

하나의 카메라를 이용한 이동로봇의 이동물체 추적기법

Visual Tracking of Moving Target Using Mobile Robot with One Camera

한 영준*, 한 헌수

(Youngjune Han and Hernsoo Hahn)

Abstract : A new visual tracking scheme is proposed for a mobile robot that tracks a moving object in 3D space in real time. Visual tracking is to control a mobile robot to keep a moving target at the center of input image at all time. We made it possible by simplifying the relationship between the 2D image frame captured by a single camera and the 3D workspace frame. To precisely calculate the input vector (orientation and distance) of the mobile robot, the speed vector of the target is determined by eliminating the speed component caused by the camera motion from the speed vector appeared in the input image. The problem of temporary disappearance of the target from the input image is solved by selecting the searching area based on the linear prediction of target motion. The experimental results have shown that the proposed scheme can make a mobile robot successfully follow a moving target in real time.

Keywords : visual tracking scheme, mobile robot, moving target, real time, one camera

I. 서론

비전 시스템은 인간의 시각시스템에 유사한 형태로써 탐지 영역이 넓고 비접촉 상태에서 다양한 정보획득이 가능하기 때문에 다양한 분야에서 사용되고 있다. 예를 들면 물체의 인식[1,2], 검사[4], 3차원 모델링[4,5] 등 한정된 범위의 분야뿐만 아니라 최근의 이미지 처리 장비의 발달로 실시간으로 데이터 처리를 필요로 하는 원격제어[6,7], 지능형 교통정보 시스템[8], 물체추적 시스템[9] 등에 사용되고 있다. 특히, 물체만 움직이거나 로봇과 물체가 동시에 움직이는 동적 환경에서 움직이는 목표물 추적 시스템은 환경인식을 위해 비전 시스템을 주로 사용하고 있다. 이동로봇이 주행 중에 움직이는 목표물을 검출하고 추적함으로써 타물체와의 충돌을 피하면서 주행 경로를 설정할 수 있으며 궁극적으로는 목표물에 대해 필요한 작업을 수행할 수 있게 된다. 이와 같이 이동로봇의 동작이나 작업에 필요한 제어변수를 비전시스템을 이용하여 결정하는 기법을 시각서보(visual servo)시스템이라 한다.

시각서보 시스템은 출력오차신호가 3D 작업공간 좌표계 상에 정의된 경우와 직접적으로 영상 특징 공간상에서 정의된 경우에 따라 각각 위치 기반 제어(position-based control)와 영상기반 제어(image-based control)로 분류된다[10]. 예를 들면, Papanikolopoulos[9]는 2D 공간상에 미결정 속도로 움직이는 3D 물체의 위치정보를 제공하여 추적하는 알고리즘을 제시했다. 카메라를 빠르고 정확하게 제어하기 위해 비전시스템에서 관측되는 특징의 위치와 추적하려는 물체의 위치 차를 오차신호로 제공하는 제어 공식을 제시했다. 영상기반 제어 방법을 이용한 예는, Cadernat 등[11]이 제시한 움직이는 물체를 추적하는 시각서보 시스템과 2D

레이저 센서를 사용하여 장애물을 회피하는 알고리즘에서 찾아볼 수 있다. 또한 Hunt[12]는 물체 중심의 수학적 예측에 근거한 시각서보 시스템을 사용한 물체 추적에 관한 알고리즘을 제시했다. 이들 알고리즘은 중심 좌표의 계산이 필요했으며 느린 속도의 물체인 경우에만 추적이 가능했다.

일반적으로 시각서보 제어기법을 이용한 추적 시스템이 해결해야 과제는 다음의 3가지로 요약된다. 첫 번째 과제는 영상 내에서 움직이는 목표물의 위치 정보를 정확하게 얻는 것이다. 이를 위해 기존의 방법들은 3D 작업환경을 1D 또는 2D로 제한하는 방법[13]을 사용하거나 양안 카메라(stereo camera)를 사용하여 물체의 3D 정보를 구했다[14,15]. Luo[16]는 초음파 센서와 영상 정보를 융합하여 위치정보를 측정하는 방법을 제시했다. 이 알고리즘은 물체의 이동속도를 알고 있다는 가정 하에서 1D 시각서보 시스템을 이용한 물체 추적을 수행하였다. 대부분의 로봇은 3D 작업 환경에서 임무를 수행하기 때문에 작업 환경을 제한하는 방법은 범용성이 떨어진다. 후자의 방법은 물체의 3D 위치 정보를 예측할 수 있으나 시스템의 복잡성과 같은 장면의 다른 영상에서 일치하는 두 점을 찾아야 하는 어려움을 있다. 두 번째는 영상 내에서 움직이는 목표물을 안정적으로 검출하고 추적하는 것이다. 기존의 방법들은 템플릿 정합 방법인 SSD 알고리즘[17]을 사용하거나 Optical flow 방법을 사용하고 있다[18]. 물체와 카메라가 같이 움직이는 경우 물체의 형태가 시간에 따라 변형되기 때문에 템플릿 정합 방법을 계속적으로 사용하는데 한계가 있다. Dev[18]는 단일 카메라에서 얻어지는 연속 영상으로부터 optical flow를 계산해 복도의 중앙을 통과하도록 이동로봇을 제어하였다. 이동로봇의 제어에 필요한 시스템의 상태, 벽의 상대적 거리와 방향 정보를 optical flow로부터 유도하였다. Optical flow 방법은 카메라가 고정되어 있는 경우는 잘 적용이 되지만 물체와 카메라가 함께 움직이는 경우는 카메라의 움직임 성분을 보상에 주어야 하기 때문에 이동로봇을 사용하여 움직이는 목표물 추적 시스템을 구현하기 힘들다. 세 번째는 물

* 책임저자(Corresponding Author)

논문접수 : 2003. 4. 21., 채택확정 : 2003. 7. 29.

한영준, 한헌수 : 숭실대학교정보통신전자공학부

(young@ssu.ac.kr/hahn@ssu.ac.kr)

※ 본 연구는 숭실대학교 교내 연구비 지원으로 연구되었음.

체의 추적을 위해 시각서보 시스템을 구현하고 제어하는 것이다. 기존의 방법들은 시각서보 시스템을 구현하는데 있어서 영상 내에서 물체의 특징 변화만을 고려하여 로봇의 joint-level control을 수행하거나 자코비안을 사용하였다[29]. 카메라가 이동하면서 물체를 추적하는 시각서보 시스템의 경우 영상 내에서 물체의 특징 변화는 카메라 이동에 따른 성분과 물체의 이동에 따른 성분의 합으로 표현된다[14]. 따라서 비주얼 서보 시스템을 정확히 제어하기 위해 두 가지 성분을 함께 고려해야 한다.

본 논문은 앞에서 거론한 3가지의 문제 중에서 추적하려는 물체의 위치정보를 하나의 카메라 영상에서 정확히 얻는 문제와 물체의 이동정보를 카메라(이동로봇)의 이동과 분리하여 이동로봇의 제어변수를 정확히 결정하는 문제를 해결하도록 한다. 제안하는 방법은 이동로봇과 추적하려는 물체가 일반적인 이동로봇의 작업환경에 움직인다는 점을 고려하여 물체의 위치정보를 간단히 결정하는 방법을 사용한다. 또한 이동로봇의 제어변수들은 영상좌표상의 특징 위치를 원하는 위치로 이동하도록 하는 피드백 성분과 물체의 움직임을 보상하는 피드포워드(feed-forward) 성분을 보상함으로써 유도된다.

II. 단일영상을 이용한 목표물의 위치 결정

1. 좌표계의 설정

로봇이 물체를 조작하거나 추적하는 경우, 물체의 크기 정보보다는 로봇을 기준으로 한 물체의 상대적인 위치정보가 일차적으로 중요하다. 물체의 로봇에 대한 상대적 위치는 로봇좌표계를 기준으로 2차원 평면상에서 물체의 위치를 결정하는 일이 우선적으로 이루어진다. 따라서 본 논문에서는 단일 카메라에서 얻어진 영상으로부터 물체의 위치 정보를 얻는 효과적인 방법을 제시하기 위해 이동로봇의 작업환경에 대해 다음의 2가지 사항을 고려한다.

- 1) 물체는 로봇좌표계가 정의하는 X-Y 평면상에 놓인다.
- 2) 물체의 이동은 로봇좌표계의 X-Y 평면과 접하여 발생한다.

1)항은 로봇의 이동이 로봇좌표계의 X-Y 평면상에서 이루어지고, 추적하려는 물체도 이와 동일함을 고려한 것이고, 2)항은 1)항과 연계하여 측정하려는 물체가 바닥과 접촉한 상태로 이동함을 고려한 것이다. 이와 같은 고려사항들은 일반적으로 이동로봇을 응용하는 환경을 기초한 것으로 보편성을 갖는다. 이와 같은 고려사항들을 기반으로 본 논문에서 다루려는 문제, 즉 움직이는 물체를 이동로봇을 이용하여 추적하는 시스템에 포함되는 각 좌표계들 간의 관계를 그림 1에서 보여준다. 그림 1에서 Σ_w 는 실세계좌표계, Σ_M 는 이동로봇좌표계, Σ_C 는 카메라좌표계를 각각 나타낸다. Σ_M 의 원점은 이동 로봇의 바퀴축의 중앙에 위치하는 것으로 가정하며, Σ_C 에서 영상이 맺히는 영상센서의 좌표계를 영상좌표계(Σ_1)로 정의한다.

추적하려는 Σ_w 상의 3차원 물체는 위치이동과 회전 정도에 따라 Σ_1 상에서 형태가 상당한 정도로 변화된다.

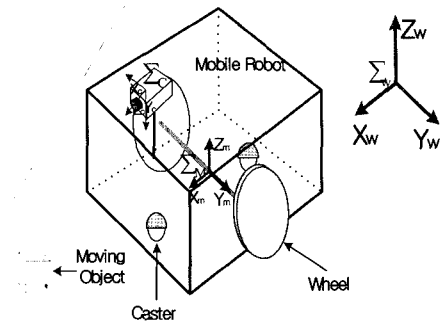


그림 1. 물체추적을 위한 좌표계의 설정.
Fig. 1. Definition of Coordinate Systems.

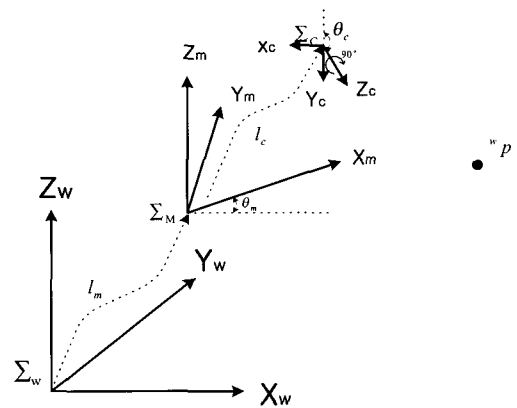


그림 2. 단일 카메라를 장착한 이동로봇을 이용한 물체 추적 시스템 좌표계.
Fig. 2. Relationship among coordinate systems: World(Σ_w), Robot(Σ_M), and Camera(Σ_C).

따라서 추적물체는 입력영상에서 SSD (Sum-of-Squared Difference) 알고리즘을 적용하여 유사성을 비교함으로써 검출하며 추적물체의 위치는 검출된 물체영역의 하단 중앙점의 좌표로 정의한다.

2. 물체의 위치 측정

그림 1과 같은 시스템 구성에서 좌표계간의 관계는 다음의 그림 2를 이용하여 설명한다. 작업환경에 관한 3가지 가정 중 두 번째 조건에 따라 이동로봇은 Σ_w 의 X-Y평면 위에서 이동하며 Σ_M 의 Z축을 중심으로 회전을 한다. 실세계 원점으로부터 이동로봇 좌표계의 이동변위를 (l_{mx}, l_{my}, l_{mz})로 표현한다. 여기서 Σ_M 은 바퀴축의 중앙에 위치하기 때문에 l_{mz} 는 상수이다. 그리고 이동 로봇의 Z축을 중심으로 양의 회전각을 θ_m 으로 표현된다.

Σ_C 는 Σ_M 의 원점으로부터 X축의 양의 방향으로 l_{xc} 그리고 Z축의 양의 방향으로 l_{zc} 떨어진 곳에 위치한다. 카메라는 Σ_C 의 X축을 중심으로 음의 방향으로 θ_c 만큼 회전하여 이동로봇의 상단 앞부분에 고정된 받침대 위에 놓여있다. Σ_M 은 Σ_w 의 Z축을 중심으로 양의 방향으로 θ_m 만큼

회전과 X축으로 l_{mx} , Y축으로 l_{my} , Z축으로 l_{mz} 만큼 이동했다면 Σ_M 와 Σ_W 사이의 관계는 다음과 같다.

$$\begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} = \begin{bmatrix} \cos \theta_m & \sin \theta_m & 0 \\ -\sin \theta_m & \cos \theta_m & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} - \begin{bmatrix} l_{mx} \\ l_{my} \\ l_{mz} \end{bmatrix} \quad (1)$$

카메라가 받침대 위에 고정되어 이동로봇 위에 장착되어 있기 때문에 받침대좌표계는 별도로 고려하지 않았다. Σ_C 의 원점은 Σ_M 의 원점으로부터 X축 양의 방향으로 l_{cx} 그리고 Z축 양의 방향으로 l_{cz} 떨어진 곳에 위치하고 Z축을 중심으로 음의 방향으로 90° 만큼 회전하고 X축을 중심으로 음의 방향으로 θ_c 만큼 회전했다. Σ_C 와 Σ_M 간의 변환은 다음 식으로 표현된다.

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ \cos \theta_c & 0 & -\sin \theta_c \\ \sin \theta_c & 0 & \cos \theta_c \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} - \begin{bmatrix} l_{cx} \\ 0 \\ l_{cz} \end{bmatrix} \quad (2)$$

따라서 Σ_C 와 Σ_W 간의 변환은 (3)에 의해 주어진다.

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} \sin \theta_m & -\cos \theta_m & 0 \\ \cos \theta_c \cos \theta_m & \cos \theta_c \sin \theta_m & \sin \theta_c \\ \sin \theta_c \cos \theta_m & \sin \theta_c \sin \theta_m & -\cos \theta_c \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} - \begin{bmatrix} l_{cx} - l_{my} \\ l_{mx} \cos \theta_c - l_{mz} \sin \theta_c \\ l_{mx} \sin \theta_c + l_{mz} \cos \theta_c + l_{cz} \end{bmatrix} \quad (3)$$

마지막으로 3차원 Σ_C 와 이상적인 영상좌표계(Σ_I)의 변환은 (4)에 의해 주어진다.

$$\begin{aligned} x_i &= f \frac{x_c}{z_c} \\ y_i &= f \frac{y_c}{z_c} \end{aligned} \quad (4)$$

Σ_W 와 Σ_I 관계식을 얻기 위해 (4)에 (3)를 대입하면 다음의 식이 얻어진다.

$$\begin{aligned} x_i &= f \frac{\sin \theta_m x_w - \cos \theta_m y_w + l_{my} - l_{cx}}{\sin \theta_c \cos \theta_m x_w + \sin \theta_c \sin \theta_m y_w} \\ &\quad - \cos \theta_c z_w - l_{mx} \sin \theta_c - l_{mz} \cos \theta_c - l_{cz} \\ y_i &= f \frac{\cos \theta_c \cos \theta_m x_w + \cos \theta_c \sin \theta_m y_w}{\sin \theta_c \cos \theta_m x_w + \sin \theta_c \sin \theta_m y_w} \\ &\quad + \sin \theta_c z_w - l_{mx} \cos \theta_c + l_{mz} \sin \theta_c \\ &\quad - \cos \theta_c z_w - l_{mx} \sin \theta_c - l_{mz} \cos \theta_c - l_{cz} \end{aligned} \quad (5)$$

작업환경에 대한 고려사항으로부터 물체가 Σ_W 의 X-Y 평면 위에 놓이기 때문에 물체의 위치는 $(x_w, y_w, 0)$ 로 표현된다. Σ_M 의 원점이 이동 로봇의 바퀴축의 중앙에 위치하기 때문에 l_{mz} 는 상수이고 카메라가 이동로봇에 고정되어 있기 때문에 l_{cx} , l_{cz} , θ_c 도 상수이다. (5)로부터 영상좌표 (x_i, y_i) 에 대응되는 실세계 좌표 $(x_w, y_w, 0)$ 에 대한 방

정식을 구하면 다음과 같다.

$$\begin{aligned} &(\sin \theta_c \cos \theta_m x_i - f \sin \theta_m) x_w \\ &\quad + (\sin \theta_c \sin \theta_m x_i - f \cos \theta_m) y_w \\ &= (l_{mx} \sin \theta_c + l_{mz} \cos \theta_c + l_{cz}) x_i - (l_{cx} - l_{my}) f \\ &(y_i \sin \theta_c \cos \theta_m - f \cos \theta_c \cos \theta_m) x_w \\ &\quad + (y_i \sin \theta_c \sin \theta_m - f \cos \theta_c \sin \theta_m) y_w \\ &= (l_{mx} \sin \theta_c + l_{mz} \cos \theta_c + l_{cz}) y_i \\ &\quad - (l_{mx} \cos \theta_c - l_{mz} \sin \theta_c) f \end{aligned} \quad (6a)$$

실세계 좌표 $(x_w, y_w, 0)$ 의 x_w 와 y_w 에 관해 2차 연립 방정식을 풀면 다음과 같다.

$$\begin{aligned} x_w &= \frac{DE - BF}{AD - BC} \\ y_w &= \frac{AF - CE}{AD - BC} \end{aligned} \quad (6b)$$

여기에서 각 변수의 값은 다음의 (6c)와 같다.

$$\begin{aligned} A &= \sin \theta_c \cos \theta_m x_i - f \sin \theta_m \\ B &= \sin \theta_c \sin \theta_m x_i - f \cos \theta_m \\ C &= y_i \sin \theta_c \cos \theta_m - f \cos \theta_c \cos \theta_m \\ D &= y_i \sin \theta_c \sin \theta_m - f \cos \theta_c \sin \theta_m \\ E &= (l_{mx} \sin \theta_c + l_{mz} \cos \theta_c + l_{cz}) x_i - (l_{cx} - l_{my}) f \\ F &= (l_{mx} \sin \theta_c + l_{mz} \cos \theta_c + l_{cz}) y_i \\ &\quad - (l_{mx} \cos \theta_c - l_{mz} \sin \theta_c) f \end{aligned} \quad (6c)$$

따라서 작업환경에 대한 고려사항으로부터 Σ_W 의 X-Y 평면에 접한 물체의 한 점에 대응되는 Σ_I 상의 점을 얻을 수 있다면 Σ_W 의 X-Y 평면 위에 놓인 물체의 위치는 식 (6)에 의해 결정된다.

III. 이동물체 추적을 위한 로봇제어변수의 결정

움직이는 목표물이 검출되면 이동로봇이 물체를 놓치지 않고 추적할 수 있게 다음의 조건을 만족하도록 로봇제어 변수, 즉 이동방향과 속도를 결정한다. 1) 이동로봇과 물체는 일정거리를 유지한다. 2) 추적물체는 이동로봇의 카메라에 항상 포착된다. 위의 조건을 만족하는 로봇제어변수를 결정하기 위해서는 영상에 주어진 정보를 이용하여 추적하려는 목표물의 속도를 측정하고 이동방향을 예측하여야 한다.

1. 영상 좌표계 상에서 이동물체의 속도측정

물체정보를 제공하는 카메라는 이동로봇에 장착되어 로봇과 함께 움직인다. 즉 카메라에 포착된 물체의 움직임은 실제물체의 움직임 성분과 카메라의 움직임에 의한 성분을 합한 것이 된다. 그림 3은 카메라와 움직이는 물체가 같이 움직였을 때 영상에서 움직이는 물체의 속도를 물체 움직임에 의한 속도와 카메라의 움직임에 의한 속도로 분해한 것을 보여준다.

그림 3에서 Σ_I 상의 물체속도($\dot{\mathbf{x}}_i$)는 카메라 이동에 의한 물체의 속도($\dot{\mathbf{x}}_i^c$)와 이동물체의 움직임에 의한 속도($\dot{\mathbf{x}}_i^t$)의 합으로 다음의 (7)에 의해 표현된다.

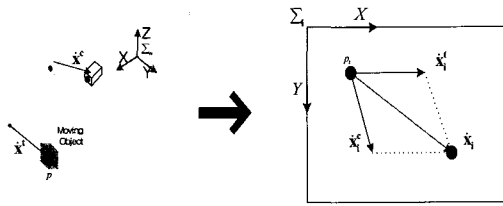


그림 3. 영상에서 보여지는 물체의 움직임 벡터를 물체와 카메라의 움직임 성분으로 분해.

Fig. 3. Separation of the speed vector in the input image into two speed components: one caused by motion of the mobile robot and one caused by motion of the target.

$$\dot{\mathbf{x}} = \dot{\mathbf{x}}_c + \dot{\mathbf{x}}_i \quad (7)$$

위 식에서 $\dot{\mathbf{x}}_i$ 는 Σ_1 상에서 움직이는 물체의 속도이기 때문에 영상에서 움직이는 물체의 이동 변위가 얻어지고 영상을 획득하는 주기(T)는 주어지기 때문에 $\dot{\mathbf{x}}_i$ 는 다음의 (8)로 근사화 할 수 있다. (8)에서 u 와 v 는 SSD 알고리즘에서 얻어지는 영상의 X축과 Y축 상에서의 이동물체의 변위 성분이다.

$$\dot{\mathbf{x}}_i \approx [u/T, v/T]^T \quad (8)$$

카메라의 움직임에 의한 이동물체의 속도 성분($\dot{\mathbf{x}}_c$)을 얻기 위해 (7)의 카메라 움직임에 의한 속도벡터 성분($\dot{\mathbf{x}}_c$)을 영상 자코비안(Jacobian)으로 표현하면 다음의 (9)와 같다.

$$\dot{\mathbf{x}}_c = \mathbf{J}_c \dot{\theta} + \dot{\mathbf{x}}_i \quad (9)$$

여기서 $\mathbf{J}_c = \frac{\partial \mathbf{x}}{\partial \theta}$ 이다. \mathbf{J} 는 목표물의 영상 특징 변화와 카메라의 joint 벡터 변화 사이의 관계를 표현하는 영상 자코비안이다. (9)로부터 이동하는 물체가 고정되었다고 가정하면 (9)는 다음의 (10)과 같이 된다.

$$\dot{\mathbf{x}}_i = \mathbf{J}_c \dot{\theta} \quad (10)$$

(7)와 (9)가 동일한 식이므로 영상 자코비안을 얻기 위해서 Σ_C 상에서 물체의 속도를 구한다. 이동물체가 고정된 상태에서 카메라의 이동속도 $\mathbf{T}_c = [t_{cx}, t_{cy}, t_{cz}]^T$ 와 각속도 $\mathbf{Q}_c = [\omega_{cx}, \omega_{cy}, \omega_{cz}]^T$ 가 주어졌을 때 Σ_C 상의 한 점 $\mathbf{x}_c = [x_c, y_c, z_c]^T$ 의 속도는 다음의 (11)으로 주어진다.

$$\dot{\mathbf{x}}_c = -\mathbf{T}_c - \mathbf{Q}_c \times \mathbf{x}_c \quad (11)$$

$\mathbf{x}_c = [x_c, y_c, z_c]^T$ 의 영상 좌표계 상의 대응점을 $\mathbf{x}_i = [x_i, y_i]^T$ 라고 할 경우, 카메라의 이동에 따른 \mathbf{x}_c 의 미분은 카메라 핀홀 모델 (4)을 (11)에 대입해서 얻는다.

$$\begin{aligned} \dot{x}_c &= -z_c \omega_{cy} + \frac{y_i z_c}{f} \omega_{cz} - t_{cx} \\ \dot{y}_c &= z_c \omega_{cx} - \frac{x_i z_c}{f} \omega_{cz} - t_{cy} \\ \dot{z}_c &= \frac{z_c}{f} (x_i \omega_{cy} - y_i \omega_{cx}) - t_{cz} \end{aligned} \quad (12)$$

위 식을 (4)를 미분한 방정식에 대입하여 행렬식으로 표현하면 다음의 (13)을 얻는다.

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \begin{bmatrix} -\frac{f}{z_c} & 0 & \frac{x_i}{z_c} & \frac{x_i y_i}{f} & \frac{-(f)^2 - (x_i)^2}{f} & y_i \\ 0 & -\frac{f}{z_c} & \frac{y_i}{z_c} & \frac{(f)^2 + (y_i)^2}{f} & -\frac{x_i y_i}{f} & -x_i \end{bmatrix} \begin{bmatrix} t_{cx} \\ t_{cy} \\ t_{cz} \\ \omega_{cx} \\ \omega_{cy} \\ \omega_{cz} \end{bmatrix} \quad (13)$$

따라서 (13)과 (10)에서 영상에서의 이동과 Σ_C 상에서의 이동에 대한 관계를 설명하는 영상 자코비안(\mathbf{J}_c)을 구하면 다음의 (14)과 같다.

$$\mathbf{J}_c = \begin{bmatrix} -\frac{f}{z_c} & 0 & \frac{x_i}{z_c} & \frac{x_i y_i}{f} & \frac{-(f)^2 - (x_i)^2}{f} & y_i \\ 0 & -\frac{f}{z_c} & \frac{y_i}{z_c} & \frac{(f)^2 + (y_i)^2}{f} & -\frac{x_i y_i}{f} & -x_i \end{bmatrix} \quad (14)$$

따라서 영상에서 이동물체의 속도벡터는 다음의 (15)에 의해 구할 수 있다.

$$\dot{\mathbf{x}}_i = \dot{\mathbf{x}}_c - \mathbf{J}_c \dot{\theta} \quad (15)$$

2. 이동로봇의 제어변수 결정

목표물을 추적하기 위해 이동로봇이 있어야할 목표위치와 방향은 영상에서 얻어진 추적 목표물의 위치와 방향에 의해 결정되며 목표위치로 이동하기 위한 제어입력은 목표위치와 현재위치의 차에서 얻어진다. 즉, 이동로봇의 제어는 영상에서 추적 목표물의 위치가 원하는 목표위치에 도달할 때까지 목표위치와 현재위치의 오차를 이동로봇의 제어변수로 제공함으로 이루어진다.

영상에서 얻어진 추적 목표물의 위치와 방향정보로부터 이동로봇의 제어입력벡터, \mathbf{u} 를 직접적으로 결정하는 시각서보 개념을 사용한다. 영상기반의 시각서보 제어기법은 영상에서 얻어진 추적 목표물의 위치와 속도를 기반으로 추적 목표물이 영상의 중앙에 위치하도록 이동로봇의 목표위치를 정하고 이를 영상 자코비안을 이용하여 이동로봇의 제어입력벡터 \mathbf{u} 로 변환하는 방법을 말한다. 따라서 \mathbf{u} 는 이동 로봇의 이동거리를 결정하는 속도성분 $\mathbf{T}_m = [t_{mx}, t_{my}, t_{mz}]^T$ 과 방향을 결정하는 각속도 성분 $\mathbf{Q}_m = [\omega_{mx}, \omega_{my}, \omega_{mz}]^T$ 을 결합한 $\mathbf{u} = [\mathbf{T}_m, \mathbf{Q}_m]^T$ 로 정의한다. 3.1절에서는 Σ_1 에서의 속도벡터를 영상자코비안 \mathbf{J}_c 를 이용하여 Σ_C 상에서의 속도벡터로 변환하는 수식을 유도하였다. 이동로봇을 제어하기 위해서는 Σ_C 가 Σ_M 으로 변환되어야 하므로 Σ_1 에서의 속도벡터를 Σ_M 에서의 속도벡터로 변환하기 위한 새로운 영상자코비안 \mathbf{J}_m 을 구해야 한다. 이는 그림 4에 주어진 Σ_C 와 Σ_M 에서의 속도성분과 각속도 성분들 간의 관계에서 유도될 수 있다. 먼저, Σ_C 에서의 제

어입력벡터 $[t_{cx}, t_{cy}, t_{cz}, \omega_{cx}, \omega_{cy}, \omega_{cz}]^T$ 가 Σ_M 상에서의 제어입력 벡터 $[t_{mx}, t_{my}, t_{mz}, \omega_{mx}, \omega_{my}, \omega_{mz}]^T$ 로 변환되는 과정이 (16)으로 설명됨을 보여준다.

$$\begin{bmatrix} t_{cx} \\ t_{cy} \\ t_{cz} \\ \omega_{cx} \\ \omega_{cy} \\ \omega_{cz} \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & l_{cx} & 0 & l_{cx} \\ \cos \theta_c & 0 & -\sin \theta_c & -l_{cx} \sin \theta_c & l_{cx} \cos \theta_c + l_{cz} \sin \theta_c & -l_{cx} \cos \theta_c \\ \sin \theta_c & 0 & \cos \theta_c & l_{cx} \cos \theta_c & l_{cx} \sin \theta_c - l_{cz} \cos \theta_c & -l_{cx} \sin \theta_c \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & \cos \theta_c & 0 & -\sin \theta_c \\ 0 & 0 & 0 & \sin \theta_c & 0 & \cos \theta_c \end{bmatrix} \begin{bmatrix} t_{mx} \\ t_{my} \\ t_{mz} \\ \omega_{mx} \\ \omega_{my} \\ \omega_{mz} \end{bmatrix} \quad (16)$$

영상 특징 변화와 이동로봇의 joint 벡터 변화 사이의 관계를 얻기 위해 (16)을 (13)식에 대입하면 이동로봇에 대한 영상 자코비안(J_m)을 다음의 (17)과 같이 얻을 수 있다.

$$J_m = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \end{bmatrix} \quad (17)$$

여기서 각각의 값은 다음과 같이 결정된다.

$$\begin{aligned} a_{11} &= \frac{x_i}{z_c} \sin \theta_c, \quad a_{12} = -\frac{f}{z_c}, \quad a_{13} = \frac{x_i}{z_c} \cos \theta_c, \\ a_{14} &= \frac{1}{z_c} (x_i l_{cy} \cos \theta_c - f l_{cz}) - \frac{f^2 + x_i^2}{f} \cos \theta_c + y_i \sin \theta_c, \\ a_{15} &= x_i \left\{ \frac{1}{z_c} (l_{cx} \sin \theta_c + l_{cz} \cos \theta_c) - \frac{y_i}{f} \right\}, \\ a_{16} &= -\frac{1}{z_c} (f l_{cx} + x_i l_{cy} \sin \theta_c) + \frac{f^2 + x_i^2}{f} \sin \theta_c + y_i \cos \theta_c, \\ a_{21} &= \frac{1}{z_c} (y_i \sin \theta_c - f \cos \theta_c), \quad a_{22} = 0, \\ a_{23} &= \frac{1}{z_c} (f \sin \theta_c + y_i \cos \theta_c), \end{aligned}$$

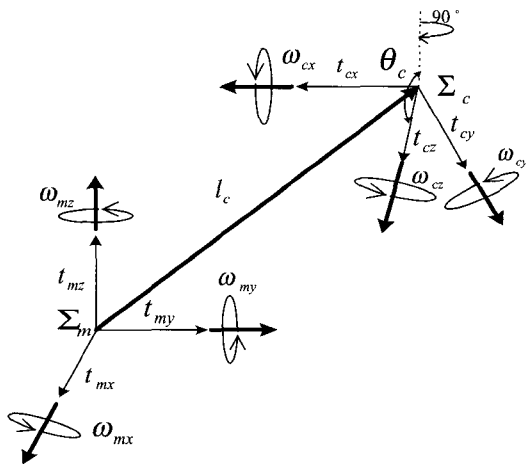


그림 4. 이동 로봇과 카메라의 속도 관계.

Fig. 4. Relationship between the speeds in Σ_m and Σ_c .

$$\begin{aligned} a_{24} &= \frac{l_{cy}}{z_c} (f \sin \theta_c + y_i \cos \theta_c) - x_i \left(\frac{y_i}{f} \cos \theta_c + \sin \theta_c \right), \\ a_{25} &= \frac{1}{z_c} \{ y_i (l_{cx} \sin \theta_c - l_{cz} \cos \theta_c) \\ &\quad - f (l_{cx} \sin \theta_c + l_{cz} \cos \theta_c) \} - \frac{f^2 + y_i^2}{f}, \\ a_{26} &= \frac{l_{cy}}{z_c} (f \cos \theta_c - y_i \sin \theta_c) + x_i \left(\frac{y_i}{f} \sin \theta_c - \cos \theta_c \right) \end{aligned}$$

이동 로봇의 제어는 Σ_I 에서 추적 목표물의 위치 x_i 를 원하는 위치 x_i^d 로 이동하도록 피드백 성분과 목표물의 움직임을 보상하는 피드포워드 성분을 보상함으로써 유도해 낼 수 있다. 이동로봇에 대한 영상 자코비안(J_m)이 얻어진다면, Σ_I 상에서 측정된 추적 목표물의 위치정보와 목표물의 움직임에 의한 속도 성분을 기반으로 카메라의 제어 입력 벡터 u 는 다음의 (18)로 표현할 수 있다.

$$u = J_m^+ \{ A (\dot{x}_i^d - \dot{x}_i) \} - J_m^+ \dot{x}_i^T \quad (18)$$

여기서 J_m^+ 는 J_m 의 pseudo 역 행렬이고 A는 $m \times m$ positive 이득 행렬이다.

3. 추적하는 목표물을 놓치는 경우 다시 추적하는 기법

카메라와 목표물이 움직이는 동적 작업환경에서 움직이는 목표물을 놓치는 상황은 다양하게 발생한다. 움직이는 물체를 추적하는 동안 목표물을 놓치는 경우는 크게 3가지로 요약할 수 있다. 첫 번째는 물체의 갑작스런 움직임에 의해 영상에서 목표물이 SSD 알고리즘의 탐색 범위를 벗어나는 경우이다. 본 논문에서 SSD 알고리즘의 탐색범위는 영상에서 추적물체의 직전위치로부터 16x16 영역으로 한정하고 직전위치가 영상의 중앙에 가깝기 때문에 탐색범위를 벗어난 경우에도 추적 목표물이 거의 영상 안에 존재하게 된다. 추적 목표물이 영상을 벗어나지만 않는다면 탐색범위를 확대하여 다시 찾을 수 있다. 두 번째는 추적 목표물의 갑작스런 회전, 혹은 물체 형태의 불규칙성에 의해 영상에서 물체의 형태가 급격히 변함으로 SSD 알고리즘의 템플릿 적용되지 않는 경우이다. 이 경우는 추적물체가 SSD 알고리즘의 탐색범위 안에 존재하는 경우로, 탐색범위 내에 존재하는 물체 중에서 추적 목표물을 선택해야한다. 세 번째는 작업환경에서 목표물이 다른 움직이는 물체를 가리면 움직이는 경우이다. 짧은 시간에 가리는 경우는 확장된 스네이크 알고리즘을 사용하기 때문에 템플릿이 이웃하는 물체의 영역까지 포함하더라도 이웃하는 물체가 사라진 후 다시 복구되지만 장시간 동안 가리는 경우는 목표물의 템플릿이 다른 물체의 영역으로 확대되기 때문에 물체를 놓칠 수 있다. 추적물체가 탐색영역에서 사라지는 원인을 위의 3가지로 분류했지만 실제로 어떤 원인에서 사라졌는지는 정확하게 판단하기 힘들다. 위의 원인 중에서 첫 번째의 탐색범위를 벗어난 경우를 제외하고는 영상처리기법을 통해 물체의 위치추적이 가능하다. 따라서 본 논문에서는 첫 번째 경우에 대한 해결방법만을 제시하기로 한다.

물체의 갑작스런 움직임에 의해 영상에서 목표물이 SSD

알고리즘의 탐색 범위를 벗어나는 경우는 물체의 템플릿의 면적의 크기가 순간적으로 커진다. 그리고 SSD 알고리즘의 척도값은 탐색 영역 안에 목표물이 존재하지 않기 때문에 갑작스럽게 커진다. 따라서 목표물의 면적 순간 변화율이 크고 목표물의 SSD 알고리즘의 척도값이 순간적으로 커지는 상황은 물체의 갑작스런 움직임에 의해 영상에서 목표물이 SSD 알고리즘의 탐색 범위를 벗어나는 경우로 판별한다. 이 경우 추적 목표물의 누적 경로정보를 이용하여 목표물의 진행경로를 선형 예측하는 방법과 탐색 시간을 줄이기 위해 지역 나선형 탐색(local spiral search) 방법을 사용한다. 추적물체의 위치를 예측하기 위해 가중치(weight)가 w_1, w_2, \dots, w_M 이고 영상에서 물체의 움직임에 의한 누적 성분 $x_i^t(n-1), x_i^t(n-2), \dots, x_i^t(n-M)$ 으로 구성된 전방향 선형 필터(forward linear predictor)를 사용한다. 추적대상물체의 예측 위치는 다음의 (19)으로 주어진다.

$$\hat{x}_i^t(n) = \sum_{k=0}^M w_k x_i^t(n-k) \quad (19)$$

위 식에서 가중치(w_k)는 위너필터(Wiener Filter)이론에 따라 최소자승 관점에서 최적화한다.

물체의 예측점이 구해지면 그림 5에서처럼 평균 탐색 시간을 줄이기 위해 지역 나선형 탐색 방법을 수행한다. 지역 나선형 탐색은 예측점으로부터 시계방향으로 회전하면서 한정된 영역(20x20)에 대해 SSD 알고리즘을 수행한다. SSD 알고리즘의 척도값이 임계값 이하이면 놓친 추적목표물이라 판단하고 탐색을 멈춘다.

IV. 실험

본 논문의 목표물 추적을 위한 이동로봇은 Nomadic Technology 사의 Nomad Scout Robot II이다. Nomad Scout Robot II는 16개의 초음파 센서와 접촉 센서를 갖춘 다양한 기능을 갖는 이동 로봇 시스템이다. Nomad Scout Robot II는 Intel Pentium 233, 64M 램(RAM), 그리고 4GB 하드디스크(Hard Disk)를 갖춘 산업용 보드를 장착하고 있다. Nomad Scout Robot II의 로봇 움직임을 제어 위해 서버는 시리얼라인을 통해 MC68332 제어보드와 통신을 한다.

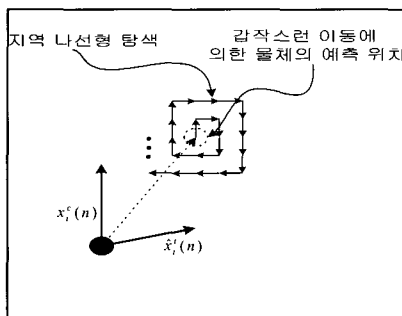


그림 5. 추적목표물의 선형예측과 지역 나선형 탐색.
Fig. 5. Forward linear predictor and local spiral search of the target object.

제어보드는 바퀴의 엔코더에서 측정된 선 속도, 각 속도 그리고 위치 정보와 같은 주행 정보를 제공한다. Nomad Scout Robot II는 이동을 위해 두 개의 독립적인 모터를 장착한 차륜 구동 방식을 사용한다. 따라서 Nomad Scout Robot II는 회전과 병진운동을 할 수 있는 2자유도의 특성을 갖는다. 효과적인 실험을 위해 다음의 그림 6와 같은 GUI를 구성하였다.

GUI는 Pentium III 700MHz가 장착된 MS Window 기반 PC에서 수행된다. 컴파일러는 Visual C++ 6.0을 사용했다. GUI는 카메라에서 얻어지는 영상을 보여주는 영역, SSD 알고리즘의 수행을 시각적으로 보여주는 영역, 이동물체의 템플릿과 이동물체 추적을 실시간으로 보여주는 영역, 그리고 이동물체와 이동로봇의 실제계 좌표상의 궤적과 정보를 제공하는 영역으로 구성된다. 물체의 영상정보를 얻기 위해 흑백용 아날로그 CCD 카메라와 On-Air TV Card를 사용하였고 320x240 크기의 영상을 12~13Hz의 주기로 획득하여 처리하였다.

실험은 그림 7에서처럼 추적목표물이 임의로 움직이는 상황에서 이동로봇이 어떻게 반응하는 지를 분석하는 방법으로 이루어졌다.

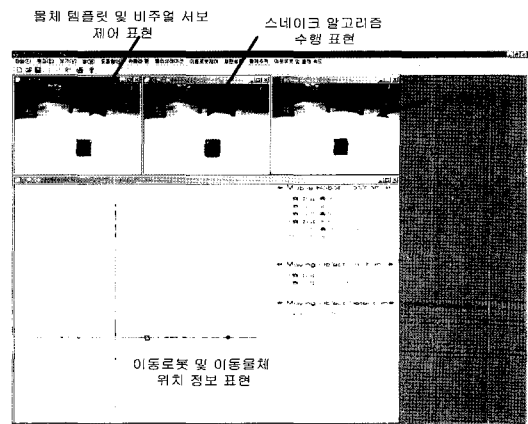


그림 6. 실험용 GUI.
Fig. 6. GUI for experiments.

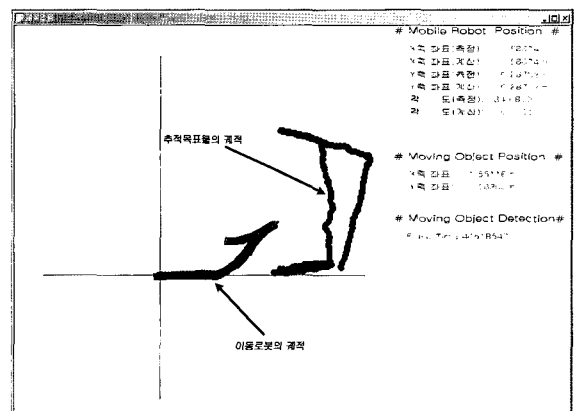


그림 7. 추적목표물과 이동로봇의 궤적.
Fig. 7. Trajectories of the target object and the mobile robot in the X-Y plane of Σ_w .

그림 7은 목표물을 추적하는 동안 추적목표물과 시각서 보 제어에 의한 이동로봇의 위치를 Σ_w 의 X-Y좌표계 상에서 보여준다. 이동물체 추적 시스템은 이동로봇과 목표물 사이의 거리를 0.5~6.0m로 설정할 수 있다. 본 실험은 이동로봇과 목표물 사이의 거리를 1.0m로 설정하였으며 이동로봇이 추적목표물과 같은 방향을 유지하도록 했다. 결과적으로 영상에서 목표 좌표점은 (163,170)으로 주어진다. 그림 8(a)는 추적목표물의 X축 좌표값이 증가한 후 좌표점 (1.41m, 0.09m)에서 X축 좌표값이 감소하는 방향으로 목표물이 움직이는 영상을 보여준다. 그림 8(b)는 좌표점(0.97m, 0.03m)에서 목표물이 다시 X축이 증가하는 방향으로 움직이는 영상을 보여준다. 그림 8(c)는 좌표점(1.37m,0.07m)에서 Y축의 좌표값이 증가하는 방향으로 목표물이 움직이는 영상을 보여준다. 그림 8(d)는 목표물이 Y축 좌표값이 증가하는 방향으로 움직인 후 좌표점(1.38m, 1.07m)에서 약 68° 방향으로 회전하여 X축의 좌표값이 감소하는 방향으로 이동하는 영상을 보여준다. 그림 8(e)는 좌표점(1.07m, 1.16m)에서 이동 물체가 방향을 유지하면서 뒤로 후진하는 영상을 보여준다. 마지막으로 그림 8(f)는 좌표점(1.79m, 0.90m)에서 목표물이 약 85°회전하고 나서 Y축의 좌표값이 감소하는 방향으로 움직이는 영상을 보여준다.

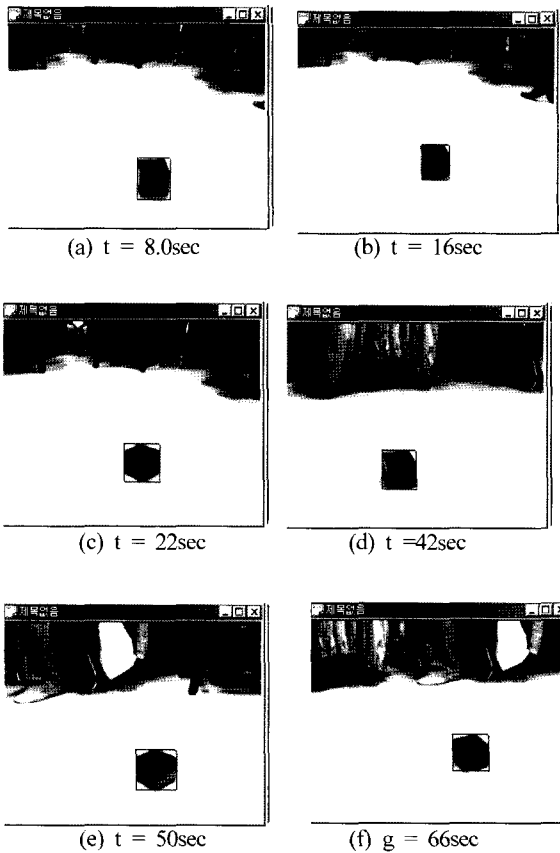
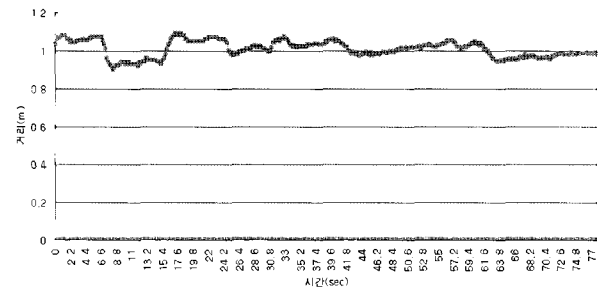


그림 8. 입력영상에서 SSD 알고리즘에 의해 검출된 추적목표물의 위치.

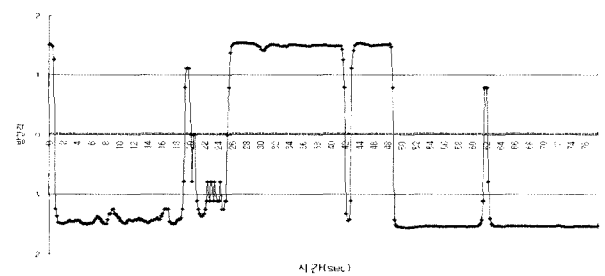
Fig. 8. Position of the target object detected in the input image using the SSD algorithm.

그림 9(a)는 0.2초 간격으로 77.6초 동안 획득한 이동로봇에 대한 추적목표물의 상대적인 거리값을 보여준다. 앞에서 언급했듯이 이동로봇은 추적목표물과 1.0m 거리를 유지되도록 설정되었다. 그림 9(a)를 보듯이 설정 목표 거리값과 에러는 $\pm 0.1m$ 범위에서 유지되었다. 그림 9(b)는 0.2초 간격으로 77.6초 동안 획득한 이동로봇에 대한 추적목표물의 상대적인 방향각을 보여준다. 여기서 목표물을 추적하면서 설정 목표 방향값과 오차는 $-1.55^\circ \sim 1.54^\circ$ 의 범위에서 유지되었다.

그림 10은 추적도중 영상좌표계 상에서 설정된 목표 위치점(163,170)과 측정된 추적목표물의 위치 사이의 오차를 보여준다. 영상좌표계 상에서 설정한 목표 좌표값과 측정된 추적목표물의 위치값 사이의 오차는 0 ~ 7.95 픽셀의 범위를 갖는다. 그림에서 26.6초 부근에서 목표 좌표값 에러가



(a) 이동로봇에 대한 추적 목표물의 상대적인 거리값



(b) 이동로봇에 대한 추적 목표물의 상대적인 방향

그림 9. 이동로봇 및 목표물의 상대적인 거리값과 방향.

Fig. 9. Positional variations of the mobile robot and the target object.

목표 좌표값 에러

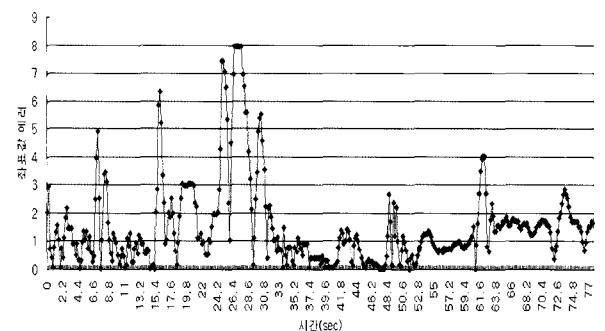


그림 10. 추적목표물의 목표좌표와 측정위치의 오차.

Fig. 10. Error in the measured position of the target object with reference.

상대적으로 큰 이유는 이동로봇이 목표 위치로 수렴하기 전에 추적목표물이 갑작스런 회전을 했기 때문이다. 초기에 수렴이 안된 상태에서 이동물체의 방향전환이 이루어져 상대적으로 에러가 컸으며 77.6초 실험동안 이동물체를 놓친 경우는 없었다.

V. 결론

본 논문은 이동로봇이 이동목표물과의 거리를 일정하게 유지하면서 추적하는 시각서보(visual servo)기법에 관한 것으로 추적하려는 물체의 3차원 위치정보를 정확히 얻는 문제와 이동로봇의 제어변수를 결정하는 방법을 새롭게 제안하였다. 추적대상물의 3차원 위치정보는 하나의 카메라에서 얻은 2차원 영상 한 개만을 이용하여 결정하는 기법을 제안하였다. 이동로봇과 추적하려는 물체가 일반적인 이동로봇의 작업환경에 즉 동일한 평면에서 움직인다는 점을 고려하여 영상좌표계와 이동로봇의 실세계좌표계와의 관계를 단순화하였고 이를 통해 복잡한 연산을 줄일 수 있었다. 또한 이동로봇이 목표물을 정확히 추적하도록 하기 위한 정확한 제어변수의 계산을 위해 영상에서 얻어진 물체의 속도벡터에서 카메라의 이동에 의한 성분을 제거함으로써 순수한 물체의 이동속도를 계산할 수 있게 했다. 이를 기반으로 이동로봇의 입력벡터는 추적 목표물의 위치정보와 목표물의 움직임에 의한 속도 성분을 합하여 계산된다. 실험결과는 제안된 기법의 안정성과 실시간성을 잘 보여주고 있다. 추적대상물의 이동속도를 일정한 범위의 가속도로 변화시키거나 이동방향을 급히 바꾸는 궤적을 만들어 실험한 결과 초당 12프레임의 영상을 처리하면서 추적대상물이 입력영상의 중앙부근에서 벗어나지 않도록 추적하는 결과를 보여주었다. 또한 이동물체를 다른 물체가 순간적으로 가리는 경우에도 잘 추적하고 있음을 보여주었다. 본 논문에서 제시된 추적기법은 추적대상 물체가 한 개이고 입력영상 내에서 이동하는 다른 물체에 관한 고려는 없었다. 따라서 본 논문에서 제시한 연구결과를 토대로 다중 이동물체가 자유로이 얹히는 환경에서 특정한 물체를 추적할 수 있는 방법에 관한 연구가 추후 진행되어야 보다 실용적인 시스템의 구축이 가능해 질 것이다.

참고문헌

- [1] 이부형, 한현수, "하나의 2차원영상에서 표면의 signature를 이용한 다면체의 표현 및 인식 알고리즘," 대한전자공학회는논문지-A, 제34-s권 제2호, pp. 63-70, 1997년 2월.
- [2] S. Z. Li, "Recognizing multiple overlapping objects in image : an optimal formulation," *IEEE Transaction on Image Processing*, vol. 9, no 2, pp.273 - 277, February 2000.
- [3] 한문용, 한현수, "소형(1mm이하) hole의 형태 및 크기 측정을 위한 자동초점 비전검사기," 한국 정밀 공학회지 16권 10호, pp. 80-86, 1999년 10월.
- [4] Y. Tsubaki et al., "All-focused image generation and 3D modeling of microscopic images of insects," *2001. Proceedings. 2001 International Conference on Image Processing*, vol. 2, pp. 197-200, October 2001.
- [5] T. Fujiwara et al., "3D modeling system of human face and full 3D facial caricaturing," *2001. Proceedings. Seventh International Conference on Virtual Systems and Multimedia*, pp. 625-633, October 2001.
- [6] J.-W. Kim et al., "Remote control system using real-time MPEG-4 streaming technology for mobile robot," *2002. ICCE. 2002 Digest of Technical Papers. International Conference on Consumer Electronics*, pp. 200 - 201, June 2002.
- [7] R. C. Luo et al., "An internet-based remote control and monitoring rapid prototyping system," *Industrial Electronics Society, 2001. IECON '01. The 27th Annual Conference of the IEEE*, vol. 1, pp. 159-164, December 2001.
- [8] 정영기, 호요성, "차량의 영상추적을 통한 지능적인 교통정보 추출 알고리즘," *Telecommunications Review*, 제 9권 4호, pp. 582-596, 1999년 7-8월.
- [9] N. P. Papanikolopoulos et al., "Visual tracking of a moving target by a camera mounted on a robot : a combination of control and vision," *IEEE Transaction on Robotics and Automation*, vol. 9, no. 1, February 1993.
- [10] A. C. Sanderson and L. E. Weiss, "Image-based visual servo control using relational graph error signals," *Proc. IEEE*, pp. 1074-1077, 1980.
- [11] V. Cadenat et al., "A control to perform a visually guided tracking task in a cluttered environment," *IEEE Int. Conf. Intelligent Robots and System*, pp. 775-780, 1999.
- [12] A. E. Hunt and A. C. Sanderson, "vision-based predictive tracking of a moving target," *Tech. Carnegie Mellon Univ., The Robotics Inst., Rep. CMU-RI-TR-82-15*, January, 1982.
- [13] R. Kelly, "Robust asymptotically stable visual servoing of planar robots," *IEEE Trans. Robotics and Automation*, vol. 12, no. 5, pp. 759-766, Oct. 1996.
- [14] M. Asada et al., "Adaptive binocular visual servoing for independently moving target tracking," *IEEE Int. Conf. Robotics and Automation*, vol. 3, pp. 2076-2081, April. 2000.
- [15] W. Chen. U. Korde, and S. Skaar, "Position control experiments using vision," *International Journal of Robotics Research*, vol. 13, pp. 199-208, June 1994.
- [16] R. C. Luo et al., "An adaptive robotic tracking system using optical flow," in *Preceeding of IEEE International Conference Robotics Automation*, pp. 568-573, 1988.
- [17] C. Balkenius and L. Kopp, "Visual tracking and target selection for mobile robots," *Proceedings of the First Euromicro Workshop on Advanced Mobile Robot*, pp. 166 -171, 1996.
- [18] A. Dev et al., " 'Where are you driving to?' heading

direction for a mobile robot from optical flow", *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1578-1588, 1998.

[19] A. Castano and S. Hutchinson, "Visual compliance: task-directed visual servo control," *IEEE Trans. Robotics and Automation*, vol. 10, no. 3, pp. 334-342, June. 1994.



한 영 준

1996년 숭실대학교 전자공학과 졸업, 동대학 석사 졸업(1998), 동대학원 박사졸업(2003), 2003년 9월 ~ 현재 숭실대학교 정보통신전자공학부 전임강사, 관심분야는 비주얼 서보잉, 이동로봇, 로봇센서, 영상처리.

한 현 수

제어·자동화·시스템공학 논문지 제 1 권 제 2 호 참조