.

VI. Conclusions

We have presented a method for closed-loop visual tracking. The approach more directly links established visual tracking algorithms (segmentation-based and template-based) with nonlinear feedback control. The results use a simulation to verify the conclusions from a Lyapunov stability analysis. The system is robust to unmodelled target motion and camera frame disturbances while being invariant to depth.

A similar visual *regulation* strategy is possible, wherein a specific pose must be achieved relative to the target in spite of unknown target motion. Furthermore, the regulation can be performed using gradient descent quantities computed in the image plane and mapped back to the full three dimensional space. Converting the gradient equations into depth-invariant update laws will be required.

Acknowledgments

P. Karasev, M. Serrano, and P. Vela are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA. Allen Tannenbaum is with the School of Electrical and Computer Engineering, Boston University. This work was supported in part by grants from NSF *(ECS #0846750, #0625218)*, AFOSR, ARO, and an NSF Graduate Fellowship.

References

[1] F. Chaumette and S. Hutchinson. Visual servo control. ii. advanced approaches [tutorial]. *Robotics & Automation Magazine, IEEE*, 14(1):109–118, 2007.

[2] J. Chen, WE Dixon, M. Dawson, and M. McIntyre. Homography-based visual servo tracking control of a wheeled mobile robot. *IEEE Transactions on Robotics*, 22(2):406–415, 2006.

[3] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 564–575, 2003.

[4] D. Cremers, D. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. 72(2):195–215, 2007.

[5] A. Dani and W. Dixon. Single Camera Structure and Motion Estimation. *Visual Servoing via Advanced Numerical Methods*, pages 209–229, 2010.

[6] AP Dani, N. Gans, and WE Dixon. Position-based visual servo control of leader-follower formation using image-based relative pose and relative velocity estimation. In *American Control Conference, 2009. ACC'09.*, pages 5271–5276. IEEE, 2009.

[7] B. Deutsch, C. Graessl, Bajramovic F., and J. Denzler. *A Comparative Evaluation of Template and Histogram-Based 2D Tracking Algorithms*, pages 269–276. Springer, 2005.

[8] G. Hu, N. Gans, and W. Dixon. Quaternion-based visual servo control in the presence of camera calibration error. *International Journal of Robust and Nonlinear Control*, 20(5):489–503, 2010.

[9] G. Hu, N. Gans, N. Fitz-Coy, and W. Dixon. Adaptive homography-based visual servo tracking control via a quaternion formulation. *Control Systems Technology, IEEE Transactions on*, 18(1):128–135, 2009.

[10] A. Kapadia, D. Braganza, DM Dawson, and ML Mclntyre. Adaptive camera calibration with measurable position of fixed features. In *American Control Conference, 2008*, pages 3869–3874, 2008.

[11] P.A. Karasev, M.M. Serrano, P.A. Vela, and A. Tannenbaum. Visual closed-loop tracking with area stabilization. In *American Control Conference (ACC), 2010*, pages 6955–6961. IEEE, 2010.

[12] H.K. Khalil. *Nonlinear systems, Third Edition*. Prentice Hall, 2002.

[13] P. Kokotović and M. Arcak. Constructive nonlinear control: a historical perspective. *Automatica*, 37(5):637–662, 2001.

[14] D. Kumar and C. Jawahar. Robust homography-based control for camera positioning in piecewise planar environments. *Computer Vision, Graphics and Image Processing*, pages 906–918, 2006.

[15] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. An Invitation to 3D Vision: From Images to Models Springer Verlag, 2003.

[16] L. Matthews, T. Ishikawa, and S. Baker. The template update problem. 26(6):810–815, 2004.

[17] E. Montijano and C. Sagues. Fast pose estimation for visual navigation using homographies. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 2704–2709. IEEE, 2009.

[18] R.M. Murray, Z. Li, and S.S. Sastry. *A mathematical introduction to robotic manipulation*. CRC, 1994.

[19] M. Niethammer, A. Tannenbaum, and S. Angenent. Dynamic active contours for visual tracking. *Automatic Control, IEEE Transactions on*, 51(4):562–579, 2006.

[20] D. Schreiber. Robust template tracking with drift correction. *Pattern recognition letters*, 28(12):1483–1491, 2007.

[21] E. Sontag. Input to state stability: Basic concepts and results. *Nonlinear and Optimal Control Theory*, pages 163–220, 2008.

[22] R. Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 2(1):1–104, 2006.

[23] C. Wang, Z. Liang, J. Du, and S. Liang. Robust Stabilization of Nonholonomic Moving Robots with Uncalibrated Visual Parameters. In *American Control Conference, 2009. ACC'09.*, pages 1347–1352, 2009.

[24] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys (CSUR)*, 38(4), 2006.