

A “Robust” Convergent Visual Servoing System

A paper submitted to IROS '95

IEEE/RSJ International Conference on Intelligent Robots and Systems

D. Kim, A. A. Rizzi, G. D. Hager* and D. E. Koditschek

Artificial Intelligence Laboratory

Department of Electrical Engineering and Computer Science

The University of Michigan

Ann Arbor, MI 48109

Department of Computer Science*

Yale University

New Haven, CT 06520-8285

Abstract

This paper describes a simple visual servoing control algorithm capable of robustly positioning a three degree of freedom end effector based only on information from a stereo vision system. The proposed control algorithm does not require estimates of the gripper's spatial position, a significant source of calibration sensitivity. We show formally that the controller is immune to positional camera calibration errors. Additionally we demonstrate robustness to orientational miscalibration through a series of simulations and experiments.

1 Introduction

With the advent of affordable and computationally powerful computer vision hardware, it has become attractive to consider developing systems which make use of vision as their primary form of feedback. This trend towards “visual servoing” systems stands in contrast to the traditional notions of machine vision. In that tradition, a system uses vision as a means of statically planning future actions. Visual servoing systems make continuous use of visual information both to evaluate their moment to moment performance and to plan future action. Again, traditional vision systems typically measure features or target positions in a robot’s workspace in support of off-line motion planning. Inevitably the performance of such a system is highly dependent on the accuracy of the calibration of both the manipulator and camera subsystems, as the connection between them is open loop by design. In contrast, if vision is used both for the planning and execution of manipulator motion, the desired robot configuration can be described as a particular set of visual observations, and possibly, the need for exact calibration information may be relaxed.

Recently, a number of researchers have begun to report on such “visual servoing” systems [1]. In a previous paper, two of the present authors [2] have further classified these techniques into two sub-categories: one relying on a position-based approach and the other an image-based method. This distinction is based on whether the servo error is specified in the image planes of the camera systems or in the workspace of the manipulation system [3]. The methods described in this paper rely on stereo vision to produce image-based servo errors which in turn generate (via the “visual servoing” algorithm) velocity commands for what we will construe to be a point manipulator in space. The only task considered will be “point positioning”: moving the manipulator in such a manner that its point image coincides with a prespecified image target location.

The objective of this work is to explore the robustness issues surrounding such a visual servoing system. In particular, we wish to understand the effects of miscalibration of extrinsic camera parameters on the performance of visually based control strategies. In this paper, we present a simple control algorithm suitable for a three degree of freedom relative positioning problem – alignment of an end effector with an identified target in space. Since the proposed algorithm does not require an estimate of the spatial position of the end effector while carrying out such a motion, we can avoid performing algebraic triangulation from stereo image data, a significant source of calibration sensitivity. The methods presented here, based on modified gradient and Newton descent of visually measurable cost functions, accord particular attention to the descent structure in order to guarantee convergence while minimizing the calibration sensitivity. We believe that the work presented here provides a starting point for the design of robust controllers for the more generic problem of visually servoing the rigid transformation (six degrees of freedom) describing the gripper.

2 Background

A number of researchers have developed control systems capable of achieving goals expressed only as visual events. Predominantly this literature uses a Newton like method to minimize the measured visual error.

Espiau *et al.* [4] introduced interaction matrices for primitive visual features, a concept similar to the feature Jacobian introduced in [5], and used its generalized inverse for generating gripper velocity. Papanikolopoulos and Khosla [6] encoded a tracking task as minimizing the sum-of-squared difference optical flow. They used an adaptive mechanism to compensate for uncertainties in the model and to determine the depth related parameters in tracking an object. This algorithm requires an extra initialization process to align camera and target frames. Castano and Hutchinson [7] introduced a new hybrid vision/position control structure. They decompose the robot's task such that visual servoing is used only to control motion in the plane parallel to the camera's image plane, while errors in depth are controlled by a trajectory planner. The success of task decomposition depends strongly on the calibration of the camera.

More recently some researchers have introduced uncalibrated stereo cameras in their systems and provided successful working demonstrations [8, 9]. Hollinghurst and Cipolla [8] propose a position based servo algorithm using a static stereo vision system. Since they use an affine stereo algorithm to estimate positions of a robot and an object, it is valid only when the depth of an object is small compared to the viewing distance.

In the direct antecedent of this paper [9], Hager, *et al.* proposed a robust image based visual servoing technique, which made use of a static stereo camera system. The vision system was used to observe both the manipulator and target, forming the so-called 'endpoint-closed-loop' system. The control error for this system was defined as a visual "distance" from the gripper to a target (a similar problem was formulated in [10],[8]). The control system regulated this "visual error" by commanding velocities of a manipulator's end-effector. The particular control strategy chosen for this system required knowledge of the manipulator's Cartesian position - data expressly precluded by the desire to work solely in image plane coordinates as described above. Instead, the authors proposed a nonlinear observer that would estimate this data. Experiments demonstrate that the proposed control scheme is robust in the presence of relatively large calibration error, however the stability of the overall closed-loop system still remains unproven. We achieve the same result in this paper without requiring any state observer, and we are able to furnish a stability proof for the proposed algorithm as well.

3 A New Family of “Local” Visual Servos

This section offers a brief overview of the models associated with visual servo systems, then goes on to present a novel strategy based on insights gained from the structure of those models.

3.1 Stereo Camera Model

Consider two camera coordinate systems given by Σ_{c_1} , Σ_{c_2} and a world coordinate system Σ_w as illustrated in Fig 1. The stereo projective transformation, $g : \mathbf{R}^3 \rightarrow \mathbf{R}^4$, maps a spatial point, $\mathbf{r}$ to a pair of image plane points, θ . In general, this transformation can be written as [11]

$$\theta = \begin{pmatrix} \theta_{1,x} \\ \theta_{1,y} \\ \theta_{2,x} \\ \theta_{2,y} \end{pmatrix} = g(\mathbf{r}) = \begin{pmatrix} f_1 \frac{[\mathbf{r} - \mathbf{c}_1]^T \mathbf{x}_1}{[\mathbf{r} - \mathbf{c}_1]^T \mathbf{z}_1} \\ f_1 \frac{[\mathbf{r} - \mathbf{c}_1]^T \mathbf{y}_1}{[\mathbf{r} - \mathbf{c}_1]^T \mathbf{z}_1} \\ f_2 \frac{[\mathbf{r} - \mathbf{c}_2]^T \mathbf{x}_2}{[\mathbf{r} - \mathbf{c}_2]^T \mathbf{z}_2} \\ f_2 \frac{[\mathbf{r} - \mathbf{c}_2]^T \mathbf{y}_2}{[\mathbf{r} - \mathbf{c}_2]^T \mathbf{z}_2} \end{pmatrix} \quad (1)$$

where, $\mathbf{c}_i$ and f_i represent the location and the focal length of camera i ; $\mathbf{x}_i$, $\mathbf{y}_i$ and $\mathbf{z}_i$ are the orthogonal unit vectors defining the local coordinate system of camera i , with $\mathbf{z}_i$ directed along the optical axis of the camera. If we define $D_i := (\mathbf{r} - \mathbf{c}_i)^T \mathbf{z}_i$, then the Jacobian of g evaluated at $\mathbf{r}$, $Dg(\mathbf{r})$ can be written as

$$Dg(\mathbf{r}) = \begin{pmatrix} -f_1 [(\mathbf{r} - \mathbf{c}_1) \times \mathbf{y}_1]^T / D_1^2 \\ f_1 [(\mathbf{r} - \mathbf{c}_1) \times \mathbf{x}_1]^T / D_1^2 \\ -f_2 [(\mathbf{r} - \mathbf{c}_2) \times \mathbf{y}_2]^T / D_2^2 \\ f_2 [(\mathbf{r} - \mathbf{c}_2) \times \mathbf{x}_2]^T / D_2^2 \end{pmatrix} \quad (2)$$

where $\times$ denotes the vector cross product. Finally, we note that this Jacobian will lose rank when $(\mathbf{r} - \mathbf{c}_1) \times (\mathbf{r} - \mathbf{c}_2) = \mathbf{0}$, or equivalently, $\mathbf{r} \in \overline{\mathbf{c}_1 \mathbf{c}_2}$ or $\mathbf{c}_1 = \mathbf{c}_2$.

3.2 Problem Statement

Suppose a distinguished point on a robot gripper, $\mathbf{r}$, can be imaged by a stereo camera system as defined above, $\theta = g(\mathbf{r})$. Then the visual servoing task we wish to address is: given a desired image plane point, $\theta_d \in g(\mathbf{r}_d)$, and a robot controller capable of commanding arbitrary velocities in the Cartesian workspace,

$$\dot{\mathbf{r}} = \mathbf{u},$$

find a velocity control strategy, $\mathbf{u}(\theta, \theta_d)$ such that

$$g(\mathbf{r}) \rightarrow \theta_d.$$

As stated this problem is a three degree of freedom positioning task for the manipulator's gripper. If we define $e_\theta := \theta_d - \theta$ as the “visual” error, then the closed loop error dynamics for such a system becomes

$$e_\theta = g(r_d) - g(r) \quad (3)$$

$$\dot{e}_\theta = -Dg(r)u \quad (4)$$

3.3 Newton and Gradient Algorithms

Equation (4) takes a very standard form for which there are well understood solutions. In the sequel we will distinguish between Newton and gradient based variants of these solutions.

3.3.1 Gradient Method

Consider a control strategy of the form

$$u = [Dg(r)]^T e_\theta. \quad (5)$$

Since $\|e_\theta\|^2$ decreases along the motion of the resulting closed loop system, solutions of this form can result in overall system stability. Unfortunately, such gradient based descent techniques suffer from a number of well known shortcomings, the most relevant being:

Slow convergence: A property characteristic of gradient systems in general.

Calibration sensitivity: This follows from the dependence of $[Dg(r)]^T$ on r , which in practice must be computed as $g^\dagger(\theta)$ (i.e. requires triangulation), within the visual servoing assumption of no Cartesian data.

3.3.2 Newton Method

Similarly, a control strategy of the form

$$u = [Dg(r)]^\dagger e_\theta, \quad (6)$$

(here $M^\dagger$ denotes the generalized inverse of the non-square matrix M) can be used to generate stable motion. Moreover such an approach has the added benefit of producing uniform convergence, overcoming one of the drawbacks of the gradient method. Unfortunately, calibration sensitivity remains unaddressed as it is still necessary to compute r via triangulation of stereo image plane measurements.

3.4 Some Algebraic Identities

Before considering some alternative controller strategies we pause to review a number of algebraic properties associated with the perspective projection.

Begin by noting that equation (1) can equivalently be written in the form

$$\mathbf{A}(\boldsymbol{\theta})\mathbf{r} = \mathbf{b}(\boldsymbol{\theta}), \quad (7)$$

where

$$\mathbf{A}(\boldsymbol{\theta}) = \begin{pmatrix} [\theta_{1,x}z_1 - f_1x_1]^T \\ [\theta_{1,y}z_1 - f_1y_1]^T \\ [\theta_{2,x}z_2 - f_2x_2]^T \\ [\theta_{2,y}z_2 - f_2y_2]^T \end{pmatrix} \in \mathbf{R}^{4 \times 3}, \quad (8)$$

$$\mathbf{b}(\boldsymbol{\theta}) = \begin{pmatrix} c_1^T[\theta_{1,x}z_1 - f_1x_1] \\ c_1^T[\theta_{1,y}z_1 - f_1y_1] \\ c_2^T[\theta_{2,x}z_2 - f_2x_2] \\ c_2^T[\theta_{2,y}z_2 - f_2y_2] \end{pmatrix} \in \mathbf{R}^4. \quad (9)$$

From this it follows that $Dg(\mathbf{r})$ in (2) can be factored into state dependent and measurement dependent matrices (as shown in Appendix A):

$$Dg(\mathbf{r}) = -\Gamma(\mathbf{r})^{-1}\mathbf{A}(\boldsymbol{\theta}), \quad (10)$$

where

$$\Gamma(\mathbf{r}) = \begin{pmatrix} \mathbf{z}_1^T(\mathbf{r} - \mathbf{c}_1)\mathbf{I}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{z}_2^T(\mathbf{r} - \mathbf{c}_2)\mathbf{I}_2 \end{pmatrix} \in \mathbf{R}^{4 \times 4}$$

and $\mathbf{I}_n$ denotes the $n \times n$ identity matrix. Note that whenever $\mathbf{r}$ is visible from both cameras (in front of them), $\Gamma(\mathbf{r})$ is a positive definite matrix.

Additionally we can develop an algebraic relationship between work-space errors and visual errors. Given that $\mathbf{A}(\hat{\boldsymbol{\theta}})\hat{\mathbf{r}} = \mathbf{b}(\hat{\boldsymbol{\theta}})$, it follows that

$$\mathbf{A}(\hat{\boldsymbol{\theta}})[\hat{\mathbf{r}} - \mathbf{r}] = -\Gamma(\mathbf{r})[\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}] \quad (11)$$

$$\mathbf{A}(\boldsymbol{\theta})[\hat{\mathbf{r}} - \mathbf{r}] = -\Gamma(\hat{\mathbf{r}})[\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}] \quad (12)$$

as shown in Appendix A.

These properties of the projective transformation afford the design of simple controllers, as we now show.

3.5 “Local” Newton and Gradient Methods

In place of the “global” controllers presented above, we propose here what may be thought of as their “local” variants. These strategies have the property that the controllers use “fixed gain” laws based only on the desired set point.

The Newton type controller which results from this approach takes the form

$$\mathbf{u} = -\mathbf{A}^\dagger(\theta_d)\mathbf{K}\mathbf{e}_\theta. \quad (13)$$

While the corresponding gradient algorithm is given by

$$\mathbf{u} = -\mathbf{A}^T(\theta_d)\mathbf{K}\mathbf{e}_\theta, \quad (14)$$

In these equations $\mathbf{K} \in \mathbb{R}^{4 \times 4}$ is an arbitrary positive definite diagonal gain matrix. Note that both of these proposed control laws are independent of the current gripper position and that $\mathbf{A}(\theta_d)$ is a constant matrix for a given image plane target. Note that $\mathbf{A}(\cdot)$ depends upon neither c_1 nor c_2 , so that (13) and (14) are completely immune to baseline miscalibration.

4 “Global” Stability Analysis of the “Local” Newton and Gradient Variants

Consider, first, the case of perfect calibration information – that is the orientations of both cameras are known¹. We will show that the algorithms of Section 3.5 are asymptotically stable with a large domain of attraction. Thus, despite their reliance on only local measurement data, these “local” variants (13) and (14) have as large a domain of attraction as (5) and (6), assuming perfect calibration.

4.1 Proof of Stability

Making use of (10), the error dynamics of (4) can be rewritten as

$$\Gamma(\mathbf{r})\dot{\mathbf{e}}_\theta - \mathbf{A}(\theta)\mathbf{u} = \mathbf{0} \quad (15)$$

The stability of this closed-loop system, using the local newton control law, can now be examined by making use of

$$V(\mathbf{e}_\theta, \mathbf{r}) = \frac{1}{2}\mathbf{e}_\theta^T \Gamma(\mathbf{r})\Gamma(\mathbf{r})\mathbf{e}_\theta,$$

¹Note that since the algorithms in question are independent of camera position we need not be concerned with positional calibration information.

as a Lyapunov candidate function.

Taking the derivative of V along the systems trajectories yields

$$\begin{aligned}
\dot{V}(e_\theta, r) &= e_\theta^T \Gamma(r) (\dot{\Gamma}(r) e_\theta + \Gamma(r) \dot{e}_\theta) \\
&= e_\theta^T \Gamma(r) (D(Z\dot{r}) e_\theta + A(\theta) \dot{r}) \\
&= e_\theta^T \Gamma(r) A(\theta_d) u \\
&= -e_\theta^T \Gamma(r) A(\theta_d) A^\dagger(\theta_d) e_\theta \\
&= (r_d - r)^T A^T(\theta_d) A(\theta_d) A^\dagger(\theta_d) e_\theta
\end{aligned}$$

where $D(\cdot)$ denotes the diagonal matrix with its argument and Z as defined in (18).

If the orientations of the cameras are perfectly known – as has been presumed – it then follows that

$$\begin{aligned}
\dot{V}(e_\theta, r) &= (r_d - r)^T A^T(\theta_d) e_\theta \\
&= -e_\theta^T \Gamma(r) e_\theta \\
&\leq 0,
\end{aligned}$$

and we conclude that $e_\theta \rightarrow 0$.

Similarly for the gradient control law of Section 3.5, we begin by choosing as a candidate lyapunov function,

$$V(r_d, r) = \frac{1}{2} (r_d - r)^T (r_d - r).$$

This leads to the derivatives of V along the systems trajectory being

$$\begin{aligned}
\dot{V}(r_d, r) &= -(r_d - r)^T u \\
&= (r_d - r)^T A^T(\theta_d) e_\theta \\
&= -e_\theta^T \Gamma(r) e_\theta \\
&\leq 0.
\end{aligned}$$

Again we conclude that the closed loop system converges (regardless of any positional miscalibration of either camera), provided the gripper remains visible from both cameras throughout the motion.

4.2 The Need for Containment

Unfortunately the “global” results described in this section are conditioned by the fact that the end-effector of the robot must stay “in front” of both cameras during its motion. However, none of the algorithms presented have guaranteed this fact, and as a result, the exact meaning of the

conclusions must be carefully examined. Fortunately, the systems in question are all first order, and it immediately follows that conservative estimates for the domain of attraction for any of the algorithms can be derived directly from the Lyapunov functions used in the stability proofs. Specifically a maximal Lyapunov ball may be found which lies completely in front of both cameras, and it follows that any initial condition starting in that ball will remain in front of the cameras for all time, and, thus, converge to the desired goal.

We believe that with some attention to this problem of containment – choosing a control law such that the gripper does not leave either field of view – a class of visual servoing algorithms could be developed which respect the “obstacle” (avoiding end effector motions that drift “behind” a camera). In such a case, the meaning of “global” in this domain could be extended to include all initial conditions which start in front of both cameras. Of course, we have not yet studied this problem carefully, and have no such result at present.

It is worth noting that we can show a direct correspondence between the proposed local (fixed gain) controllers and the traditional Newton (essentially gain scheduled) method by recourse to a time-varying change of coordinates (see Appendix B).

5 Robustness Experimentation and Simulation

The following section offers a short set of experimental results which demonstrate the notably better robustness properties of the “new” local control strategies, followed by a more thorough evaluation of the robustness properties of these controllers based on a series of simulations.

5.1 Experimental Data

The positioning task with respect to a target point in space was implemented on testbed, consisting of a Zebra robot arm and two Sony XC-77 cameras with 12.5 mm lens and two pan-tilt heads. The cameras are placed approximately 90 cm from the robot along the y axis and 30 cm apart along x axis. They are oriented to point back along the y axis of robot. A point target was chosen roughly 90 cm from both cameras.

We test the proposed algorithm with tuned and detuned parameters by placing a gripper with different offset positions from the target (see Table 1). The detuned system is obtained by rotating the camera1 12 deg inward and camera2 8 degree inward. The entire visual control system runs at a rate of 18.5 Hz. We are forced to limit robot velocities to a maximum of 5 cm/sec due to inherent limitation of robot control system.

Experiment	Offset from target
Exp1	+10 cm in Y
Exp2	+10 cm in Z
Exp3	10 cm in X, +2 cm in Z
Exp4	-10 cm in X , +2 cm in Z
Exp5	-6 cm in X, +6 cmin Y, +6cm in Z

Table 1: Initial gripper Positions

The experimental results are shown in Fig 5. Each set shows the trajectories of the gripper performing given task using tuned parameters (solid line) and detuned parameters (dashed line).

5.2 Simulation

A large sequence of computer simulations has been performed to evaluate the relative performance of the controllers of Section 3.5 against the more traditional “Newton method” in the presence of camera calibration error. As was noted in the previous section, the proposed controller is immune to displacement errors, and thus we will focus only on situations involving rotational miscalibration.

The simulation setup consists of a pair of cameras whose optical axis are parallel and perpendicular to the base line. The separation between the cameras is, and 100 cm. The “task” we wish to perform is to position the simulated robot gripper at a point between the two cameras, and 250 cm from baseline.

A complete study of the domains of convergence for every possible type and magnitude of miscalibration would be near impossible to present due to the sheer volume of data required. Thus we choose to present a systematic set of miscalibration situations which covers a fairly significant region of the space of possible miscalibrations. In particular we will consider situations where the calibration data represents cameras which are rotated from the actual cameras about one of the principle axis, and we will consider these in a pair-wise fashion (i.e. camera one is rotated up 5 degrees and camera two is rotated in 5 degrees and so on). Of the possible 36 such configurations (two possible direction of rotation and 3 degrees of freedom) we will eliminate the symmetric cases to limit ourselves to an evaluation of only 21 “types” of miscalibration, each of which is evaluated over the range of 0 to 30 degrees in increments of 2 degrees.

For each such case, 1000 different initial gripper positions were uniformly placed within a range of 100 cm from a desired target position, and the percentage of trajectories which resulted in convergence were measured for both the newton algorithm and the proposed local variants (13) of Section

3.5. A trajectory was classified as converging if it met the following criteria:

1. The position of gripper remains visible to both cameras throughout the trajectory.
2. The final position arrives within 1 cm of the desired target within 90 sec.

The overall gains were comparable² in both control algorithms.

Figure 2 demonstrates that the Newton method begins to fail when the magnitude of miscalibration approaches 6 degrees. In contrast, our newly proposed method performs successfully up to 10 degrees of miscalibration. Although not explicitly documented in Fig 2, we observed that the drastic increase in failure rates between 10 to 20 degrees were the result of total failure for the case when both of the actual cameras were pointed inward. Not surprisingly both methods demonstrated significant robustness with respect to rotations around the optical axis.

Looking at the particular performance of both control algorithms for a particular motion allows some insight in to the differences between the algorithms. Figure 4 show the motion of the gripper from an initial position of (50,-100,0) to the desired position of (-50, -50, 100) when the estimate of one camera is rotated ($y_1, 15^\circ$) and the estimate of the second camera is rotate ($x_2, 15^\circ$) from the actual cameras.

6 Conclusion

We present a simple visual servo control algorithm for a 3 d.o.f positioning of gripper using stereo vision. The proposed controller does not resort to a dynamic estimator or inverse transformation in commanding gripper motion, distiguishing it from all previous approaches. We have shown that the proposed control algorithm is globally stable regardless of the positional calibration error of cameras. Even though we cannot offer a formal characterization of its robustness to rotational modelling errors, preliminary experiments and exhaustive simulation results show it is robust in the presence of significant miscalibration.

Futher research will be directed toward the case of rigid transformations (the six d.o.f problem).

References

- [1] P. Corke, "Visual control of robot manipulators - a riview," in *Visual Servoing*, K. Hashimoto, editor, pp. 1 –32, World Scientific, 1994.

²We have taken this to mean that the local dynamic behavior of both algorithms in the vicinity of the target location are similar.

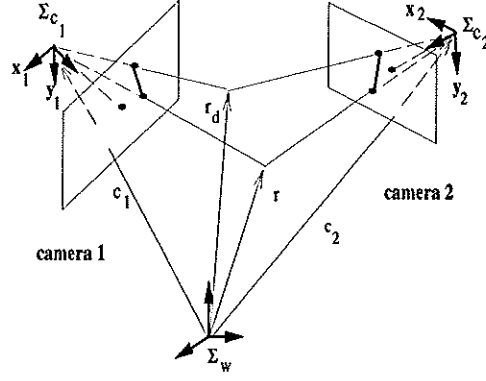
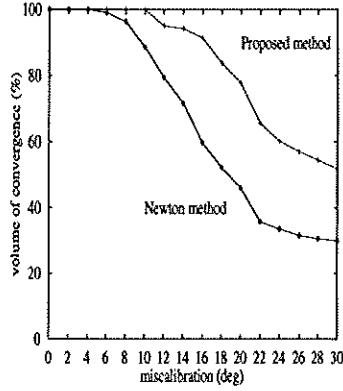
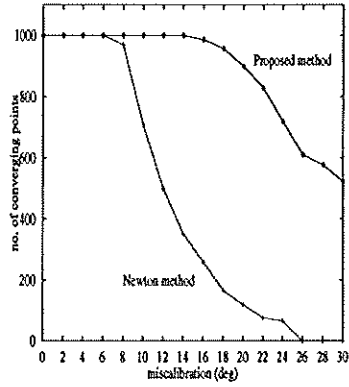


Figure 1: Stereo Camera Model

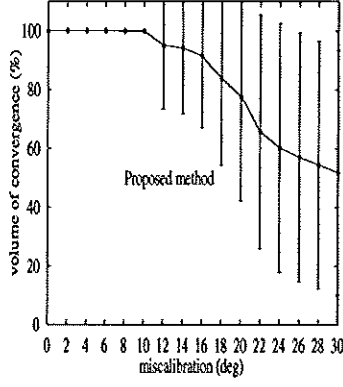


(a)

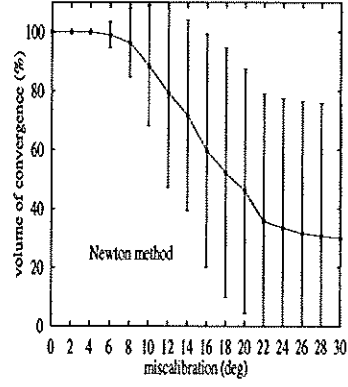


(b)

Figure 2: (a) Average volume of convergence counting all 21 types of miscalibration, (b) Number of converging points in a specific type of miscalibration (estimated camera1 is pointing inward and estimated camera2 is pointing downward)

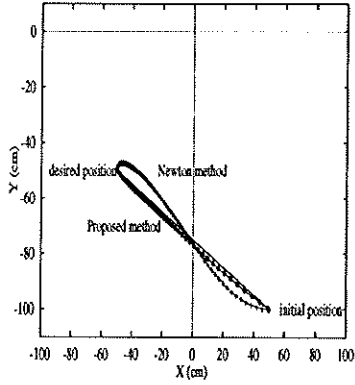


(a)

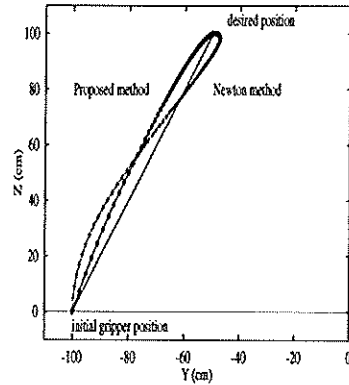


(b)

Figure 3: Average volume of convergence counting all 21 types of miscalibration and standard deviation: (a):Proposed method (b):Newton method

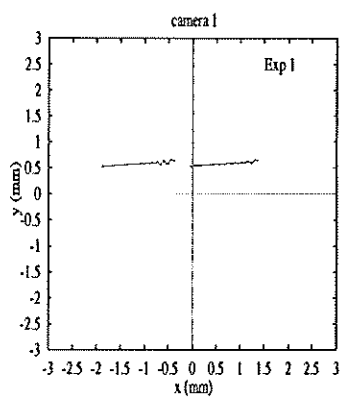


(a)

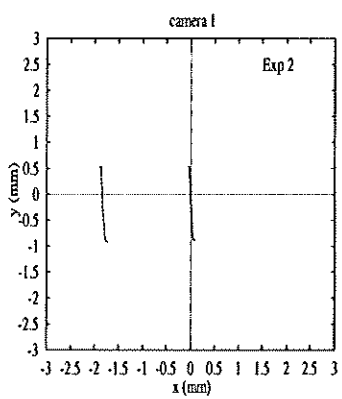
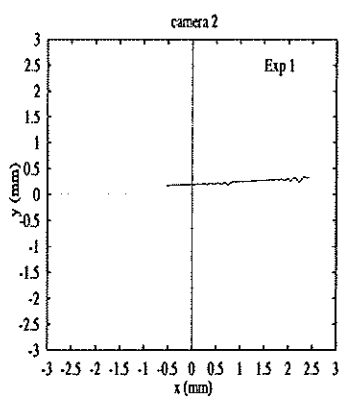


(b)

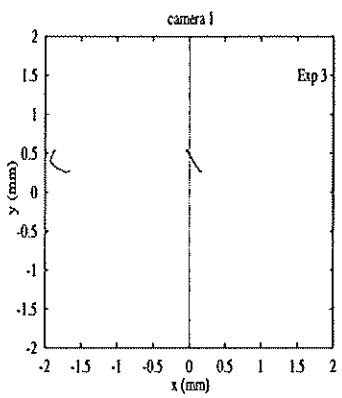
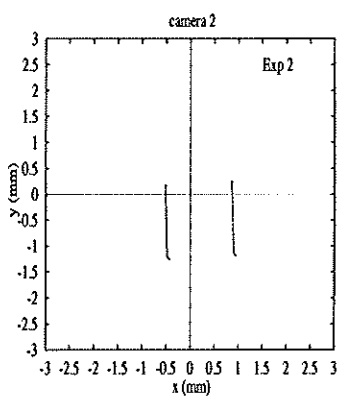
Figure 4: Trajectories of gripper in Cartesian space when estimated cameras are rotated inward and upward by 15 deg each : (a) X-Y projection, (b) Y-Z projection



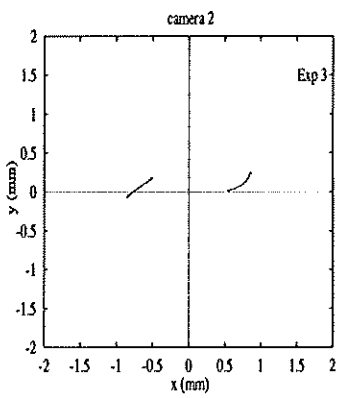
Exp1: +10 cm in Y

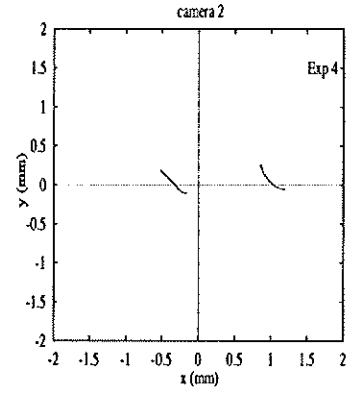
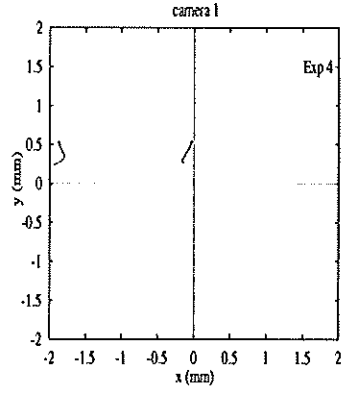


Exp2: +10 cm in Z

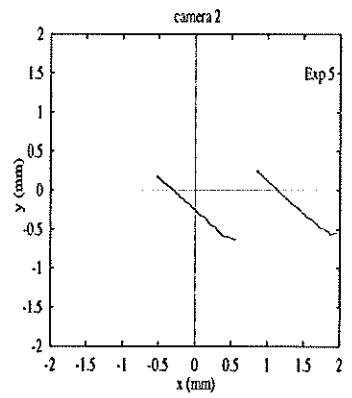
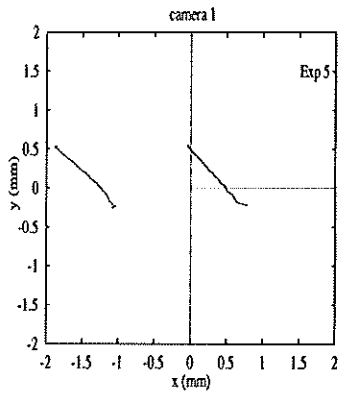


Exp3: +10 cm in X + 2 cm in Z





Exp4: -10 cm in X + 2 cm in Z



Exp5: -6 cm in X , 6 cm in Y +6 cm in Z

Figure 5: Trajectories viewed from stereo cameras of relative positioning from 5 different offsets (solid line shows trajectory with tuned system and dashed line shows the trajectory of gripper after rotating cameras)

- [2] A. A. Rizzi and D. E. Koditschek, "An active visual estimator for dexterous manipulation," *IEEE Transactions on Robotics and Automation*, vol. (submitted), , 1994.
- [3] L. E. Weiss, A. C. Sanderson, and C. P. Newman, "Dynamic sensor-based control of robots with visual feedback," *IEEE Trans. on Robotics and Automation*, vol. RA-3, No. 5, pp. 404 – 417, Oct 1987.
- [4] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. on Robotics and Automation*, vol. 8, No. 3, pp. 313 – 326, June 1992.
- [5] K. Hashimoto and H. Kimura, "Lq optimal and nonlinear approaches to visual servoing," in *Visual Servoing*, K. Hashimoto, editor, pp. 165 –198, World Scientific, 1994.
- [6] N. P. Papankolopoulos and P. K. Khosla, "Adaptive robotic visual tracking: Theory and experiments," *IEEE Trans. on Automatic Control*, vol. 38, No. 3, pp. 1 –32, Mar 1993.
- [7] A. Castano and S. Hutchinson, "Visual compliance: Task-directed visual servo control," *IEEE Trans. on Robotics and Automation*, vol. 10, No. 3, pp. 334 – 342, June 1994.
- [8] N. Holinghurst and R. Cipolla, "Uncalibrated stereo hand-eye coordination," Technical report, Cambridge University, Dept. of Engineering, September 1993.
- [9] G. G. Hager, W. Chang, and A. S. Morse, "Robot feedback control based on stereo vision: Towards calibration free hand-eye coordination," in *Proc. IEEE int'l Conf. Robotics and Automation*, pp. 2850 – 2856, June 1994.
- [10] S. Wijesoma, D. Wolfe, and R. Richard, "Eye-to-hand coordination for vision-guided robot control applications," *Int. J of Robotics Res*, vol. 12, No. 1, pp. 404 – 417, Feb 1993.
- [11] B. K. Horn, *Robot Vision*, MIT Press, Cambridge, 1986.
- [12] J. Munkres, *Analysis on Manifolds*, Addison-Wesley, Advanced Book Program, 1990.

A Proof of Algebraic Properties

1. $J(r) = -\Gamma(r)^{-1}A(\theta)$

Denote $F(r, \theta) = A(\theta)r - b(\theta)$. Using the Implicit Function Theorem [12], the system Jacobian $Dg(r)$ can be written as

$$Dg(r) = - \left(\frac{\partial F(r, \theta)}{\partial \theta} \right)^{-1} \frac{\partial F(r, \theta)}{\partial r}.$$

Since

$$\begin{aligned}\frac{\partial F(\mathbf{r}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial \mathbf{A}(\boldsymbol{\theta}) \mathbf{r}}{\partial \boldsymbol{\theta}} - \frac{\partial \mathbf{b}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \\ &= \mathbf{D} \begin{pmatrix} \mathbf{z}_1^T (\mathbf{r} - \mathbf{c}_1) \\ \mathbf{z}_1^T (\mathbf{r} - \mathbf{c}_1) \\ \mathbf{z}_2^T (\mathbf{r} - \mathbf{c}_2) \\ \mathbf{z}_2^T (\mathbf{r} - \mathbf{c}_2) \end{pmatrix} = \Gamma(\mathbf{r}) \\ \frac{\partial F(\mathbf{r}, \boldsymbol{\theta})}{\partial \mathbf{r}} &= \mathbf{A}(\boldsymbol{\theta})\end{aligned}$$

where $\mathbf{D}(\cdot)$ denotes the diagonal matrix, we have

$$\mathbf{J}(\mathbf{r}) = -\Gamma(\mathbf{r})^{-1} \mathbf{A}(\boldsymbol{\theta}),$$

or

$$\mathbf{A}(\boldsymbol{\theta}) = -\Gamma(\mathbf{r}) \mathbf{D}g(\mathbf{r}).$$

$$2. \mathbf{A}(\hat{\boldsymbol{\theta}})(\hat{\mathbf{r}} - \mathbf{r}) = -\Gamma(\mathbf{r})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})$$

From equation (8) and (9) $\mathbf{A}(\hat{\boldsymbol{\theta}})$ and $\mathbf{b}(\hat{\boldsymbol{\theta}})$ can be written as

$$\mathbf{A}(\hat{\boldsymbol{\theta}}) = \mathbf{A}(\boldsymbol{\theta}) + \mathbf{D}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \mathbf{Z} \quad (16)$$

$$\mathbf{b}(\hat{\boldsymbol{\theta}}) = \mathbf{b}(\boldsymbol{\theta}) + \mathbf{D} \begin{pmatrix} \mathbf{c}_1^T \mathbf{z}_1 \\ \mathbf{c}_1^T \mathbf{z}_1 \\ \mathbf{c}_2^T \mathbf{z}_2 \\ \mathbf{c}_2^T \mathbf{z}_2 \end{pmatrix} (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \quad (17)$$

where

$$\mathbf{Z} = \begin{pmatrix} \mathbf{z}_1^T \\ \mathbf{z}_1^T \\ \mathbf{z}_2^T \\ \mathbf{z}_2^T \end{pmatrix} \in \mathbf{R}^{4 \times 3}. \quad (18)$$

$$\begin{aligned}\mathbf{A}(\hat{\boldsymbol{\theta}})(\hat{\mathbf{r}} - \mathbf{r}) &= \mathbf{A}(\hat{\boldsymbol{\theta}})\hat{\mathbf{r}} - \mathbf{A}(\hat{\boldsymbol{\theta}})\mathbf{r} \\ &= \mathbf{A}(\hat{\boldsymbol{\theta}})\hat{\mathbf{r}} - [\mathbf{A}(\boldsymbol{\theta}) + \mathbf{D}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})\mathbf{Z}]\mathbf{r} \\ &= \mathbf{b}(\hat{\boldsymbol{\theta}}) - \mathbf{b}(\boldsymbol{\theta}) - \mathbf{D}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})\mathbf{Z}\mathbf{r} \\ &= -\Gamma(\mathbf{r})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})\end{aligned}$$

3. $A(\theta)(\hat{r} - r) = -\Gamma(\hat{r})(\hat{\theta} - \theta)$ Similarly,

$$\begin{aligned}
A(\theta)(\hat{r} - r) &= A(\theta)\hat{r} - A(\theta)r \\
&= [A(\hat{\theta}) - D(\hat{\theta} - \theta)Z]\hat{r} - A(\theta)r \\
&= b(\hat{\theta}) - b(\theta) - D(\hat{\theta} - \theta)Z\hat{r} \\
&= -\Gamma(\hat{r})(\hat{\theta} - \theta)
\end{aligned}$$

B A Change of Coordinates

Begin by define a new variable $\tilde{\theta}$ to be a transformation of the original error e_θ through a time-varying positive definite matrix $\Gamma(r)$:

$$\tilde{\theta} \triangleq \Gamma(r)e_\theta.$$

Since

$$\begin{aligned}
\dot{\tilde{\theta}} &= \dot{\Gamma}(r)e_\theta + \Gamma(r)\dot{e}_\theta \\
&= D(Z\dot{r})\Gamma^{-1}(r)\tilde{\theta} + \Gamma(r)\dot{e}_\theta,
\end{aligned}$$

$\dot{e}_\theta$ becomes

$$\dot{e}_\theta = \Gamma^{-1}(r) \left[\dot{\tilde{\theta}} - D(Z\dot{r})\Gamma^{-1}(r)\tilde{\theta} \right].$$

This gives rise to a new error system defined by

$$\dot{\tilde{\theta}} - D(Z\dot{r})\Gamma^{-1}(r)\tilde{\theta} - A(\theta)\dot{r} = 0.$$

By making use of (16), we can further simplify this error system to be

$$\begin{aligned}
&\dot{\tilde{\theta}} - D(Z\dot{r})\Gamma^{-1}(r)\tilde{\theta} - A(\theta_d)\dot{r} + D(e_\theta)Z\dot{r} \\
&= \dot{\tilde{\theta}} - A(\theta_d)\dot{r} - D(Z\dot{r})\Gamma^{-1}(r)\tilde{\theta} + D(Z\dot{r})\Gamma^{-1}(r)\tilde{\theta} \\
&= \dot{\tilde{\theta}} - A(\theta_d)\dot{r} \\
&= 0
\end{aligned}$$

Thus, we can write the overall error dynamic as

$$\dot{\tilde{\theta}} = A(\theta_d)u. \tag{19}$$

Equation (19) so could take $u = A^\dagger(\theta_d)\tilde{\theta}$.