특징점이 Field of View를 벗어나지 않는 새로운 Visual Servoing 기법

A Novel Visual Servoing Approach For Keeping Feature Points Within The Field-of-View

박도환*, 염준형*, 박노용**, 하인중***
Do-Hwan Park, Joon-Hyung Yeom, Noh-Yong Park, In-Joong Ha

Abstract – In this paper, an eye-in-hand visual servoing strategy for keeping feature points within the FOV(field-of-view) is proposed. We first specify the FOV constraint which must be satisfied to keep the feature points within the FOV. It is expressed as the inequality relationship between (i) the LOS(line-of-sight) angles of the center of the feature points from the optical axis of the camera and (ii) the distance between the object and the camera. We then design a nonlinear feedback controller which decouples linearly the translational and rotational control loops. Finally, we show that appropriate choice of the controller gains assures to satisfy the FOV constraint. The main advantage of our approach over the previous ones is that the trajectory of the camera is smooth and circular-like. Furthermore, ours can be applied to the large camera displacement problem.

Key Words : Field-of-View, Visual Servoing, Nonlinear Feedback Control

1. Introduction

The aim of eye-in-hand visual servoing is to regulate the pose(position and orientation) of the camera at the desired pose relative to the object. In visual servoing, it is critical to keep the feature points of the object within the FOV of the camera through the whole period of servoing.

In IBVS(Image-Based Visual Servoing), the errors between the initial and desired positions of the feature points on the image plane are computed and the feature points are controlled to move to the desired positions on the image plane. Nonetheless, since IBVS does not control the camera motion in the Cartesian space directly, it sometimes reveals unnatural trajectories in the Cartesian space, such as camera retreat[1], and even might not converge to the desired pose[2].

On the other hand, in PBVS(Position- Based Visaul Servoing), the error between the initial pose and the desired pose in the Cartesian space are computed, but there is no direct control the feature points on the image plane. Therefore, the feature points might get out of the FOV. In [3], a switching control strategy among position-based control strategies and backward motion is proposed to resolve this problem. However, the camera motion might not be natural. In fact, the camera could be retreated excessively far from the object.

In this paper, a visual servoing approach for keeping the features within the FOV is proposed. In our approach, knowledge of the full 3D CAD model of the object is not required. Instead, knowledge of the distance between the object and the camera at the desired pose; and the size of the object is required. However, such information can be acquired through some off-line teaching process. In our approach, both the feature points on the image plane and the camera pose in the Cartesian space are considered simultaneously. Hence, the trajectory of the camera is smooth while the feature points remain within the FOV. Some simulation results using a 6 degree-of-freedom robotic manipulator show the validity and the practicality of our approach.

2. Preliminaries

First, we introduce some nomenclatures used in our development. Let X and Y be any two frames. Then, we denote the coordinate of a vector p and a point P in the work space with respect to frame X is denoted by ${}^{X}p \in R^3$ and ${}^{X}P \in R^3$, respectively. And the rotation matrix of frame Y with respect to the frame X is denoted by ${}^{X}R_Y \in R^{3\times 3}$. Next we define some frames needed in our development. We will often denote the

저자 소개
* 박도환: 서울대학 전기컴퓨터 공학부 박사과정
* 염준형: 서울대학 전기컴퓨터 공학부 박사과정
** 박노용: ASRI/IIRC 서울대학 전기컴퓨터 공학부 석사과정
***하인중: ASRI/IIRC 서울대학 전기컴퓨터 공학부 교수

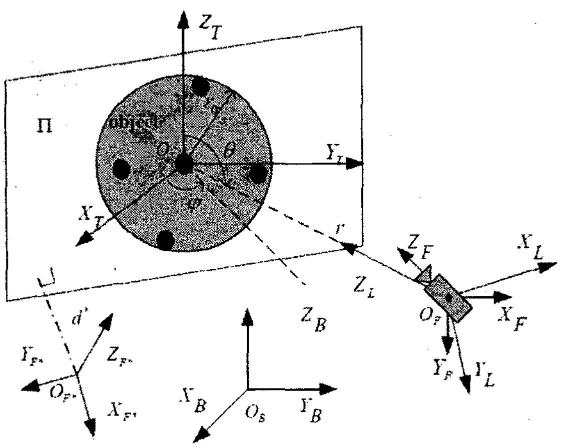

Fig. 1. Various frames used in our development

current camera frame C, the desired camera frame C^*, the current LOS frame L, the desired LOS frame L^*, the object frame O, and the robot base frame B.As can be seen from Fig. 1., the origins of the LOS frame and the camera frame are colocated and the Euler angles $\epsilon \triangleq [\epsilon_x \, \epsilon_y \, \epsilon_z]^T$ represent the angular displacements of frame L with respect to frame C. Here, the frames O and B are assumed to be fixed on the ground.

For practical convenience, we use the spherical coordinate system to express ${}^O p_C$ as follows.

$$ {}^O p_C = f_s(r,\theta,\phi) \triangleq [r\sin\theta\cos\phi \;\; r\sin\theta\sin\phi \;\; r\cos\theta]^T \tag{1} $$

We also introduce the new parameters ρ, ρ^* as follows.

$$ \rho(t) \triangleq r(t)/r_0, \; \rho^* \triangleq r^*/r_0 \tag{2} $$

where r_0 is the radius of the object sphere. Then, we define the state vector $x(t) \in R^6$ by

$$ x(t) \triangleq [\rho(t) \; \theta(t) \; \phi(t) \; \epsilon_x(t) \; \epsilon_y(t) \; \epsilon_z(t)]^T \tag{3} $$

and the current pose of the camera is represented by x uniquely. Now suppose that we have k feature points. In case of planar objects, the homography matrix H can be determined. And we can determine uniquely the quantities of x from H at each frame.

3. Main Results

In visual servoing applications, we often need more information beyond the image data from the camera. Some PBVS approaches might require the 3D CAD model of the object to estimate the camera pose[4]. On the other hand, the IBVS approaches need the depth estimation to calculate the image Jacobian[5]. We here use the ratio of (i) the distance between the camera and the object and (ii) object size.

The FOV(Field-of-View) of a rectangular shaped CCD camera can be described by the following set Ω_{xy}.

$$ \Omega_{xy} \triangleq \{[x \; y \; 1]^T \in R^3 | -L_{xm} < x < L_{xm} \; -L_{ym} < y < L_{ym}\} \tag{4} $$

where L_{xm} and L_{ym} are some positive constants. Control variables chosen in our approach for keeping the feature points within the FOV are the boresight errors and the scaled distance $(\rho, \epsilon_x, \epsilon_y)$. Therefore, we need to convert the physical constraint of the FOV to a more useful form.

$$ \Omega_0 \triangleq \cap \, \Omega_k(\rho) \tag{5} $$

where

$$ \Omega_k(\rho) \triangleq \{[\rho \; \epsilon_x \; \epsilon_y]^T \in R^3 | f_k(\epsilon_x, \epsilon_y) < \rho, \; [\epsilon_x \; \epsilon_y]^T \in \Omega_\epsilon\} \tag{6} $$

Here, the set Ω_ϵ and the function f_k are the constraints for ϵ_x, ϵ_y. Now we can easily show that all image feature points are whithin the FOV if and only if $[\rho(t) \; \epsilon_x(t) \; \epsilon_y(t)]^T \in \Omega_0$. Futhermore, we can also show that the set Ω_0 is an open convex subset of R^3.

Considering visual servoing task, first we take a teaching image of target object, and at the beginning of servoing we take initial image of target object. Using these two images, we can check the visibility constraint, that is $x_i, x^* \in \Omega_0$. If these two image satisfy the visibility condition, and we control the robotic manipulator to follow straight line trajectory in $r-\epsilon_x-\epsilon_y$ space, then $x(t) \in \Omega_0$ which means that all image feature points are whithin the FOV.

4. Control Scheme

Now, we propose the following nonlinear feedback controller.

$$ \begin{bmatrix} {}^C R_B v_C^* \\ {}^C R_B \omega_C^* \end{bmatrix} = \hat{L}^{-1} \left\{ \dot{\hat{x}}_d + K_p(\hat{x}_d - \hat{x}) + K_i \int (\hat{x}_d - \hat{x}) d\tau \right\} \tag{7} $$

where $\hat{L}^{-1}$ is the estimates of L^{-1} and K_p, K_i are diagonal matrices with proportional and integral gain. Here, the integral term is turned on when the camera reaches around the desired pose to compensate for the off-set uncertainties of the system.

The desired trajectory $\hat{x}_d(t)$ is given by

$$ \hat{x}_d(t) \triangleq \hat{x}_i + (\hat{x}^*(t) - \hat{x}_i) s(t) \tag{8} $$

where $s(t)$ is some trajectory such as well-known S-curve.

5. Experimental Results

In this section, experimental results are presented. Visual servoing system consists of a Sony XC-50 camera with Matrox meteor-II frame-grabber, Samsung MMC controller with Pentium4 3.2GHz, and Samsung AS-3 6-dof robot manipulator. Object is planar with 10 feature points which are white circles with black background. The state x is shown in Fig. 2 and Fig. 3 and it shows that this experiment has large positional and rotational displacement. Even though it has large camera displacement, state $\hat{x}$ follows x_d and converges to $\hat{x}^*$ and the feature points in image plane is kept within FOV in Fig. 4.

6. Conclusion

In this paper, a new visual servoing approach for keeping feature points within the Field-of-View is presented. Through some experimental results, we have shown that our approach keeps the feature points within the FOV, even though camera displacement is large. Furture work is to consider singularity avoidance and to adapt our approach to dynamic controller instead of kinematic controller in this paper.

Acknowledgement

본 연구는 한국과학기술원 영상정보특화연구센터(IIRC)를 통한 국방과학연구소의 연구비 지원으로 수행되었습니다.

참 고 문 헌

[1] O. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," Int. J. Pattern Recognition and Artificial Intelligence, vol. 2, no. 3, pp. 485-508, 1988.

[2] E. Malis, F. Chaumette, and S. Boudet, "2 1/2-D visual servoing," IEEE Trans. on Robotics and Automatation, vol. 15, pp.238-250, Apr. 1999

[3] Y. Fang, A. Behal, et al, "Adaptive 2.5D visual servoing of kinematically redundant robot manipulators," IEEE Conference on Decision and Control, vol 3. pp. 2860-2865, 2002

[4] W. Wilson, C. Hulls, and G. Bell, "Relative end-effector control using Cartesian position-based visual servoing", IEEE Trans on Robotics and Automation, vol. 12, October 1996, pp. 684-696

[5] Peter I. Corke, Seth A. Hutchinson, "A New Partioned Approach to Image-Based Visual Servo Control", IEEE Trans. on Robotics and Automation, Vol. 17, no. 4, August 2001, pp. 507-515

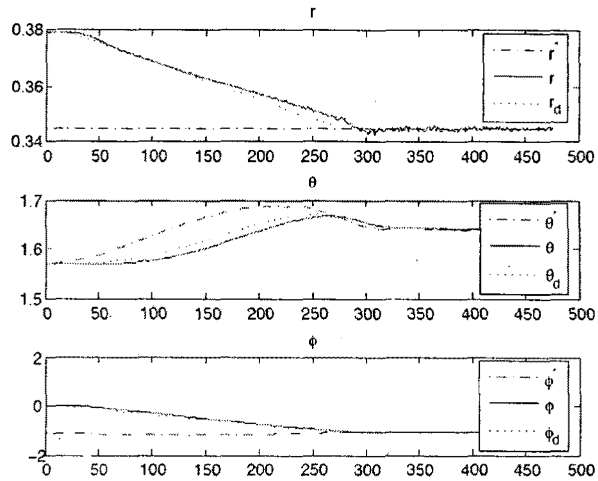

Fig. 2. Final, current, and desired trajectory value of states r, θ, ϕ.

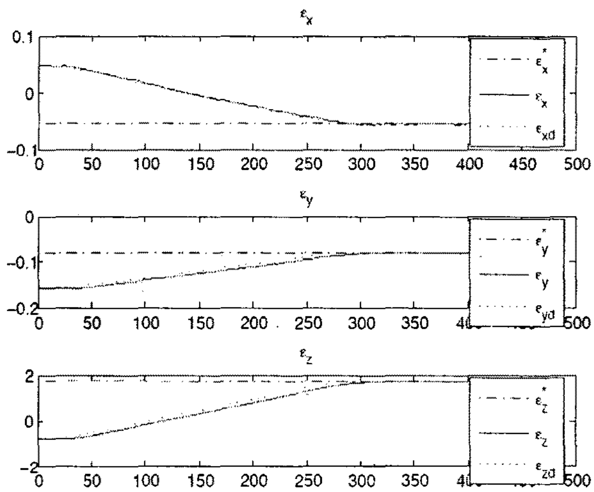

Fig. 3. Final, current, and desired trajectory value of states $\epsilon_x, \epsilon_y, \epsilon_z$.

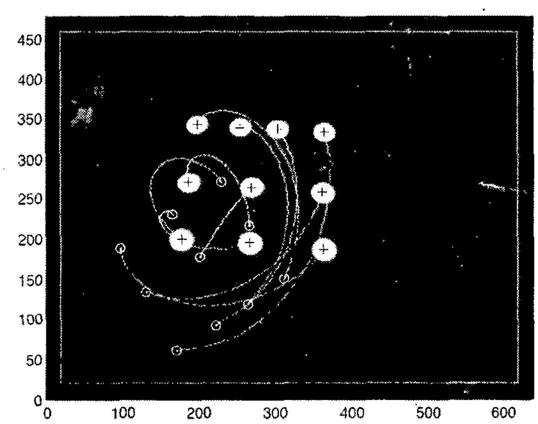

Fig. 4. Trajectory of feature points in image plane. White rectangular if FOV boundary and 'o' denotes the initial position and '+' denotes the final position.