

# 이동 물체 포착을 위한 비전 서보 제어 시스템 개발 Development of Visual Servo Control System for the Tracking and Grabbing of Moving Object

최규종 · 조월상 · 안두성

G. J. Choi, W. S. Cho and D. S. Ahn

**Key Words** : Taylor Series Expansion(테일러 급수전개), Gradient Descent Method(최급 강하법), Dynamic Broyden's Method(다이나믹 브로이던 방법), Visual Servoing(비주얼 서보잉), Stereo Vision System(스테레오 비전 시스템).

**Abstract** : In this paper, we address the problem of controlling an end effector to track and grab a moving target using the visual servoing technique. A visual servo mechanism based on the image based servoing principle, is proposed by using visual feedback to control an end effector without calibrated robot and camera models. Firstly, we consider the control problem as a nonlinear least squares optimization and update the joint angles through the Taylor Series Expansion. And to track a moving target in real time, the Jacobian estimation scheme(Dynamic Broyden's Method) is used to estimate the combined robot and image Jacobian. Using this algorithm, we can drive the objective function value to a neighborhood of zero. To show the effectiveness of the proposed algorithm, simulation results for a six degree of freedom robot are presented.

## 1. 서론

인간이 가지고 있는 감각 기관 중에서 가장 많은 정보를 인식할 수 있는 것은 시각이다. 따라서, 인간의 눈을 대신하는 카메라를 사용하여 로봇이 물체의 모양과 움직임을 가장 자연스럽게 해석할 수 있을 것이다. 이러한 비전 시스템(vision system)은 조립 작업, 경로 탐색, 주행 및 충돌회피, 무인 감시 장치, 교통량 제어, 불량 검출 등에서 폭 넓게 응용되고 있다.

로보틱스(robotics)와 관련된 대부분의 분야에서는 자코비안(Jacobian)을 이용하여 말단 효과 장치(end effector)를 제어한다. 그러나, 이러한 자코비안은 시스템 파라메타(system parameter)의 함수이며 또한 본질적으로 로봇의 위치에 따라 다르기 때문에 정확한 값을 얻기가 힘들다. 이러한 방식과는 달리 최근에 주목을 받고 있는 비주얼 서

보잉(visual servoing)은 기구학 및 시스템에 관련된 파라메타를 알지 못하더라도 카메라를 통해 입력된 영상을 이용하여 자코비안을 계속적으로 추정해 나가는 방식이다. 이러한 추정 자코비안(estimated Jacobian)은 영상에서의 특징 값들(features)을 목표 값들(desired features)로 점차적으로 수렴하도록 한다. 따라서, 고속의 계산을 필요로 하는 실시간 시스템에서 훨씬 더 효과적인 제어를 수행할 수 있다.

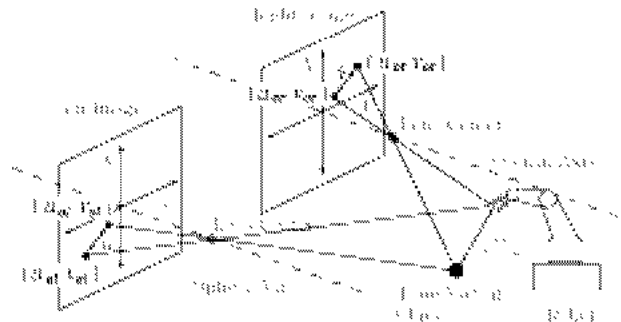


Fig. 1 Stereo vision system to extract features

접수일 : 2001년 8월 2일  
안두성 : 부경대학교 기계공학부  
최규종, 조월상 : 부경대학교 대학원

본 논문에서는 실시간으로 이동 물체를 포착 (grabbing)하여 추적(tracking)할 수 있는 시스템을 개발하고자 한다. 스테레오 비전 시스템(stereo vision system)을 사용하여 이동 물체(time varying object)와 말단 효과 장치의 영상을 획득하고, 획득된 영상을 분석하여 로봇의 말단 효과 장치와 이동 물체간의 상대적인 거리 오차(error)를 얻어내어 그 값이 영(zero)으로 수렴하도록 본 논문에서 제시하는 알고리즘을 적용하고자 한다.

## 2. 스테레오 비전 시스템

3차원 공간상에서 로봇의 말단 효과 장치가 이동하는 물체를 추적하기 위해서는 말단 효과 장치와 이동 물체의 상대적인 거리를 측정할 수 있어야 한다. 본 논문에서는 두 대의 CCD카메라를 사용하는 스테레오 비전 시스템을 구성하여 말단 효과 장치와 이동하는 물체의 상대적인 거리 오차를 계산한다.

먼저, 이동 물체와 말단 효과 장치의 위치를 왼쪽 영상 평면(left image plane)과 오른쪽 영상 평면(right image plane)에 대하여 구한다. Fig. 1에서  $[u_{ol} \ v_{ol}]$ 과  $[u_{or} \ v_{or}]$ 는 두 영상 평면에서 U V축에 관한 이동 물체의 위치 값들을 나타내며,  $[u_{el} \ v_{el}]$ 과  $[u_{er} \ v_{er}]$ 는 말단 효과 장치의 위치 값들을 나타낸다. 각 영상 평면에서 구한 이동 물체와 말단 효과 장치의 위치 값들을 이용하여 영상 평면상에서의 오차 값  $f_l$ 과  $f_r$ 을 계산할 수 있다. 여기서, 아래첨자  $o, e, l, r$ 는 각각 이동 물체, 말단 효과 장치, 왼쪽 영상 평면, 오른쪽 영상 평면을 나타낸다.

따라서, 영상 평면상에서 시간의 함수인 이동 물체의 위치  $y_o(t)$ 와 관절각의 함수인 말단 효과 장치의 위치  $y_e(\theta)$ 를 이용하여 상대 거리 오차  $f(\theta, t)$ 를 식(1)과 같이 표현할 수가 있으며, 이러한 값이 영으로 수렴하도록 말단 효과 장치의 관절각을 제어함으로써 이동 물체를 추적할 수 있다.

$$f(\theta, t) = y_e(\theta) - y_o(t) \quad (1)$$

## 3. 이동 물체 예측

각 영상 평면상에서의 이동 물체와 말단 효과

장치의 위치를 찾기 위하여 처리되는 영역을 일정한 크기의 관심영역(interest area)으로 설정함으로써 실시간 처리(real time processing)가 가능하다. 이동 물체와 말단 효과 장치의 초기 위치 값들을 알고 있다는 가정 하에 그 초기 위치 값들을 중심으로 하는 관심영역들을 설정하여 매 입력되는 전체 영상에서 설정된 관심영역 내의 영상들만을 처리함으로써 영상 처리 시간을 줄일 수 있다. 관심영역의 중심 위치는 과거에 입력된 영상에서부터 현재 입력된 영상까지의 이동 물체와 말단 효과 장치의 위치 값들을 이용하여 다음 관심영역의 중심 위치들을 예측할 수 있다.

따라서, 본 논문에서는 Fig. 2에서와 같이 각각의 영상 평면상에서 이동 물체와 말단 효과 장치의 위치 값을 중심으로 하는 관심영역을 설정하여 이전에 입력된 영상에서의 위치 정보  $u(k-1)$ 와  $v(k-1)$ , 그리고 현재 영상에서의 위치 정보  $u(k)$ 와  $v(k)$ 만을 이용하여 다음 관심영역의 중심 위치  $u(k+1)$ 와  $v(k+1)$ 를 식(2)와 같이 구한다.

$$\begin{aligned} du &= u(k) - u(k-1) \\ dv &= v(k) - v(k-1) \\ u(k+1) &= u(k) + du \\ v(k+1) &= v(k) + dv \end{aligned} \quad (2)$$

여기서,  $k$ 는 스텝(step)을 나타낸다.

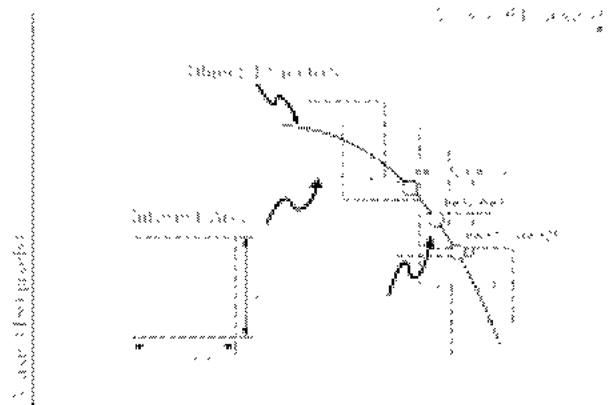


Fig. 2 The prediction algorithm of object position in the image plane

## 4. 이동 물체 추적 알고리즘

### 4.1 목적 함수

본 논문에서 제시하는 알고리즘(algorithm)의 목표는 비전 시스템에서 획득된 영상에서 말단 효과

장치와 이동 물체 사이의 상대적인 거리 오차  $f(\theta, t)$ 를 최소화시키도록 로봇의 관절각들을 실시간(real time)으로 갱신(update)시키는 것이다. 이러한 문제는 본질적으로 다변수 최적화 문제(multivariate optimization problem)로 볼 수 있으며, 비선형 최소 자승법2)(nonlinear least squares method)을 사용하여 풀 수 있다. 따라서, 목적 함수  $F(\theta, t)$ 를 식(3)과 같이 오차의 제곱(square)으로 표시한다.

$$F(\theta, t) = \frac{1}{2} f^T(\theta, t)f(\theta, t) \quad (3)$$

#### 4.2 목표 궤적 생성

말단 효과 장치의 궤적(desired trajectory)은 식(3)의 목적 함수값을 영(zero)으로 수렴하도록 생성해야 한다. Fig. 3과 같은 궤적( $x_d$ )을 생성하기 위해 관절각( $\theta_n$ )들은 테일러 급수 전개(Taylor series expansion)를 적용함으로써 구할 수 있다. 알고리즘은 Table. 1과 같이 유도할 수 있다. 여기서,  $h_\theta$ 와  $h_t$ 는 각각 관절각( $\theta$ )과 시간( $t$ )의 증분량이며,  $O^2$ 는 고차항들을 나타낸다.

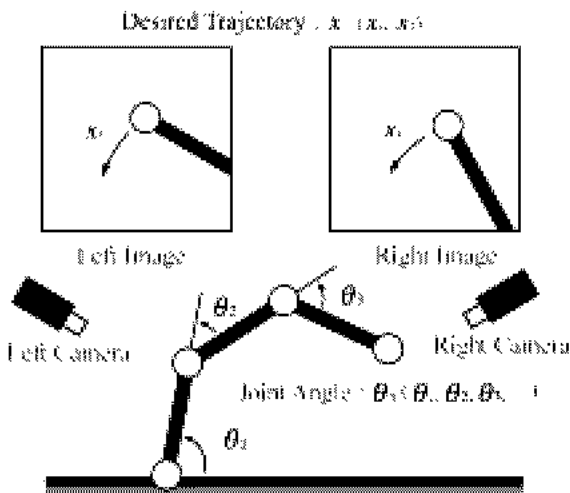


Fig. 3 Robot system equipped with visual sensor

즉, 목적함수  $F(\theta, t)$ 를 최소가 되게 하는  $\theta$ 값을 찾기 위해 테일러 급수 전개와 최급강하법(gradient descent method)을 사용함으로써 Table. 1과 같은 이동 물체 추적 알고리즘을 유도할 수 있다.

Table 1 The algorithm of desired trajectory

Step 1:	Optimize function $f(\theta, t) = \frac{1}{2} f^T(\theta, t)f(\theta, t)$
Step 2:	$f(\theta(t), t-h) - f(\theta(t)) + h \frac{\partial}{\partial \theta} f(\theta, t) + h \frac{\partial}{\partial t} f(\theta, t) = 0$
Step 3:	$\frac{\partial}{\partial \theta} f(\theta, t) + h \frac{\partial^2}{\partial \theta^2} f(\theta, t) + h \frac{\partial}{\partial \theta} f(\theta, t) = 0$
Step 4:	$h \frac{\partial^2}{\partial \theta^2} f(\theta, t) = - \left( \frac{\partial}{\partial \theta} f(\theta, t) + h \frac{\partial}{\partial t} f(\theta, t) \right)$
Step 5:	$h = \frac{\partial}{\partial \theta} f(\theta, t) \left  \frac{\partial}{\partial \theta} f(\theta, t) + h \frac{\partial}{\partial t} f(\theta, t) \right $
Step 6:	$\theta = \theta + \frac{\partial}{\partial \theta} f(\theta, t) \left  \frac{\partial}{\partial \theta} f(\theta, t) + h \frac{\partial}{\partial t} f(\theta, t) \right $
where	$\begin{cases} \frac{\partial^2}{\partial \theta^2} f(\theta, t) = \frac{\partial^2 f(\theta, t)}{\partial \theta^2} \\ \frac{\partial}{\partial \theta} f(\theta, t) = \frac{\partial f(\theta, t)}{\partial \theta} \frac{\partial \theta_1}{\partial \theta} + \frac{\partial f(\theta, t)}{\partial \theta} \frac{\partial \theta_2}{\partial \theta} + \frac{\partial f(\theta, t)}{\partial \theta} \frac{\partial \theta_3}{\partial \theta} \\ \frac{\partial}{\partial t} f(\theta, t) = \frac{\partial f(\theta, t)}{\partial t} \frac{\partial t}{\partial t} \\ \frac{\partial^2}{\partial \theta^2} f(\theta, t) = \text{matrix} \\ \frac{\partial^2}{\partial \theta^2} f(\theta, t) = \text{column of target} \end{cases}$

#### 4.3 자코비안 추정

제시된 알고리즘을 수행하기 위해서는 로봇의 관절각 및 특징 벡터(feature vector)를 획득할 수 있어야 한다<sup>3)</sup>. 관절각은 관절에 부착된 포텐시ومي터(potentiometer), 특징 벡터는 비전 센서(visual sensor)를 사용하여 각각 얻을 수 있다. 그리고, 관절각  $\theta \in R^n$ 와 특징 벡터  $x \in R^m$ 의 관계는 식(4)와 같이 함수관계  $g(\cdot)$ 로 표현할 수 있다.

$$x = g(\theta) \quad (4)$$

식(4)를 미분하면 식(5)와 같이 특징 벡터와 관절각 사이의 속도관계식을 얻을 수 있다.

$$\dot{x} = J(\theta) \dot{\theta} \quad (5)$$

여기서,  $J(\theta) \in R^{m \times n}$ 는 비전 시스템에서 획득한 영상의 특징 벡터와 로봇의 관절각을 사상시켜 주는 자코비안을 나타낸다. 그러나 실제 자코비안(true Jacobian)을 계산하기 위해서는 로봇의 기구학 구조, 카메라 파라메타, 카메라와 로봇의 변환행렬, 특징 벡터와 카메라 사이의 변환행렬 등을 모두 계산하여야 하며, 또한 자코비안은 본질적으로 시변함수(time variant function)이기 때문에 실시간 계산을 수행하기는 어렵다. 따라서, 본 논문

에서는 실시간 이동 물체 추적을 위하여 실제의 자코비안 기능을 수행할 수 있는 추정 자코비안 (estimated Jacobian:  $\hat{J}(t)$ )을 사용한다. 즉, 브로이던 방법(Broyden's Method)<sup>4)</sup>을 사용하여 실제 자코비안을 해석적으로 계산하는 대신에 Table. 2에서 유도한 알고리즘으로 자코비안을 계속적으로 추정하여 사용한다.

Table 2 The algorithm of estimated Jacobian

<p>Step 1. <math>X(t) = f(\theta)</math></p> <p>Step 2. <math>\frac{\partial X(t)}{\partial t} = \frac{\partial f(\theta)}{\partial \theta} \frac{\partial \theta(t)}{\partial t}</math></p> <p>Step 3. <math>\frac{\partial X(t)}{\partial t} = J_k \frac{\partial \theta(t)}{\partial t}</math>  <math>\frac{\partial X(t)}{\partial t} = J_k \frac{\partial \theta(t)}{\partial t}</math>  <small>where <math>J_k = \text{Jacobian}</math>, <math>\hat{J}_k = \text{estimated Jacobian}</math></small></p> <p>Step 4. <math>\frac{\partial X(t)}{\partial t} = J_k^T \frac{\partial \theta(t)}{\partial t} + \hat{J}_k^T \frac{\partial \theta(t)}{\partial t}</math>  <math>\frac{\partial J_k^T}{\partial t} \frac{\partial \theta(t)}{\partial t} = \frac{\partial J_k^T}{\partial t} \frac{\partial \theta(t)}{\partial t} + \hat{J}_k^T \frac{\partial \theta(t)}{\partial t}</math>  <small>where <math>\left  \frac{\partial \theta(t)}{\partial t} \right </math> is not zero vector</small></p> <p>Step 5. <math>\frac{\partial J_k^T}{\partial t} = \left[ \frac{\partial X(t)}{\partial t} - J_k^T \frac{\partial \theta(t)}{\partial t} \right] \left[ \frac{\partial \theta(t)}{\partial t} \right]^{-1}</math>  <math>\frac{\partial J_k^T}{\partial t} = \left[ \frac{\partial X(t)}{\partial t} - J_k^T \frac{\partial \theta(t)}{\partial t} \right] \left[ \frac{\partial \theta(t)}{\partial t} - \hat{J}_k^T \frac{\partial \theta(t)}{\partial t} \right]^{-1} \frac{\partial \theta(t)}{\partial t}</math>  <math>\frac{\partial J_k^T}{\partial t} = \frac{\left[ \frac{\partial X(t)}{\partial t} - J_k^T \frac{\partial \theta(t)}{\partial t} \right] \frac{\partial \theta(t)}{\partial t}}{\frac{\partial \theta(t)}{\partial t} - \hat{J}_k^T \frac{\partial \theta(t)}{\partial t}}</math></p>
---

5. 비주얼 서보잉(Visual Servoing)

본 논문에서 사용한 알고리즘은 Fig. 4와 같이 비주얼 피드백 루프(visual feedback loop)에서 영상에서의 오차  $f$ 를 계산하여 자코비안을 추정하고 추정된 자코비안을 사용하여 로봇의 관절각을 갱신하여 이동 물체를 추적하도록 한다.

먼저, 시스템에서 관절각을 갱신하는 알고리즘은 Table. 1에서 유도한 식(6)을 사용한다.

$$\theta_{k+1} = \theta_k - \left[ \frac{\partial f^T(\theta, t)}{\partial \theta} \frac{\partial f(\theta, t)}{\partial \theta} + \frac{\partial^2 f^T(\theta, t)}{\partial^2 \theta} f(\theta, t) \right]^{-1} \frac{\partial f^T(\theta, t)}{\partial \theta} [f(\theta, t) + k_t \frac{\partial f(\theta, t)}{\partial t}] \quad (6)$$

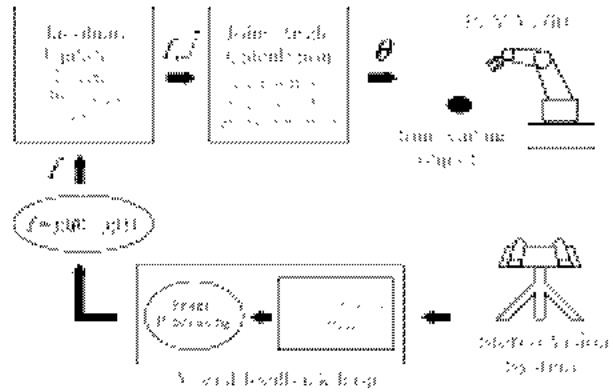


Fig. 4 Controller Block diagram using a nonlinear least squares optimization algorithm

여기서,  $\frac{\partial f(\theta, t)}{\partial t}$ 는 이동 물체의 속도를 나타내며,  $\frac{\partial^2 f^T(\theta, t)}{\partial^2 \theta}$ 는 사실상 추정하기 어려우며  $\theta_k$ 가 해(solution)로 수렴해갈 때 영(zero)으로 접근하기 때문에 무시할 수가 있다<sup>5)</sup>. 그리고, Table. 2에서 유도한 추정 자코비안( $\hat{J}_k$ )은 물체의 이동을 고려하기 위하여 이동 물체에 관련된 항( $\hat{y}_t$ )을 추가하여 정리하면 식(7)과 같다.

$$\hat{J}_k = \hat{J}_{k-1} + \frac{(\Delta f - \hat{J}_{k-1} k_\theta + \hat{y}_t k_t) k_\theta^T}{k_\theta^T k_\theta} \quad (7)$$

여기서,  $\Delta f = f_k - f_{k-1}$ 이다.

6. 시뮬레이션

본 논문에서는 이동 물체를 포착하여 실시간으로 추적하기 위해 Fig. 5와 같은 시스템을 구성하였다. 이동 물체를 포착 및 추적하기 위하여 스테레오 비전 시스템을 구성하였으며 이동 물체는 임의적으로 움직일 수 있는 이동 로봇을 사용하였다. 그리고, 이동 로봇을 비전에서 쉽게 인식할 수 있도록 색깔로써 구분을 지었다.

먼저, 본 논문에서 제시한 방법의 타당성을 검증하기 위하여 Corke가 제공하는 PUMA 560 로봇틱스 매트랩 툴박스<sup>6)</sup>를 사용하여 직선 운동과 원 운동을 하는 이동 물체에 대하여 시뮬레이션을 수행하였다. 식(8)과 (9)는 이동물체의 궤적을 나타내며 그래프 상에서 실선으로 표시를 하였으며 점선은 이동물체를 추적하는 말단효과장치의 궤적을 나타낸다. 그리고 샘플링 시간(sampling time)은 0.01초로 설정하였다.

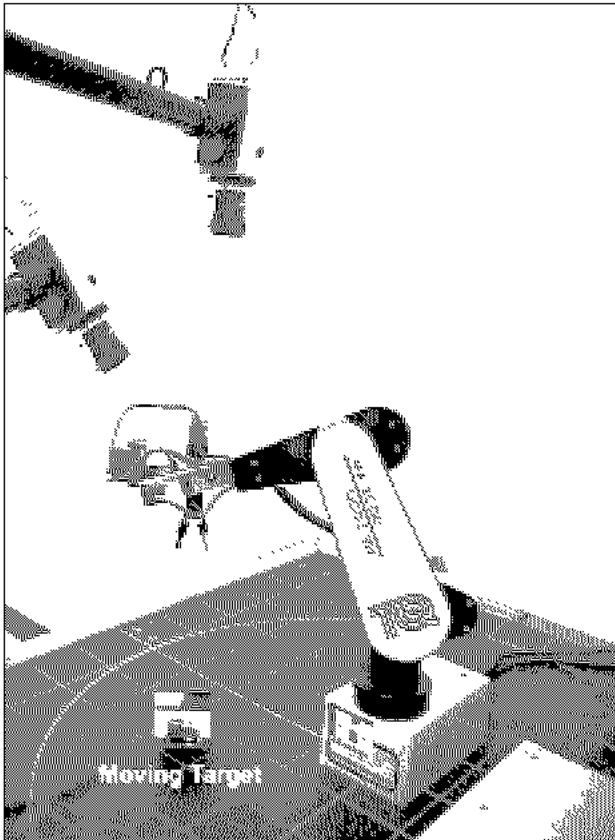


Fig. 5 Visual servoing system

직선 운동의 경우에 식(8)과 같이 직선 궤적을 따라 속도가 가변 되면서 움직인다. Fig. 6과 Fig. 7에서 알 수 있듯이 약 0.08초 후에 물체를 정상적으로 추적하고 있다.

$$\begin{aligned} x_{target} &= -0.5 \\ y_{target} &= 0.5 \sin(5t) \\ z_{target} &= 0.5 \end{aligned} \quad (8)$$

다음으로, 이동 물체는 식(9)와 같이 원형 궤적을 따라 움직인다. Fig. 8과 Fig. 9에서 알 수 있듯이 0.1초 후에 정상 상태에 도달하는 것을 볼 수 있다. 그리고, Fig. 10과 Fig. 11은 식(9)에 잡음(7pixels ~ +7pixels)을 첨가 시켜도 강인하게 이동 물체를 추적하고 있음을 보여주고 있다.

$$\begin{aligned} x_{target} &= -0.5 \\ y_{target} &= 0.5 \sin(5t) \\ z_{target} &= 0.5 \cos(5t) \end{aligned} \quad (9)$$

식(8)과 식(9)에서  $t$ 는 시간을 나타낸다. 로봇의 팔 단 효과 장치와 이동 물체사이의 거리 오차  $f$ 를 비전 시스템에서 획득된 두 개의 영상으로부터 얻어내어 영상에서의 상대적 거리 오차가 영(zero)으로 수렴하도록

본 논문에서 제시한 방법을 사용하여 실험한 결과 이동 물체를 효과적으로 추적하고 있음을 알 수 있다.

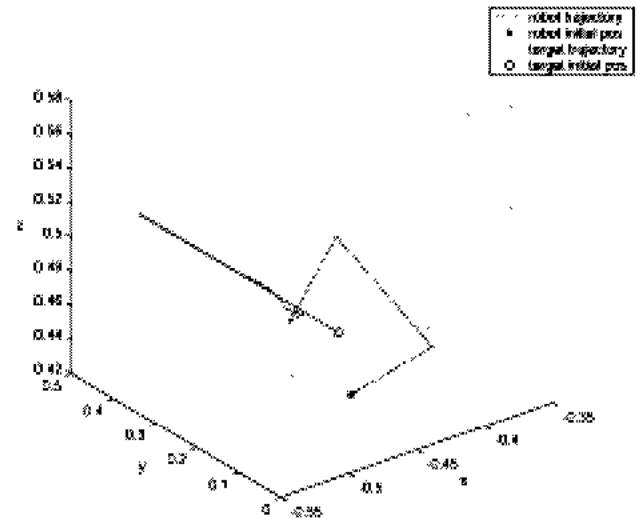


Fig. 6 Straight line trajectory without noise

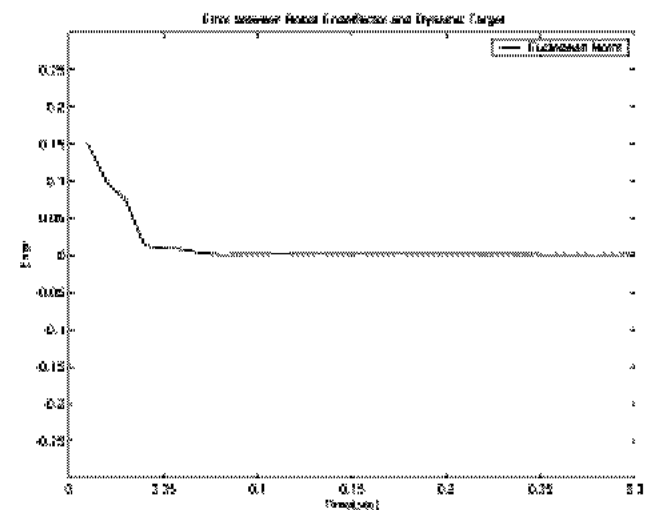


Fig. 7 Tracking error between robot and object

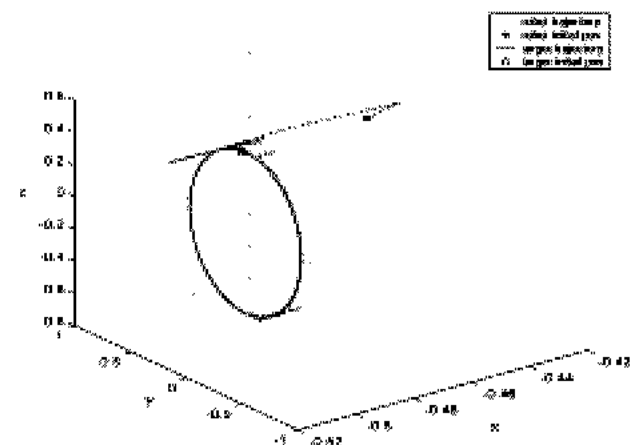


Fig. 8 Circular trajectory without noise

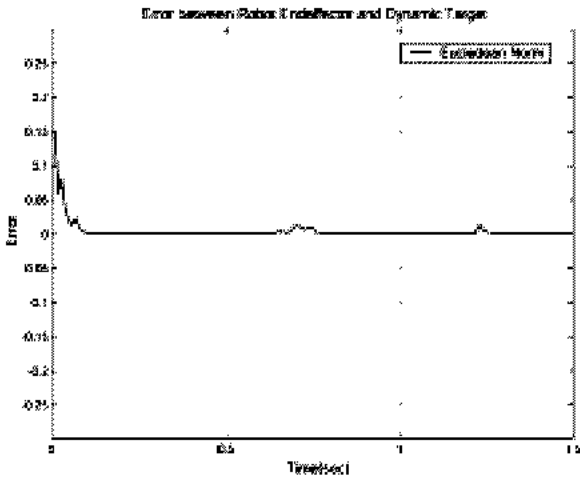


Fig. 9 Tracking error between robot and object

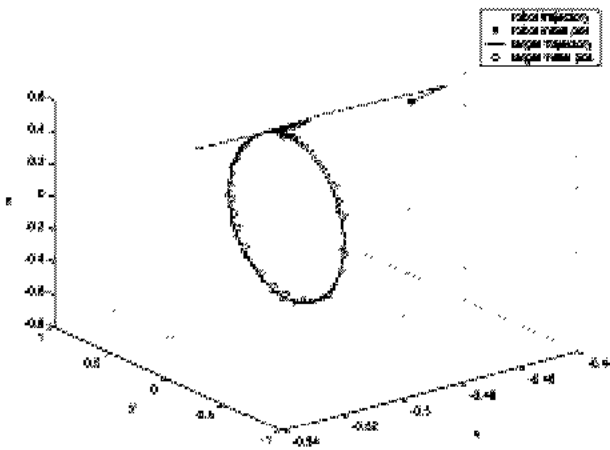


Fig. 10 Circular trajectory with noise

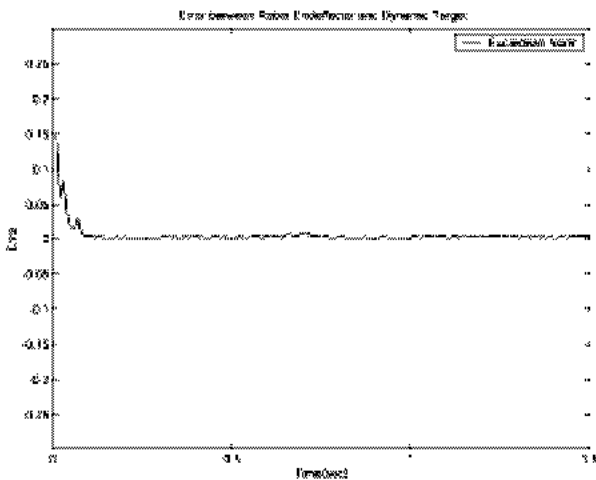


Fig. 11 Tracking error between robot and object

### 7. 결 론

본 논문에서는 로봇의 팔단 효과 장치가 실시간으로 이동 물체를 포착 및 추적하기 위해 스테레

오 비전 시스템을 이용하여 제어하는 문제를 다루었다. 획득된 영상 상에서의 팔단 효과 장치와 이동 물체사이의 상대적인 거리 오차를 측정하여 그 오차가 영(zero)으로 수렴하도록 로봇의 궤적을 생성하였다. 본 논문에서 제시한 알고리즘의 장점은 로봇 시스템과 비전 시스템사이의 캘리브레이션이 필요 없으므로 복잡한 계산과정을 필요로 하지 않기 때문에 실시간적으로 이동 물체 추적이 가능하다. 그러나, 초기화되는 자코비안의 값에 따라 정상상태로 수렴하는 시간이 다르기 때문에 최단시간 내에 정상 상태로 수렴할 수 있도록 보상해주는 알고리즘이 필요하다.

### 후 기

이 논문은 1998년도 귀뚜라미 문화재단의 연구비 지원에 의하여 연구되었음.

### 참고 문헌

1. 조기흠, 최병준, 전재현, 홍석교, "비전 시스템을 이용한 이동 물체 포착", KACC, Proceedings of the 13th, pp. 732~735, 1998
2. J. A. Piepmeyer, G. V. McMurray and H. Lipkin, "Tracking a Moving Target with Model Independent Visual Servoing : a Predictive Estimation Approach", International Conference on Robotics & Automation, Proceedings of the 1998 IEEE, pp. 944~949, 1998
3. K. Hosoda and M. Asada, "Versatile visual servoing without knowledge of true Jacobian", IEEE/RSJ/GI International Conference on Intelligent Robots & Systems, pp. 186~193, 1994
4. J. A. Piepmeyer, G. V. McMurray and H. Lipkin, "A Dynamic Jacobian Estimation Method for Uncalibrated Visual Servoing", International Conference on Advanced Intelligent Mechatronics, Proceedings of the 1999 IEEE/ASME, pp. 944~949, 1999
5. J. A. Piepmeyer, G. V. McMurray and H. Lipkin, "A Dynamic Qusi Newton Method for Uncalibrated Visual Servoing", International Conference on Robotics & Automation, Proceedings of the 1999 IEEE, pp. 1595~1600, 1999
6. P. I. Corke, "A robotics toolbox for matlab", IEEE Robotics and Automation, pp. 24~32, 1996