An Image-Based Stereo Visual Servoing Algorithm Robust to the Camera Extrinsic Parameters

카메라 외적 파라메터에 대하여 강인성을 갖는 스테레오 시각 제어 알고리즘

Dong Min Kim
(김 동 민)

요 약 : 본 논문은 카메라 파라메터의 측정오차에 대하여 강인성을 보이는 새로운 로봇의 스테레오 시각 위치제어 알고리즘을 제시한다. 제시된 알고리즘은 카메라로부터 측정된 영상 데이터만을 이용함으로써, 특히 파라메터 측정오차에 대하여 매우 민감함을 보이는 영상 데이터로부터 작업 공간에서의 위치로의 변환, 즉 역변환 추정장치의 필요성을 제거하였다. 이러한 특징이 기존 개발된 시각 제어기와의 큰 차이를 두고 있다. 그럼에도 불구하고 제시된 제어기는 전 작업 영역 내에서 시스템 안정성을 갖는다. 또한 카메라의 위치 측정 오차에 대하여 전혀 영향을 받지 않음이 증명되어지고 방향 측정 오류에 대해서도 기존 제어기보다 강인함을 시뮬레이션을 통하여 보여진다.

Keywords : stereo camera, visual servoing system, robustness, relative positioning

I. Introduction

With the advances in computer vision hardware it has become attractive to consider developing robot-vision system which makes use of vision as their primary form of feedback. This trends towards visual servoing systems stands in contrast to the traditional notions of machine vision. In that tradition, vision system typically measures features or target positions in robot's work-space in support of off-line robot motion planning. 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. The parameters of subsystems may be obtained by high accuracy calibration processes. However, since calibration process is designed using idealized models and is performed in special regimes of operation, small calibration errors in parameters are inevitable. These small errors distort the geometry of the camera and manipulator transformation so that in practice it may degrade the performance and it may even cause instability. In contrast, if vision system is used both for planning and execution of manipulator motion, the desired configuration is described as a particular set of visual observations, and possibly, the need for exact calibration information may be relaxed. This trend towards "visual servoing" systems stands in contrast to the traditional vision system for statically planning future actions of robot. Corke presents an historical overview of the field of visual servoing for manipulation[1].

The control objective of the visual servoing system is to control manipulator's gripper motion in order to place the image plane coordinates of features on the target at some desired position. The desired image coordinates could be constant or changing with time. The control strategy used to achieve the control objective is based on the minimization of an objective function at each instant of time. Predominantly literatures use a Newton like method to minimize the measured visual error. In [2] Espiau *et al.* introduced interaction matrices for primitive visual features, similar concept to the feature Jacobian introduced in [3], and used its generalized inverse for generating gripper velocity. In their approach the feature Jacobian is assumed to be known. It should be evaluated at the relative position of an object with respect to the camera using inverse projection, significant source of calibration sensitivity. Papanikolopoulos and Khosla encoded a tracking task as a minimizing the sum-of-squared difference optical flow[4]. They used an adaptive mechanism to compensate for uncertainties in the model and to determine the depth related parameters in tracking an object. the major drawback of this algorithm is to require an extra initialization process to align camera and target frames. Castano and Hutchinson introduced a new hybrid vision/position control structure[5]. They decomposed the 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 camera. More recently some researchers have introduced uncalibrated stereo cameras in their systems and provided successful working demonstrations[6][7]. In [6] Hollinghurst and Cipolla proposed a position based

접수일자 : 1998. 4. 14., 수정완료 : 1998. 8. 7.
김동민 : 홍익대학교 과학기술대학 전기전자컴퓨터 공학부

servoing algorithm based on the weak perspective model of stereo cameras. It is valid to estimate position of a robot and an object only when the depth of an object is small compared with the viewing distance. Hager *et al.* introduced image based servoing algorithm using a nonlinear position estimator to remove the inverse transformation for the manipulator's Cartesian position[7]. Experiments demonstrate that the proposed control algorithm is robust to the relatively large calibration error, however the stability of overall system remains unproven.

The objective of this work is to explore the robustness issues surrounding such a visual servoing system using fixed stereo cameras external to the manipulator. In particular, we wish to understand the effects of the extrinsic camera parameter calibration errors on the performance of visually based control strategies.

This paper is organized as follows. In section II, we briefly review a stereo camera model and develop useful identities regarding camera perspective transformation and its Jacobian. In section III, we present a formal statement of the problem of three degree of freedom relative positioning of a manipulator's gripper and review traditional approaches to the present problem setting. In section IV, we present simple and robust control algorithms based on modified gradient and Newton method, and the stability properties of the presented controllers are discussed. Finally, section V gives simulation results and comparison with previous approaches are discussed.

II. Camera model and identities

This section offers a brief overview of the models associated with stereo cameras and some algebraic identities from the structure of those models.

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 two cameras are identified as camera1 and camera 2. The stereo projective

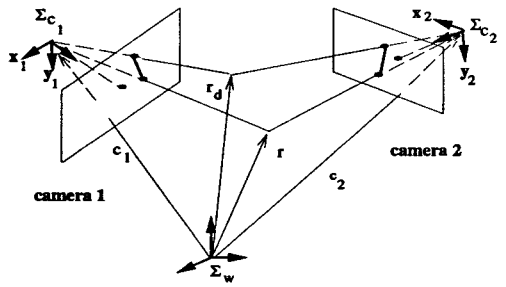


Fig. 1. Stereo camera model.

transformation, $g : R^3 \rightarrow R^4$, maps a spatial point, $\boldsymbol{r} \in R^3$ to a pair of image plane points, $\boldsymbol{\theta} \in R^4$, $\boldsymbol{\theta} = (\theta_{1,x}\ \theta_{1,y}\ \theta_{2,x}\ \theta_{2,y})^T$. In general, this transformation can be written as

$$\boldsymbol{\theta} = g(\boldsymbol{r}) = \begin{pmatrix} f_1 \dfrac{[\boldsymbol{r}-\boldsymbol{c}_1]^T \boldsymbol{x}_1}{[\boldsymbol{r}-\boldsymbol{c}_1]^T \boldsymbol{z}_1} \\ f_1 \dfrac{[\boldsymbol{r}-\boldsymbol{c}_1]^T \boldsymbol{y}_1}{[\boldsymbol{r}-\boldsymbol{c}_1]^T \boldsymbol{z}_1} \\ f_2 \dfrac{[\boldsymbol{r}-\boldsymbol{c}_2]^T \boldsymbol{x}_2}{[\boldsymbol{r}-\boldsymbol{c}_2]^T \boldsymbol{z}_2} \\ f_2 \dfrac{[\boldsymbol{r}-\boldsymbol{c}_2]^T \boldsymbol{y}_2}{[\boldsymbol{r}-\boldsymbol{c}_2]^T \boldsymbol{z}_2} \end{pmatrix} \tag{1}$$

where $\boldsymbol{c}_i$ and f_i represent the location and the focal length of camera i, and $\boldsymbol{x}_i$, $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$ are the orthogonal unit vectors defining the local coordinate system of camera i with $\boldsymbol{z}_i$ directed along the optical axis of the camera i. All positions, $\boldsymbol{c}_i$ and $\boldsymbol{r}_i$ and orientations $\boldsymbol{x}_i$, $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$ are expressed with respect to Σ_w [8].

Defining $D_i := (\boldsymbol{r}-\boldsymbol{c}_i)^T \boldsymbol{z}_i$, the Jacobian of the projective transformation g evaluated at $\boldsymbol{r}$, $Dg(\boldsymbol{r})$, can be written as

$$Dg(\boldsymbol{r}) = \begin{pmatrix} -f_1[(\boldsymbol{r}-\boldsymbol{c}_1)\times \boldsymbol{y}_1]^T/D_1^2 \\ f_1[(\boldsymbol{r}-\boldsymbol{c}_1)\times \boldsymbol{x}_1]^T/D_1^2 \\ -f_2[(\boldsymbol{r}-\boldsymbol{c}_2)\times \boldsymbol{y}_2]^T/D_2^2 \\ f_2[(\boldsymbol{r}-\boldsymbol{c}_2)\times \boldsymbol{x}_2]^T/D_2^2 \end{pmatrix} \in R^{4\times 3} \tag{2}$$

where ere $\times$ denotes the vector cross product. The Jacobian matrix relates the rate changes in position in work-space to the rate changes in camera image planes. Note that this Jacobian will lose rank when $(\boldsymbol{r}-\boldsymbol{c}_1)\times(\boldsymbol{r}-\boldsymbol{c}_2)=\boldsymbol{0}$, that is, $\boldsymbol{r} \in \overline{c_1 c_2}$. In other words, if the point is moving along the line connecting the focal points of the cameras, its image points in image planes do not change.

2. Some useful algebraic identities

We begin by noting that (2.1) can equivalently be written in the form [7]

$$\boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{r} = \boldsymbol{b}(\boldsymbol{\theta}) \tag{3}$$

where

$$\boldsymbol{A}(\theta) = \begin{pmatrix} [\theta_{1,x}\boldsymbol{z}_1 - f_1\boldsymbol{x}_1]^T \\ [\theta_{1,y}\boldsymbol{z}_1 - f_1\boldsymbol{y}_1]^T \\ [\theta_{2,x}\boldsymbol{z}_2 - f_2\boldsymbol{x}_2]^T \\ [\theta_{2,y}\boldsymbol{z}_2 - f_2\boldsymbol{y}_2]^T \end{pmatrix} \in R^{4\times 3},$$

$$\boldsymbol{b}(\boldsymbol{\theta}) = \begin{pmatrix} \boldsymbol{c}_1^T[\theta_{1,x}\boldsymbol{z}_1 - f_1\boldsymbol{x}_1] \\ \boldsymbol{c}_1^T[\theta_{1,y}\boldsymbol{z}_1 - f_1\boldsymbol{y}_1] \\ \boldsymbol{c}_2^T[\theta_{2,x}\boldsymbol{z}_2 - f_2\boldsymbol{x}_2] \\ \boldsymbol{c}_2^T[\theta_{2,y}\boldsymbol{z}_2 - f_2\boldsymbol{y}_2] \end{pmatrix} \in R^4.$$

Conversely, we can recover the Cartesian position of a point about corresponding features in the scene by the inverse transformation of the camera perspective transformation, $g^{\dagger}(\boldsymbol{\theta}) : R^4 \rightarrow R^3$,

$$g^{\dagger}(\theta) = [A^T(\theta)A(\theta)]^{-1}A^T(\theta)b(\theta) \quad (4)$$

where $g^{\dagger} \circ g$ is the identity transformation of R^3. Additionally, we can develop an algebraic properties associated with the perspective projection $g(r)$ and its Jacobian.

Property 1 : Jacobian matrix $Dg(r)$ in (2) can be factored into state (Cartesian position) dependent and measurement (its image) dependent matrices,

$$Dg(r) = -\Gamma^{-1}(r)A(\theta), \quad (5)$$

where

$$\Gamma(r) = \begin{pmatrix} z_1^T(r - c_1)I_2 & 0 \\ 0 & z_2^T(r - c_2)I_2 \end{pmatrix} \Gamma(r) \in R^{4\times 4}$$

denotes a 4×4 diagonal matrix whose two diagonal blocks correspond to the depth information missing from the respective camera image plane and I_2 denotes a 2×2 identity matrix. Note that whenever r is visible from both cameras (in front of them), $\Gamma(r)$ is a positive definite matrix.

We can further develop an algebraic relationship between two points in work-space and their corresponding image points.

Property 2 : Given that $\hat{\theta} = g(\hat{r})$, it follows that

$$\begin{aligned} A(\hat{\theta})(\hat{r} - r) &= -\Gamma(r)(\hat{\theta} - \theta) \\ A(\theta)(\hat{r} - r) &= -\Gamma(\hat{r})(\hat{\theta} - \theta). \end{aligned} \quad (6)$$

These properties of the projective transformation afford the design of simple controllers, as we show in section IV. (Proofs are shown in Appendix)

III. Statement of the problem and the approaches

In this section we present a formal statement of the problem of three degree of freedom relative positioning of a manipulator's end-effector and review traditional approaches to the presented problem setting.

1. Statement of the problem

Suppose a distinguished point on a manipulator's gripper, r can be imaged by a stereo camera system, $\theta = g(r)$. Then the visual servoing task we wish to address is: Given a desired points in stereo camera image planes, θ_d, and a robot controller, u, capable of commanding arbitrary velocities of the gripper in the Cartesian work-space,

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

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

$$g(r) \rightarrow \theta_d.$$

This is finally zero disparity between gripper and target points in images and implies the alignment of an gripper with an identified target in work-space.

If we define $\theta_e := \theta_d - \theta$ as the "visual" error, then the error dynamics for such a system becomes

$$\theta_e = g(r_d) - g(r) \quad (7)$$

$$\dot{\theta}_e = -Dg(r)u \quad (8)$$

Before presenting our control strategies we review some well known methods of root finding and point out these traditional techniques cannot be satisfactorily used to generate controllers in the present problem setting.

2. Alternative approaches

There are two traditional algorithms for iteratively solving root finding problems of the kind,

$$g(r) - g(r_d) = 0.$$

In the sequel we will distinguish between Newton and gradient based variants of these solution. With guarantees of convergence arising from well known analysis one can employ Newton-Raphson,

$$\dot{r} = Dg^{\dagger}(r)\theta_e \quad (9)$$

or gradient descent

$$\dot{r} = Dg^T(r)\theta_e \quad (10)$$

where $Dg^{\dagger}$ denotes the generalized inverse of the nonsquare Jacobian matrix[9].

Of course there are numerous variants on these schemes. Typically the gradient based descent techniques exhibit slow convergence near the solution, and so the Newton technique is generally preferable since it gives more uniform rates of convergence. The convergence of both techniques is, of course, dependent upon exact calibration. In the present problem setting we are unable to implement either (9) or (10) directly since we are missing the Cartesian vector, r. The straightforward fix would be to replace r with the "triangulate" Cartesian position $g^{\dagger}(\theta)$ so that (9) and (10) become

$$\begin{aligned} \dot{r} &= Dg^{\dagger}(g^{\dagger}(\theta))\theta_e \\ \dot{r} &= Dg^T(g^{\dagger}(\theta))\theta_e \end{aligned}$$

respectively. But this seems to render the algorithms still more sensitive to calibration and sensor error.

Recently, Hager, *et al.* proposed a robust visual control strategy[11]. They precluded a Cartesian position-data expressly in their control law by the desire to work solely in image plane coordinates using a nonlinear observer that would estimate gripper's Cartesian position:

$$\begin{aligned} u &= [Dg(r)]^{\dagger}(k_1\theta_e + k_2\int \theta_e\, dt) \\ \hat{\dot{r}} &= u - M(\theta)\theta_e \end{aligned}$$

where $M(\theta)$ is a matrix function of the current measurement of the gripper. This fixes sensitivity problem in inverse transformation but raises new problems: there are more dynamics than needed, and

the stability of the overall system still remains unproven. In the following section we propose a visual control strategy without requiring any state observer or inverse transformation, and we furnish a stability proof for the proposed algorithm as well.

IV. A new "local" Newton and gradient methods

In place of the "global" controllers presented above controllers (9) and (10), we present 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. Nevertheless, we are able to furnish as "global" a convergence result for these as for controllers (9) and (10) by exploiting the structure of the perspective map.

1. Control laws

The Newton type controller which results from "local" approach takes the form

$$\boldsymbol{u} = -\boldsymbol{A}^{\dagger}(\boldsymbol{\theta}_d)\,\boldsymbol{K}\,\boldsymbol{\theta}_e \tag{11}$$

while the corresponding gradient algorithm is given by

$$\boldsymbol{u} = -\boldsymbol{A}^{T}(\boldsymbol{\theta}_d)\,\boldsymbol{K}\,\boldsymbol{\theta}_e \tag{12}$$

In these equations $\boldsymbol{K} \in R^{4\times 4}$ is an arbitrary positive definite diagonal gain matrix. Both of these presented control laws are distinguished from traditional approaches:

1) They are estimation free of the current position of end-effector, so that we can eliminate a significant source of calibration sensitivity.

2) They use a fixed gain on visual feedback error for a given image plane target since $\boldsymbol{A}(\boldsymbol{\theta}_d)$ is a constant matrix.

3) They are immune to the baseline calibration error since $\boldsymbol{A}(\boldsymbol{\theta}_d)$ depends upon neither $\boldsymbol{c}_1$ nor $\boldsymbol{c}_2$, the camera positions as seen in (3).

2. "Global" stability analysis

The convergence of the Newton type controller can be demonstrated by appeal to the Lyapunov function

$$v_N := \frac{1}{2}\,\boldsymbol{r}_e^{T}\,\boldsymbol{A}^{T}(\boldsymbol{\theta}_d)\boldsymbol{A}(\boldsymbol{\theta}_d)\,\boldsymbol{r}_e$$

where $\boldsymbol{r}_e := \boldsymbol{r}_d - \boldsymbol{r}$. The time derivative of v_N along the motion of the system defined is negative, as can be shown using property 1 and 2

$$\begin{aligned} \dot{v}_N &= -\boldsymbol{r}_e^{T}\boldsymbol{A}^{T}(\boldsymbol{\theta}_d)\boldsymbol{A}(\boldsymbol{\theta}_d)\boldsymbol{u} \\ &= \boldsymbol{r}_e^{T}\boldsymbol{A}^{T}(\boldsymbol{\theta}_d)\boldsymbol{A}(\boldsymbol{\theta}_d)\boldsymbol{K}\,\boldsymbol{\theta}_e \\ &= -\boldsymbol{\theta}_e^{T}\,\Gamma(\boldsymbol{r})\,\boldsymbol{K}\,\boldsymbol{\theta}_e \\ &\le 0 \end{aligned}$$

as long as $\boldsymbol{r}$ remains in the field of view throughout the motion, $\boldsymbol{z}_i^{T}(\boldsymbol{r}_i - \boldsymbol{c}_i) > 0$. The convergence of the corresponding gradient algorithm may be demonstrated by appeal to the Lyapunov function

$$v_G := \frac{1}{2}\,\boldsymbol{r}_e^{T}\,\boldsymbol{r}_e$$

for we have

$$\begin{aligned} \dot{v}_G &= -\boldsymbol{r}_e^{T}\boldsymbol{u} \\ &= \boldsymbol{r}_e^{T}\boldsymbol{A}^{T}(\boldsymbol{\theta}_d)\,\boldsymbol{K}\,\boldsymbol{\theta}_e \\ &= -\boldsymbol{\theta}_e^{T}\,\Gamma(\boldsymbol{r})\,\boldsymbol{K}\,\boldsymbol{\theta}_e \\ &\le 0 \end{aligned}$$

Again, we conclude that the closed loop system converges regardless of any positional calibration errors of either cameras, if the gripper remains visible from both cameras throughout the motion.

3. The need for containment

The "global" convergence of $\boldsymbol{\theta}_e$ resulting from any one of the algorithms (9), (10), (11), and (12) is qualified by the assumption that the gripper must stay "in front" of both cameras during its motion. However, none of these algorithms assures this, and as a result, the exact meaning of the conclusion 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 Laypunov 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.

V. Simulation results

A large sequence of computer simulation has been performed to evaluate the relative performance of the controllers of section IV 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 camera displacement errors, and thus we will focus only on situations involving rotational calibration error.

The simulation setup consists of a pair of cameras on pan-tilt head whose optical axes are parallel and perpendicular to the baseline as illustrated in Fig. 2.

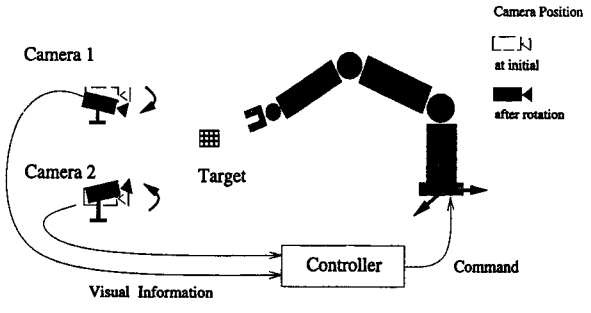


Fig. 2. Simulation setup.

The separation between the cameras is chosen as 100cm. The "task" we wish to perform is to position the simulated robot gripper at a point between the two cameras, and 250cm from baseline. From the point of view of the resolvability ellipsoid[10], the stereo pair with parallel axes exhibits less resolvability along the optical axis: that is, work-space information recovery from the sensor space is more sensitive to the camera calibration. In such situations, particularly, miscalibration does the most potential damage to the stability.

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 shear 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 miscalibration. In particular, we will consider situations where the calibration data represents cameras which are rotated from the actual cameras about one of the principle axes, and we will consider these in a pair-wise fashion(i,e, camera1 is rotated about z_i by -5 degrees and camera 2 is rotated about x_i by 5 degrees and so on). Of the possible 36 such configurations (two possible direction of rotation and 3 degrees of freedom for each camera) 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 100cm from the desired target position, and the percentage of trajectories which resulted in convertgence were measured for both Newton algorithm (9) and the proposed local variants (11) of section IV. A resulting 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 arrive within 1cm of the desired target within 90 sec.

The control gain should be carefully chosen based on practical consideration. In this simulation the overall gains of both algorithms are comparable such that the local dynamic behavior of both algorithms in the vicinity of the target location are similar. The simulation results are shown in Fig. 3. They demonstrate 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. Also, specific types of miscalibration result is shown in Fig. 4, where both miscalibrated cameras are rotated outward from the actual ones(i,e camera1 and camera 2 are rotated about $-y$ and y axis, respectively). Although not explicitly showed in Fig. 3, 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. And not surprisingly both methods showed significant robustness with respect to rotations around the optical axes.

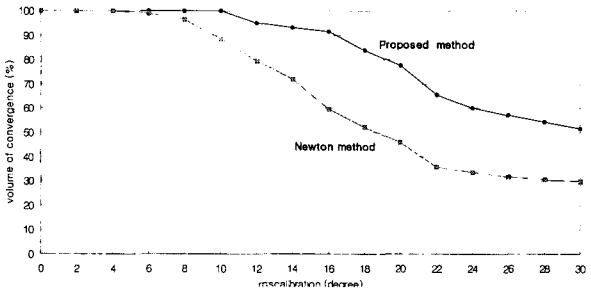


Fig. 3. Average volume of convergence counting all 21 types of rotational miscalibration.

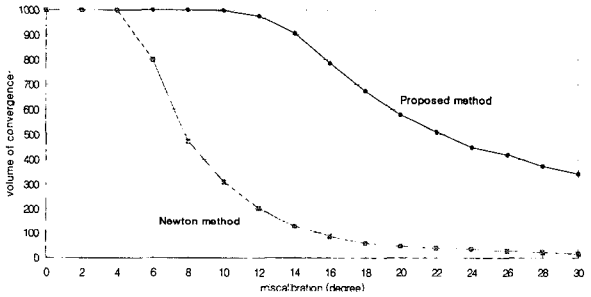


Fig. 4. Number of converging points in a specific type of miscalibration (both camreas are rotated outward).

VI. Conclusion

We present a simple visual servoing control algorithm for a three degree of freedom relative positioning of a gripper using stereo vision system. The presented controller does not resort to a dynamic estimator or inverse transformation in commanding gripper motion, distinguishing it from all previous approaches. We have shown that the presented 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, exhaustive simulation results show it is robust in the presence of significant miscalibration.

We have also ignored the problem of intrinsic parameter calibration - focal length, lens distortion effects, and so on. These are very important in practice but (with the exception of focal length) hard to model in a simple physically accurate form. Our success in simulation suggests that the algorithm may be robust against these phenomena, but clearly

more analysis would be required to draw convincing conclusions.

The presented work may provide 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.

References

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

[2] 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.

[3] K. Hashimoto and H. Kimura, "LQ optimal and nonlinear approaches to visual servoing," *Visual Servoing*, K. Hashimoto, editor, pp. 165-198 World Scientific, 1994.

[4] N. P. Papanikolopoulos and P. K. Khosla, "Adaptive robotic visual tracking: Theory and experiments" *IEEE Trans. on Automatic Control*, vol. 38, no. 3, pp. 429-445, Mar., 1993.

[5] 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.

[6] N. Hollinghurst and R. Cipolla, *Uncalibrated Stereo Hand-Eye Coordination*, Technical report, Cambridge University, Dept. of Engineering, Sep., 1993.

[7] G. D. 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.

[8] B. K. Horn, *Robot Vision*, MIT Press, Cambridge, 1986.

[9] D. Luenberger, *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, 1973.

[10] B. Nelson and P. K. Khosla, *The Resolvability Ellipsoid for Sensor Based Manipulation*, Technical report, Carnegie Mellon University, Oct. 1993.

[11] J. Munkres, *Analysis on Manifolds*, Addison-Wisley, 1990.

Appendix : Proofs of algebraic properties

Proof of Property 1 : $Dg(\boldsymbol{r}) = -I(\boldsymbol{r})^{-1}A(\boldsymbol{\theta})$

Denote $F(\boldsymbol{r}, \boldsymbol{\theta}) := A(\boldsymbol{\theta})\boldsymbol{r} - b(\boldsymbol{\theta})$ such that $F(\boldsymbol{r}, g(\boldsymbol{r})) = \mathbf{0}$. Using Implicit Function Theorem [11], the Jacobian $Dg(\boldsymbol{r})$ can be written as

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

Since

$$\frac{\partial F(\boldsymbol{r}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{\partial A(\boldsymbol{\theta})\boldsymbol{r}}{\partial \boldsymbol{\theta}} - \frac{\partial b(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$$

$$= \begin{pmatrix} \boldsymbol{z}_1^T(\boldsymbol{r} - \boldsymbol{c}_1)\, I_2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{z}_2^T(\boldsymbol{r} - \boldsymbol{c}_2)\, I_2 \end{pmatrix}$$

$$= I(\boldsymbol{r})$$

$$\frac{\partial F(\boldsymbol{r}, \boldsymbol{\theta})}{\partial \boldsymbol{r}} = A(\boldsymbol{\theta}),$$

A.1) becomes

$$Dg(\boldsymbol{r}) = -I(\boldsymbol{r})^{-1}A(\boldsymbol{\theta}).$$

Proof of Property 2 : $A(\hat{\boldsymbol{\theta}})(\hat{\boldsymbol{r}} - \boldsymbol{r}) = -I(\boldsymbol{r})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})$ in (6).

From $A(\boldsymbol{\theta})$ and $b(\boldsymbol{\theta})$ in (3) $A(\hat{\boldsymbol{\theta}})$ and $b(\hat{\boldsymbol{\theta}})$ can be written as

$$A(\hat{\boldsymbol{\theta}}) = A(\boldsymbol{\theta}) + D(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})Z \qquad \text{A.2)}$$

$$b(\hat{\boldsymbol{\theta}}) = b(\boldsymbol{\theta}) + D\left(\begin{pmatrix} \boldsymbol{c}_1^T \boldsymbol{z}_1 \\ \boldsymbol{c}_1^T \boldsymbol{z}_1 \\ \boldsymbol{c}_2^T \boldsymbol{z}_2 \\ \boldsymbol{c}_2^T \boldsymbol{z}_2 \end{pmatrix}\right)(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \qquad \text{A.3)}$$

where $D(\boldsymbol{a})$ denotes the diagonal matrix whose ith diagonal elements are a_i, and Z denotes

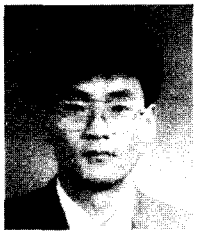
$$Z = \begin{pmatrix} \boldsymbol{z}_1^T \\ \boldsymbol{z}_1^T \\ \boldsymbol{z}_2^T \\ \boldsymbol{z}_2^T \end{pmatrix} \in R^{4\times 3}.$$

Then using A.2) and A.3) we have

$$\begin{aligned} &A(\hat{\boldsymbol{\theta}})(\hat{\boldsymbol{r}} - \boldsymbol{r}) \\ &= A(\hat{\boldsymbol{\theta}})\hat{\boldsymbol{r}} - A(\hat{\boldsymbol{\theta}})\boldsymbol{r} \\ &= A(\hat{\boldsymbol{\theta}})\hat{\boldsymbol{r}} - [A(\boldsymbol{\theta}) + D(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})Z]\boldsymbol{r} \\ &= b(\hat{\boldsymbol{\theta}}) - b(\boldsymbol{\theta}) - D(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})Z\boldsymbol{r} \\ &= D\left(\begin{pmatrix} \boldsymbol{c}_1^T \boldsymbol{z}_1 \\ \boldsymbol{c}_1^T \boldsymbol{z}_1 \\ \boldsymbol{c}_2^T \boldsymbol{z}_2 \\ \boldsymbol{c}_2^T \boldsymbol{z}_2 \end{pmatrix}\right)(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) - D(Z\boldsymbol{r})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \\ &= -I(\boldsymbol{r})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}). \end{aligned}$$

Similarly, $A(\boldsymbol{\theta})(\hat{\boldsymbol{r}} - \boldsymbol{r}) = -I(\hat{\boldsymbol{r}})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})$ in (6) can be shown as

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



김 동 민

1979 서울대학교 전기공학과 졸업. 동대학원 석사(1984). 1996년 University of Michigan, Ann Arbor 졸업(공학박사). 1997-현재 홍익대학교 과학기술대학 전기전자컴퓨터 공학부 전임강사. 주 관심분야는 동역학 시스템 제어, 로봇제어, 시각제어, 기구학적 변수 추정.