

# 전방향 구동 로봇에서의 비전을 이용한 이동 물체의 추적

## Moving Target Tracking using Vision System for an Omni-directional Wheel Robot

김 산, 김 동 환\*

(San Kim and Dong Hwan Kim)

**Abstract :** In this paper, a moving target tracking using a binocular vision for an omni-directional mobile robot is addressed. In the binocular vision, three dimensional information on the target is extracted by vision processes including calibration, image correspondence, and 3D reconstruction. The robot controller is constituted with SPI(serial peripheral interface) to communicate effectively between robot master controller and wheel controllers.

**Keywords :** omni-directional robot, binocular vision, SPI communication, moving target tracking

### I. 서론

지능형 로봇은 자율주행이 필수기능이며 동시에 빠른 기동성이 구비되어야 한다. 빠른 기동성을 위해서는 기존의 두 바퀴의 전후방 직진 이동 방식으로는 한계가 있다. 그러므로 임의의 방향으로 최소한의 회전 반경을 가지면서 이동이 가능한 전 방향바퀴(omni-directional wheel) 로봇[1-3]이 기동성 확보에 상당히 효과적이다. 더불어 지능형 로봇은 복잡한 환경에서도 빨리 목표물을 인식하고 추적할 수 있는 기능이 구비되어야 한다. 현재 3륜 구조를 갖고 바퀴와 롤러가 90도 각도로 부착된 로봇을 개발된 것뿐만 아니라 4륜 구조에 바퀴와 롤러가 45도 각도로 부착된 로봇도 개발되고 있다. 또한 바퀴의 배치와 상관없이 로봇의 안정성을 확보할 수 있도록 함으로써 속도비의 범위를 대폭 확장할 수 있는 가변 바퀴 배치[4]가 연구되고 있다.

전 방향 로봇을 구성하는 요소로는 전 방향 구동 메카니즘, 제어기, 그리고 물체의 정보를 구하는 비전 시스템이 있다. 현재 비전 시스템 중 하나인 물체의 3차원 정보를 구하는 스테레오 비전이 이동 로봇에 적용되고 있다[5,6].

본 논문에서는 우수한 기동성 확보와 목표물 인식을 위한 방법으로 전 방향 바퀴와 쌍안 카메라(binocular vision)의 비전처리 시스템을 제시하고, 전 방향 구동 로봇의 메카니즘과 운동제어 방안을 제시한다. 또한 쌍안 카메라를 이용한 이동 물체의 3차원 정보를 획득할 수 있는 비전처리 방안을 기술한다. 최근 쌍안 카메라는 자동 주행, 항공지도 작성, 모델기반 이식, 산업자동화 시스템 등에 응용되고 있으며, 3차원 모델링 시스템에 이용하는 방법[7], 이동객체 추출 및 거리측정[8] 등이 연구되고 있다. 본 논문에서는 전 방향 구동 로봇과 쌍안 카메라를 통합하는 새로운 시스

템을 제안하고 실제 적용의 효용성을 검증함으로써, 자율주행로봇의 주행 능력을 향상시키는데 목적을 두었으며, 쌍안 카메라를 이용한 3차원 복원 기법에 있어서 두 카메라를 평행하게 위치시킨 후에 적용하는 삼각측정기법(simple stereo system)[11-13]의 한계를 극복한, 새로운 3차원 복원 기법의 적용에 그 의의가 있다.

### II. 로봇 메카니즘 및 운동 기구학

전 방향 구동 로봇은 2차원 평면에서 3자유도( $x, y, \theta$ )의 운동이 가능해서 바퀴의 자세전환 없이 임의의 방향으로 주행이 가능한 로봇을 말한다. 그러므로 전 방향 구동 로봇은 기동성과 회전성을 모두 확보할 수 있는 장점을 갖고 있다. 본 논문에서는 3개의 전 방향 바퀴를 사용하여 전 방향 구동 로봇을 구현하였으며 각 바퀴를 설계 제작하였다. 전 방향 구동로봇은 전후좌우(十자) 방향으로 움직일 수 있는 특수 구조의 바퀴를 사용하여 일반적인 바퀴의 형태로선 움직일 수 없는 사각지대로의 이동을 가능하게 하였으며, 바퀴를 120° 간격으로 배치하는 삼륜 구동 방식을 채택하였다. 그림 1은 본 논문에서 제시한 전 방향 구동 로봇의 3-D 모델링 이미지이며 로봇의 사양은 표 1과 같다.

120° 간격으로 배치된 전 방향 바퀴를 원하는 이동속도( $v$ )와 방향( $\theta$ )으로 구동하기 위하여 각 바퀴별 요구되는 각속도에 대한 해석을 기술한다. 그림 2의  $v$ 는 전 방향 바퀴 로봇의 이동 속도이고  $\theta$ 는  $x$ 축으로부터의 중심 속도방

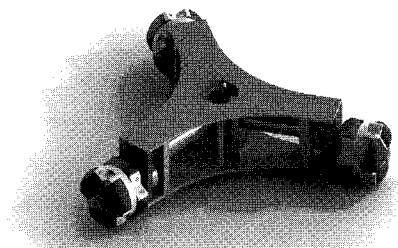


그림 1. 전방향 구동 로봇의 3D 모델링.

Fig. 1. 3D modeling of omni directional robot.

\* 책임저자(Corresponding Author)

논문접수 : 2008. 3. 13., 채택확정 : 2008. 7. 14.

김 산 : 서울산업대학교 산업대학원 메카트로닉스공학과  
(skcjojo@hotmail.com)

김동환 : 서울산업대학교 기계설계자동화공학부(dhkim@snut.ac.kr)

※ 이 연구는 서울시 산학연 협력사업의 지원으로 수행되었음(과제 번호 10583).

표 1. 전방향 구동 로봇의 명세.

Table 1. Specifications of omni directional robot.

Material	AL 6061
	MC nylon
	PVC resin
Size (Diameter×Height)	450mm(diameter)×90mm(height)
Weight	3.5kgf
DC Motors	DC servo motor TRE series 30W×3EA (TAMAGAWA motors)

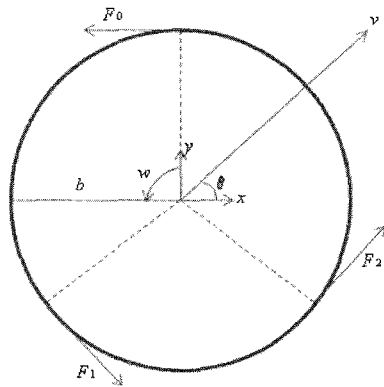


그림 2. 전방향 구동 로봇의 좌표계와 운동학.

Fig. 2. Coordinates and kinematics of omni directional robot.

향의 각도를 나타낸다.  $b$ 는 중심으로부터 바퀴중심까지의 거리를 나타내며  $v_0, v_1, v_2$ 는 각 바퀴의 선속도이다.  $F_0, F_1, F_2$ 는 각 바퀴의 단위 방향 벡터를 나타내며 그 값은 아래와 같이 결정된다.

$$\begin{aligned}
 F_0 &= [-1, 0] \\
 F_1 &= [\cos 60, -\cos 30] = \left[ \frac{1}{2}, -\frac{\sqrt{3}}{2} \right] \\
 F_2 &= [\cos 60, \cos 30] = \left[ \frac{1}{2}, \frac{\sqrt{3}}{2} \right]
 \end{aligned} \tag{1}$$

위의 단위 벡터를 사용하여 로봇 중심의 속도의  $x, y$  성분은 다음과 같이 결정된다.

$$\begin{aligned}
 v_x &= F_{0x}v_0 + F_{1x}v_1 + F_{2x}v_2 \\
 v_y &= F_{0y}v_0 + F_{1y}v_1 + F_{2y}v_2
 \end{aligned} \tag{2}$$

따라서

$$\begin{aligned}
 v_x &= -v_0 + (\cos 60^\circ)v_1 + (\cos 60^\circ)v_2 = v \cos \theta \\
 v_y &= (-\cos 30^\circ)v_1 + (\cos 30^\circ)v_2 = v \sin \theta
 \end{aligned} \tag{3}$$

그러므로 각 바퀴의 선속도 값은 (4)로 결정된다.

$$\begin{aligned}
 v_0 &= -v \cos \theta \\
 v_1 &= v \cos \theta \cos 60^\circ - v \sin \theta \cos 30^\circ \\
 v_2 &= v \cos \theta \cos 60^\circ + v \sin \theta \cos 30^\circ
 \end{aligned} \tag{4}$$

(4)로부터 로봇의 중심 속도와 방향이 결정되면 각 바퀴가 제공해야할 선속도가 결정되고 이 선속도로부터 바퀴의 각 속도가 결정된다.

### III. 제어 시스템

#### 1. 제어 시스템 개요

전 방향 구동 로봇의 제어구조는 전체적으로 상위제어기와 하위제어기로 구분된다. 상위제어기는 개인용 컴퓨터로 구성되며, 하위제어기는 마이크로 컨트롤러(TMS320LF2407A)로 구성되었다. 상위제어기는 로봇에 장착되어있는 두 대의 CCD 카메라의 영상을 획득하고 영상 처리를 통하여 카메라의 내부 및 외부 변수를 보정한 후, 영상을 분석한다. 분석된 영상 데이터는 대상 물체의 3차원 데이터를 복원하는데 이용된다. 상위제어기는 복원된 대상 물체의 3차원 데이터를 이용하여 대상 물체의 위치와 속도를 계산한 후 하위제어기에 RS-232C 통신을 이용하여 이를 전송하게 된다(그림 3).

하위제어기는 상위제어기로부터 전송받은 대상 물체의 위치, 속도 데이터를 이용하여 120° 간격으로 배치되어 독립적으로 구동되는 각 바퀴의 속도, 방향을 계산한 후 이를 각 바퀴에 장착되어 있는 모션제어기에 SPI(Serial Peripheral Interface)통신을 이용하여 전송한다(그림 3). 전방향 구동 로봇은 마스터(master)와 슬레이브(slave)의 구조로 되어있으며, 마스터는 상위제어기와 RS-232C를 이용하여 통신을 하고, 상위제어기에서 전송되어진 대상 물체의 위치와 속도 데이터를 분석하여 120° 간격으로 배치된 각 바퀴의 방향, 위치, 속도지령을 SPI를 이용하여 슬레이브에 전송하게 된다(그림 4). 독립적으로 구성된 각각의 슬레이브는 1축 모션제어기로서 마스터에서 전송된 바퀴의 방향, 위치, 속도 지령에 대하여 연결된 바퀴를 제어하는 역할을 한다. 마스터와 슬레이브간의 통신 방식인 SPI는 최대 500kbps의 전송 속도를 낼 수 있기 때문에 마스터에서 슬레이브에 지령을 전송할 때 생기는 시간지연을 최대한 줄일 수 있는 장점이 있으며, 1:n (n은 양의 정수)의 네트워크 구조를 형성 할 수 있는 장점을 지니고 있다.

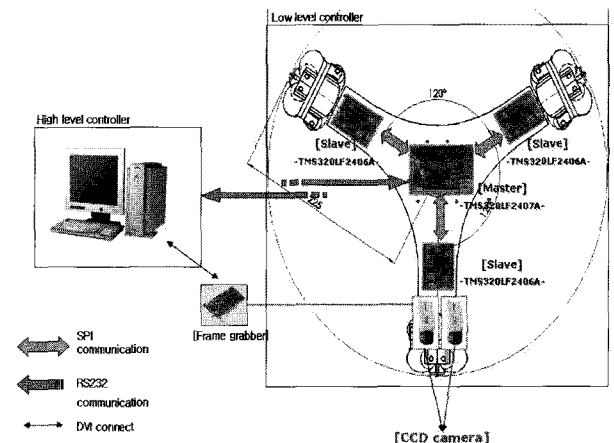


그림 3. 시스템 구성.

Fig. 3. System configuration.

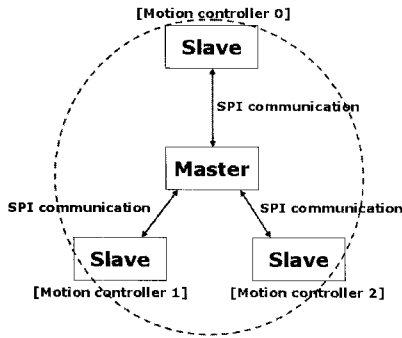


그림 4. 마스터와 슬레이브들 간의 네트워크 구성도.  
Fig. 4. Networking architecture between master and slaves.

전 방향 구동(omni drive)을 위해서 개발한 마스터는 TMS320LF2407A를 이용하였다. 슬레이브는 TMS320LF2406A[9]를 이용하였고, PID 제어를 내장하고 있으며 속도 및 위치 제어가 가능하다.

IV. 비전처리

1. 쌍안 카메라 시스템

쌍안 카메라는 비접촉식 방법 중 하나로 궁극적 목표는 인간 시각 시스템의 기능을 컴퓨터로 구현하는데 있으며, 이를 구현하기 위해서는 입력된 영상정보로부터 3차원 물체를 찾아내거나 인식하는 것이 가장 기본적인 과제라 할 수 있다. 3차원 물체 인식에 있어서 2차원 영상으로부터 3차원 정보를 추출하는 과정은 매우 중요한 단계로서 한 장의 영상으로부터 깊이 정보를 추출하는 단안시법과 두 장 이상의 영상으로부터 깊이 정보를 추출하는 복안시법이 있는데 보통 후자를 쌍안 카메라라고 한다. 쌍안 카메라는 목표물까지의 거리를 인지함으로써 자율주행로봇의 차후 행동에 대한 의사 결정을 할 수 있으므로 자율주행능력, 객체판별 등의 인공 지능을 향상하는데 중요한 부분으로 주목받고 있다.

본 논문에서 수행한 객체의 3차원 정보를 계산하는 과정은 다음과 같다. 첫 번째 단계로서, 보정판에 대하여 두 대의 카메라 영상을 추출하고, 카메라의 내부 및 외부 파라미터를 구하는 카메라 보정(camera calibration)을 한다. 두 번째 단계로서, 두 대의 카메라 영상에서 일치하는 화소를 찾는 이미지 정합(image correspondence)을 수행하였고 최종 단계로서, 정합 정보를 이용하여 3차원 복원(3D reconstruction)을 한다.

2. 카메라 보정(camera calibration)

카메라 보정을 통하여 물체와 카메라사이의 직선이동거리와 회전각도를 알아내는 외부 파라미터와 카메라가 가지는 초점거리 등의 내부 파라미터를 추출해 낼 수 있다. 쌍안 카메라를 채용했으므로 좌우 카메라 각각 보정이 이루어져야 한다. 카메라 보정을 나타내는 식은 다음과 같다.

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \{ {}^I P_c \} \{ {}^c M_r \} \begin{pmatrix} {}^r X \\ {}^r Y \\ {}^r Z \\ 1 \end{pmatrix} = H \begin{pmatrix} {}^r X \\ {}^r Y \\ {}^r Z \\ 1 \end{pmatrix} \quad (5)$$

여기서  $s$ 는 크기계수(scaling factor)이다.  $(u, v)$ 는 카메라 이미지좌표의  $(x, y)$ 와 이에 대응하는 인덱스 좌표의 가로와 세로축의 값이다.  $({}^r X, {}^r Y, {}^r Z)$ 는 기준좌표에서 물체까지의 거리값이다.  $H$ 가 보정행렬(calibration matrix)이며 여기에는 카메라 내부 파라미터정보  ${}^I P_c$ 와 카메라와 물체간의 회전 및 직선 이동량을 표현하는 외부 파라미터 정보  ${}^c M_r$ 이 포함되어 있다. (5)를 정리하면 (6)과 같다.

$$\begin{pmatrix} s u \\ s v \\ s \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & 1 \end{pmatrix} \begin{pmatrix} {}^r X \\ {}^r Y \\ {}^r Z \\ 1 \end{pmatrix} \quad (6)$$

보정행렬  $H$ 의 요소는 미지수가 11개 이므로 보정 시, 카메라 인덱스 좌표에서 6쌍을 골라 여기에 해당되는 물체의 3차원 정보를 사전에 입력하여 보정 행렬을 다음과 같이 구한다. 보정행렬의 각 성분을 벡터  $V$ 로 하면 그 값은 6개의 쌍으로 구성된 값을 이용하여 다음과 같이 결정된다.

$$V = [h_{11} \ h_{12} \ h_{13} \ h_{14} \ h_{21} \ h_{22} \ h_{23} \ h_{24} \ h_{31} \ h_{32} \ h_{33}]^T \quad (7)$$

로 정의하면 보정 행렬은 다음과 같이 구해진다.

$$V = (A^t A)^{-1} (A^t B) \quad (8)$$

여기서  $A$ 는 아래와 같이 6개의 쌍으로 구성된 인덱스 좌표 값과 기준좌표에서 물체까지의 거리 값으로 구성된 행렬이며  $B$ 는 6개의 쌍의 좌표이다[10].

$$A = \begin{bmatrix} {}^r X & {}^r Y & {}^r Z & 1 & 0 & 0 & 0 & 0 & -({}^r X u_n) - ({}^r Y u_n) - ({}^r Z u_n) \\ 0 & 0 & 0 & 0 & {}^r X & {}^r Y & {}^r Z & 1 & -({}^r X v_n) - ({}^r Y v_n) - ({}^r Z v_n) \end{bmatrix}$$

$(n = 1, 2, 3, 4, 5, 6),$

$$B = \begin{bmatrix} u_n \\ v_n \end{bmatrix}, \quad n = 1, 2, 3, 4, 5, 6) \quad (9)$$

이 때 구해진  $V$ 의 열이 보정행렬  $H$ 의 해가 된다.

3. 스테레오 이미지정합(sterео image correspondence)

스테레오 이미지 정합은 절대 좌표상의 한 점에 대한 좌우측 카메라의 미지들 간의 점을 대응시키는 과정이다. 스테레오 이미지 정합 방법은 크게 영역기반 정합 방법(correlation based method)과 특징기반 정합 방법(feature based method)으로 나눌 수 있다. 영역기반 정합을 이용한 방법들에 대하여 살펴보면 Moravec[14]은 깊이 정보를 구하여 자동 주행 장치에 이용하였고, Mori등[15]은 영역기반 정합방법에서 영상 취득구조 등에 의한 왜곡현상(영상 밝기 또는 대비 등의 변화)의 문제점을 개선하고 반복적 예측과 교정을 하는 계층적 정합 방법을 제안했으며 항공영상에 적용하였다. Quam[16]은 각 계층별로 영상의 해상도를 높이면서 각 계층에서 구한 변위정보를 다음 계층에서 이용하여 지형왜곡의 보상, 정합, 보간 등에 이용하는 자동 스테레오 정합 방법을 제안하여 영역기반 정합의 성능을 향상시키고자 하였다. Ikeuchi[17]는 photometric 스테레오 시스템에서 구한 두 쌍의 면방향 지도(surface orientation map)에서 방향 성분이 같은 영역들을 분할하여 이들 영역간의

정합을 하여 깊이 정보를 구하는 방법을 제안하였다. 그는 분할된 영역의 중심에서 깊이 정보를 구하고 영역의 나머지 점들의 깊이 정보는 photometric 스테레오로부터 구한 방향 정보를 적분하여 구함으로써 연속된 깊이 정보를 구하였다. 특징기반 정합 방법의 선행연구에 대해서 살펴보면 Grimson[19]은 Marr와 Poggio[18]의 이론을 실제로 구현하여 LoG(Laplace of Gaussian)함수의 창을 이용하여 특징점을 추출하였다. Baker와 Binford[20]와 Ohta와 Kanade[21]는 epipolar line 구속조건과 에지들 사이의 순서 구속 조건을 사용하여 여기에 동적계획(dynamic programming) 기법을 적용하여 올바른 정합을 하고자 하였다. Lloyd 등[22]은 epipolar 구속조건을 사용하여 좌우 영상의 같은 라인 상의 특징들을 정합 시에는 동적 계획 기법을 사용하고 epipolar line 들 사이의 특징에 대해서는 이완 기법을 사용하는 방법을 제안하였다. 영역기반 정합 방법에서는 이상적으로 개개의 화소에 대해서 다른 쪽 영상에서 상응하는 화소를 정합하는 것이 바람직하지만 화소 단위 정합에서 발생하는 모호성을 해결하는 것이 어려우므로 화소들이 이루는 의미 있는 영역들 간의 상관관계를 이용하여 정합하는 방식을 사용한다. 특징기반 정합 방법은 영상의 모든 화소에 대해 정합을 하지 않고 영상의 밝기 정보로부터 추출된 심볼특징(symbolic feature)을 이용한다. 이러한 심볼특징에는 에지, 끝점, 모서리 및 특별한 연산자에 의해 선택된 점 등이 있으며 이러한 특징들은 영상 자체의 밝기보다는 밝기의 변화에 관련됨으로써 영상의 물리적 변화를 잘 나타낸다.

본 논문에서는 특징기반 정합 방법을 수행하였다. 대상 물체의 특징을 검출하기 위하여 임계값을 이용한 이진화 기법 후 1차 미분을 하여 윤곽선을 추출하였다. 그림 5와 같이 왼쪽 카메라 이미지의 왼쪽에서 오른쪽으로 스캐닝을 하다가 값이 0인 검정색 화소를 만나면 그 화소에 시작점을 표시한다. 시작점을 표시한 후 위에서 아래, 왼쪽에서 오른쪽, 아래에서 위, 오른쪽에서 왼쪽 방향으로 각각의 길이를 저장하며 스캐닝 하게 된다. 스캐닝을 할 때에는 관심 화소의 주변 다섯 개 화소를 같이 스캐닝 하는데, 그 이유는 윤곽선의 일그러짐 때문이다. 이 때 주변 다섯 개 화소란 스캐닝 시 현재 좌표와 좌, 우 또는 위, 아래 두 개 화소를 포함한 다섯 개의 화소를 의미한다. 예를 들어, 현재

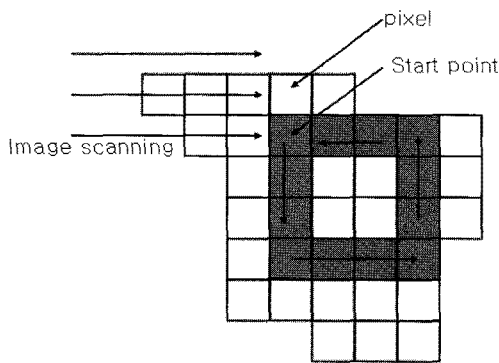


그림 5. 특징기반 정합을 위한 이미지 스캔 방향.  
Fig. 5. Image scan direction for feature based correspondence.

스캐닝의 방향이 위에서 아래 방향이고 화소의 좌표가  $(x,y)$ 일 때  $(x-2,y), (x-1,y), (x,y), (x+1,y), (x+2,y)$ 인 다섯 개 화소를 의미하며 현재 스캐닝의 방향이 왼쪽에서 오른쪽이고 화소의 좌표가  $(x,y)$ 일 때 다섯 개 화소는  $(x,y-2), (x,y-1), (x,y), (x,y+1), (x,y+2)$ 가 된다. 검출된 윤곽선은 렌즈에 왜곡으로 인하여 완벽한 직선이 되지 않기 때문에 이와 같은 스캐닝 방법을 이용하면, 대상물체의 검출에 있어 왜곡에 강인하게 대응할 수 있다. 스캐닝은 시작점과 현재 스캐닝하고 있는 화소가 만났을 때 끝나며, 이 때 저장된 각 선분의 가로, 세로의 길이 차이가 5픽셀 이하라면 직사각형이라고 판단한다. 직사각형이라고 판단이 되면, 가로와 세로의 비율을 계산한다.

본 논문에서는 대상 물체의 가로와 세로의 비율을 데이터베이스화 하였다. 스캐닝 과정에서 계산된 직사각형의 가로와 세로의 비율을 저장되어있는 대상 물체의 가로, 세로의 비율과 비교한 후 그 값이 일치하면 대상 물체라고 인식한다. 대상 물체를 인식하면 4곳의 모서리 좌표를 저장한 후 직사각형의 중심 좌표를 구한다.

왼쪽 카메라에서 대상 물체를 검출하면 오른쪽 카메라도 마찬가지로의 방법으로 스캐닝 과정을 거친다. 스캐닝 과정이 끝나면 왼쪽 카메라와 오른쪽 카메라는 정합된 4개의 점 데이터를 얻을 수 있다.

4. 3차원 좌표 복원(3D reconstruction)

인덱스 프레임상의 두 점  $(u_1, v_1), (u_2, v_2)$ 를 이용하여 기준 프레임 상의 물체의 3차원 거리 정보  $(^rX, ^rY, ^rZ)$ 을 알아내는 과정을 역투영 맵핑(Inverse projective mapping) 이라고 한다. 본 논문에서 구현한 역투영 맵핑은 왼쪽과 오른쪽의 두 카메라의 위치가 평행하도록 위치시키지 않아도 기준프레임상의 점을 3차원 복원 할 수 있다는 장점이 있다. 필요조건으로는 왼쪽과 오른쪽 카메라의 교정행렬  $H$ 를 각각 구해야만 한다. 그림 6의 왼쪽 카메라의 점  $a$ 는  $(u_1, v_1)$ , 오른쪽 카메라의 점  $b$ 는  $(u_2, v_2)$ 로 인덱스 프레임에서 표현되며, 이 데이터와 교정프레임과의 관계식은 (10)과 (11)으로 나타낼 수 있다.

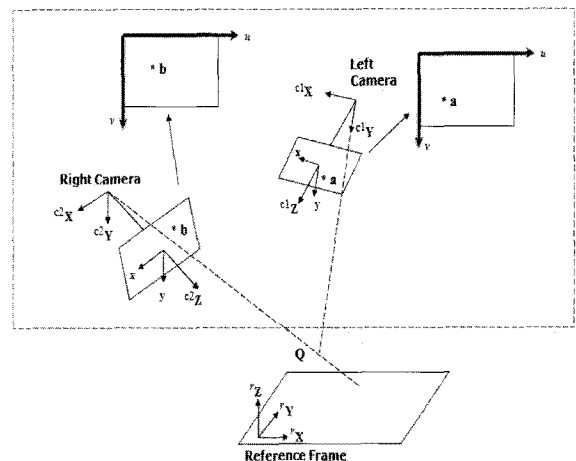


그림 6. 쌍안카메라의 기하학적 투영 모델.  
Fig. 6. Illustration of the geometric projection in binocular vision.

$$s_1 \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix} = H_1 \begin{pmatrix} {}^r X \\ {}^r Y \\ {}^r Z \\ 1 \end{pmatrix} \quad (10)$$

$$s_2 \begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix} = H_2 \begin{pmatrix} {}^r X \\ {}^r Y \\ {}^r Z \\ 1 \end{pmatrix} \quad (11)$$

여기서 왼쪽 카메라와 오른쪽 카메라의 보정행렬은 각각 아래와 같이 정의할 수 있다.

$$H_1 = \begin{pmatrix} i_{11} & i_{12} & i_{13} & i_{14} \\ i_{21} & i_{22} & i_{23} & i_{24} \\ i_{31} & i_{32} & i_{33} & 1 \end{pmatrix}, H_2 = \begin{pmatrix} j_{11} & j_{12} & j_{13} & j_{14} \\ j_{21} & j_{22} & j_{23} & j_{24} \\ j_{31} & j_{32} & j_{33} & 1 \end{pmatrix} \quad (12)$$

(10), (11)에서 크기계수(scaling factor)인  $s_1$ 는 (13)으로 표현되며, (13)으로부터  $s_1$ 를 구할 수 있다. 마찬가지로의 방법으로  $s_2$ 을 구할 수 있게 된다. 이와 같은 방법으로 서로 다른 크기계수를 구할 수 있다.

$$\begin{cases} s_1 u_1 = i_{11} {}^r X + i_{12} {}^r Y + i_{13} {}^r Z + i_{14} \\ s_1 v_1 = i_{21} {}^r X + i_{22} {}^r Y + i_{23} {}^r Z + i_{24} \\ s_1 = i_{31} {}^r X + i_{32} {}^r Y + i_{33} {}^r Z + 1 \end{cases} \quad (13)$$

기준프레임의  $({}^r X, {}^r Y, {}^r Z)$ 는 (11)과 (12)를 사용하면 (14)로 정의할 수 있으며 연고자하는 해를 구할 수 있게 된다[10].

$$\begin{pmatrix} {}^r X \\ {}^r Y \\ {}^r Z \end{pmatrix} = (A_1^t A_1)^{-1} (A_1^t B_1) \quad (14)$$

여기서

$$A_1 = \begin{pmatrix} i_{11} - i_{31}u_1 & i_{12} - i_{32}u_1 & i_{13} - i_{33}u_1 \\ i_{21} - i_{31}v_1 & i_{22} - i_{32}v_1 & i_{23} - i_{33}v_1 \\ j_{11} - j_{31}u_2 & j_{12} - j_{32}u_2 & j_{13} - j_{33}u_2 \\ j_{21} - j_{31}v_2 & j_{22} - j_{32}v_2 & j_{23} - j_{33}v_2 \end{pmatrix}$$

$$B_1 = \begin{pmatrix} u_1 - i_{14} \\ v_1 - i_{24} \\ u_2 - j_{14} \\ v_2 - j_{24} \end{pmatrix} \quad (15)$$

이다.

5. 비전 시스템 구성 및 프로그램

시스템의 구성은 CCD 카메라 두 대, 영상 획득용 프레임 그레이버(frame grabber)와 영상 획득과 이미지 처리 프로그램 작성을 위한 PC로 되어있다(그림 7).

카메라는 SONY XC-HR50이며, 해상도 659×494(horizontal/vertical), 한 픽셀의 실제 크기 7.4×7.4μm, 칩 크기는 5.84×4.94mm 이다. 카메라 렌즈는 초점 거리 8mm, F1.3이며 수동식이다. 프레임 그레이버는 Matrox사의 Meteor-II 보드로서 최대 지원 해상도 640×480, 속도 30 frame/sec 이다. PC는 AMD 3000을 사용하였고, Visual C++6.0으로 프로그램을 작성하였다.

프로그램은 카메라 보정, 이미지 정합, 3차원 복원의 3단

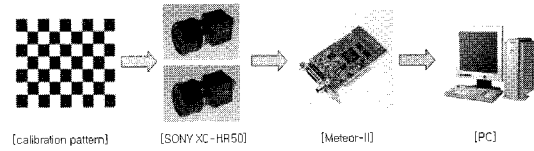


그림 7. 쌍안 카메라 시스템의 구조. Fig. 7. Structure of binocular vision system.

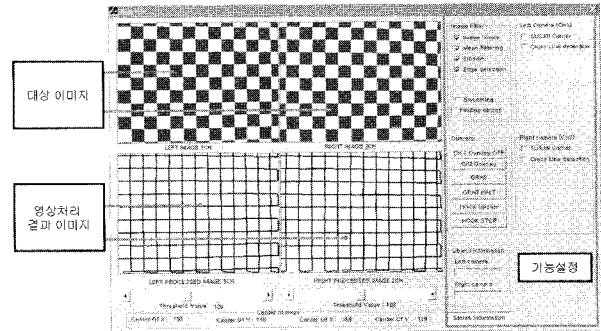


그림 8. 쌍안카메라 프로세스의 메인 프로그램 윈도우. Fig. 8. Main program window for binocular vision process.

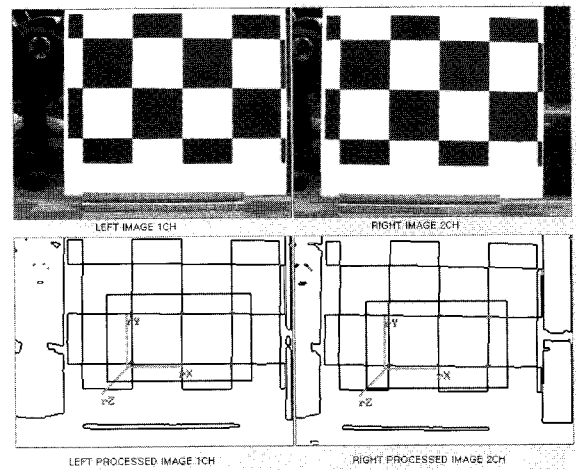


그림 9. 코너 추출 결과. Fig. 9. Conner extraction result.

계로 구성된다. 그림 8에서 볼 수 있듯이 상단의 두 이미지는 왼쪽과 오른쪽 카메라의 원이미지이며, 하단의 두 이미지는 영상처리 결과 이미지를 확인 할 수 있도록 하였다. 왼쪽과 오른쪽 카메라의 보정 데이터(camera calibration data)를 확인하는 창은 별도로 구성되어 있다.

6. 실험 결과

그림 9는 카메라 보정을 위해서 보정패턴(calibration pattern)의 코너를 추출하여 그 데이터를 표현한 것이다. 이때 코너를 추출하기 위한 검색 영역은 사용자가 마우스로 지정할 수 있도록 하였다. 영상 처리 결과 윈도우의 파랑색 부분은 코너를 추출하기 위한 검색 영역을 나타내며 녹색의 십(+)자 선은 기준프레임의 중심점(0,0)을 확인하기 위하여 표현한 것이다. 추출된 코너는 빨강색 십(+)자 선으로 표현하였다. 보정패턴의 크기는 100mm×100mm 크기의

정사각형이다.

카메라 보정의 수행결과 보정행렬  $H_1, H_2$ 는 다음과 같았다.

$$H_1 = \begin{pmatrix} 3.6561 & 2.5600 & 0.0000 & -146.1665 \\ 0 & 2.5600 & -3.6561 & 987.2118 \\ 0 & 0.0100 & 0.0000 & 1.0000 \end{pmatrix}$$

$$H_2 = \begin{pmatrix} 3.6561 & 2.5600 & 0.0000 & -328.9694 \\ 0 & 2.5600 & -3.6561 & 987.2118 \\ 0 & 0.0100 & 0.0000 & 1.0000 \end{pmatrix} \quad (16)$$

그림 10은 대상 물체의 좌우측 이미지의 정합된 좌표를 나타낸다. 대상 물체의 모서리는 파랑색의 십(+)자선으로 표현되며, 대상 물체의 중심은 빨강색의 십(+)자선으로 나

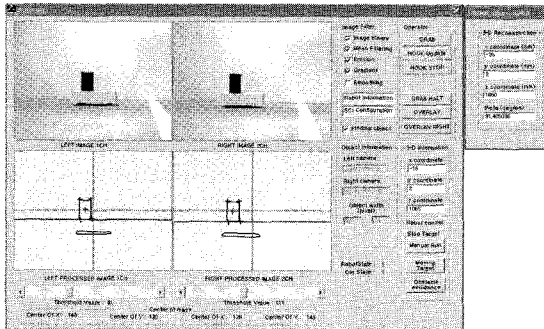


그림 10. 스테레오 이미지 정합 및 3차원 좌표복원.  
Fig. 10. Stereo image correspondence and 3-D reconstruction.

표 2. 3차원 좌표복원 결과.

Table2. Result of 3-D reconstruction.

Real distance (mm)				Measured distance (mm)			
x	y	z	$\theta$ (degree)	x'	y'	z'	$\theta'$ (degree)
0	-15	272	90	11	-11	305	87.9
0	-15	322	90	18	-10	355	87.0
0	-15	422	90	14	-9	448	88.2
0	-15	522	90	4	-7	527	89.5
0	-15	622	90	4	-6	623	89.6
0	-15	722	90	-1	-2	694	90.0
0	-15	822	90	11	1	760	89.1
0	-15	922	90	2	3	839	89.8
0	-15	1022	90	22	-1	902	88.6

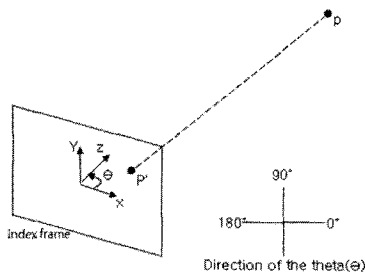


그림 11. 월드좌표계.  
Fig. 11. World coordinates.

타냈다. 또한 빨강색 십(+)자선으로 나타난 대상 물체의 중심점에 대하여 3차원 복원한 결과도 나타나고 있다.

3차원 좌표의 복원의 결과는 표 2의 측정 결과를 통해 나타내며, 측정 결과의  $x, y, \theta$ 는 그림 11과 같다.

7. 실험 결과 고찰

본 논문은 카메라 보정에서 이미지 정합 그리고 3차원 좌표 복원까지 쌓인 카메라의 모든 과정을 수행하였다. 대상 물체의 중심점을 3차원 복원 한 결과  $x$ 좌표의 측정 오차는 -1mm~22mm로 나타났으며  $y$ 좌표의 측정 오차는 4mm~18mm로 나타났다. 대상 물체의 깊이 정보인  $z$ 좌표의 측정 오차는 5mm~120mm로 나타났다. 특히  $z$ 방향으로 멀어질수록 오차가 커지는 것으로 나타났으며 이는 카메라가 멀리 떨어질 경우 이미지의 왜곡이 더 크게 되기 때문이다. 이 경우 로봇을 비교적 목표물과 근거리에서 배치시킨 후 3차원좌표 복원을 하는 것이 효율적이다. 카메라와 대상 물체 사이의 방향 정보를 측정하기 위한  $\theta$ 측정 오차는  $0.08^\circ \sim 3.0^\circ$ 로 나타났다. 실험 결과 측정 오차의 발생 원인은 측정시의 발생하는 오차와 렌즈의 왜곡에 의한 오차, 그리고 4.4장에서 언급한 변위의 오차 때문인 것으로 사료된다.  $120 \times 100 \times 130(\text{mm})$  크기의 이동 물체의 추적을 위하여 필요한 허용 오차를  $\pm 50\text{mm}, \pm 10^\circ$ 로 설정하였을 때,  $x, y, \theta$ 는 신뢰할 수 있음을 확인하였고,  $z$ 방향의 경우 로봇과 측정된 물체와의 거리가 800mm 이하일 때 신뢰할 수 있었다. 그러므로 로봇과 측정된 물체와의 거리가 800mm 이하일 때를 '거리 측정 신뢰구간'으로 설정하여 이동 물체를 추적하는데 그 데이터를 사용하였다.

V. 이동물체 추적 실험 결과

이동물체에 대한 거리 정보는 실시간으로 로봇에게 알려져야 하며 또한 이동물체의 속도 및 방향 정보가 알려져야 로봇이 추적을 하게 된다. 전 방향 로봇이 쌍안카메라에 비치는 물체의 변화를 인지하고 각 위치에 대한 3차원 정보를 추출한 후 변화된 위치에서의 3차원 정보와의 차이를 계산하여 물체의 이동속도 및 각도를 계산한다. 카메라로 물체를 촬영한 후 정해진 짧은 시간 후에 다시 촬영을 하여 물체의 3차원 정보가 변한 양을 계산한다. 실험에서는 목표물이 동일한 평면에 있는 것으로 하여 실시하였으며 전 방향 로봇의 이동이  $(x, z)$ 평면상에서 이동하는 것으로 하여 목표물의 속도 및 방향을 계산하였다. 그림 12는 물체의 최초의 위치로부터 다른 위치로 이동이 이루어 질 경우 물체의 이동에 대한 속도 및 각도를 구하는 그림이다.

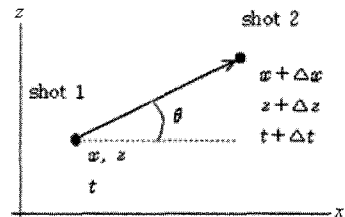


그림 12. 이동 물체의 속도와 각도.  
Fig. 12. Velocity and orientation of moving target.

$x$  방향과  $z$  방향 (목표물과 로봇간의 거리)으로의 속도  $v_x, v_z$ 는 다음과 같이 계산된다.

$$\begin{aligned} v_x &= \frac{\Delta x}{\Delta t} \\ v_z &= \frac{\Delta z}{\Delta t} \end{aligned} \quad (17)$$

로봇이 이동해야하는 속도  $v$ 는  $x$ 축과  $z$ 축의 합성 속도 값이며 이 값이 전 방향 로봇의 이동 속도가 된다. 회전각도  $\theta$ 는 두 속도의 기울기 값으로 다음과 같이 정해진다.

$$\theta = \tan^{-1} \frac{v_z}{v_x} \quad (18)$$

따라서 물체의 이동속도와 방향이 구해지면 이동하고자 하는 속도  $v$ 와 이동방향  $\theta$ 를 (4)에 대입하여 각 바퀴별 필요한 각속도를 구하게 된다.

본 논문에서 제안, 구현한 로봇의 시스템 구성은 크게 쌍안 카메라 시스템을 구현하기 위한 두 대의 CCD 카메라와 120° 간격으로 배치된 전 방향 바퀴와 이를 제어하는 모션제어기로 나눌 수 있다. 그림 13은 제작된 전 방향 로봇과 로봇 몸체에 설치된 쌍안카메라의 모습이다. 그림 14는 이동 물체의 실제 그림이다. 이 물체는 외부 명령에 의해 임의의 방향으로 이동을 하게 된다.

이동 물체는 두 개의 DC 모터로 구동되는 120×100×130 (mm) 크기의 미니 로봇이며 최대 150mm/s의 속력을 갖는다. 전 방향 구동 로봇이 이동 물체를 추적하는 실험을 하기 위해 이동 물체의 최대 속도는 100mm/sec로 제한하였고, 전 방향 구동 로봇의 최대 속도는 200mm/sec로 제한하였다. 그리고 이동 물체와 전 방향 구동 로봇의 거리(depth)가 ‘일정 거리(300mm)’와 ‘추적 완료거리(100mm)’의 두 단계로

추적하게끔 하였다. 정지하고 있는 물체의 추적과 이동하고 있는 물체의 추적의 두 단계로 실험을 하였으며, 초기의 대상 물체와 전 방향 구동 로봇의 거리는 2500mm로 하였다. 실험 결과 정지 물체의 추적은 약 12sec, 이동 물체의 경우 약 14sec가 소요되었다. 그림 15는 이동물체 추적과정을 보여준다. 그림 15는 로봇이 정지된 물체를 인식한 후 물체와 직선이 되도록 유지( $\theta=90^\circ$ 가 되도록 유지)하면서 일정 거리까지 추적한다. 일정 거리까지 추적 후, 물체가 이동하는가를 판단한 다음 물체가 이동하면 물체의 이동 방향과, 속도를 계산한다. 그런 후, 일정 거리를 유지하며 물체를 추적한다. 이동하는 물체가 정지한 후에는 정지 했는지를 다시 판단하여 정지된 물체로 인식되면 로봇의 자세를 제어하면서 추적 완료거리까지 추적한 후 추적을 마치게 된다. 여기서 ‘일정 거리’와 ‘추적 완료거리’는 프로그래밍 되어 있는 거리이다.

만약, 물체의 초기 상태가 정지되어 있지 않고 임의의 방향으로 이동하고 있을 때에는 물체의 이동방향과 속도를 계산한 후 일정 거리까지 물체와 로봇이 직선이 되도록 ( $\theta=90^\circ$ 가 되도록) 로봇의 자세를 제어하며 추적을 하게 된다.

앞서 소개된 그림 10은 본 논문에서 구현한 전 방향 구동 로봇의 스트레오비전 시스템 프로그램이다. 앞서 이야기 한 바와 같이 정합된 점은 파랑색 십(+)자선으로 나타내었고, 대상 물체의 중심점은 빨강색 십(+)자선으로 나타내도록 프로그래밍 되어있다. 중앙의 녹색 십(+)자선은 이미지의 중심을 표시하는 선이며, 빨강색으로 나타난 직선은 왼쪽과 오른쪽의 카메라 높이가 일치하는지 알아보기 위해서 표현된 대상 물체의 최 하단을 표시하는 직선이다. 실험 결과 주변의 밝기에 따라 임계값을 적절히 조정하게 함으로써 대상 물체를 찾는데 강인한 특징을 나타냈다. 이 프로그램에서는 TMS320LF2407A 마스터보드와 RS232C를 이용하

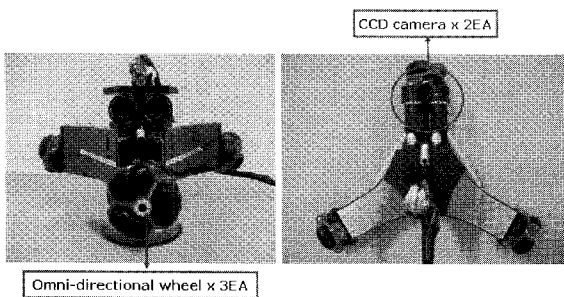


그림 13. 개발된 전방향 구동 로봇과 쌍안 카메라.  
Fig. 13. Developed omni-directional robot and binocular camera.

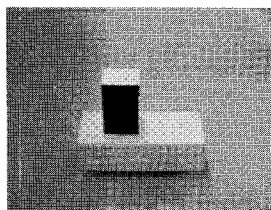


그림 14. 이동 물체.  
Fig. 14. A moving object.

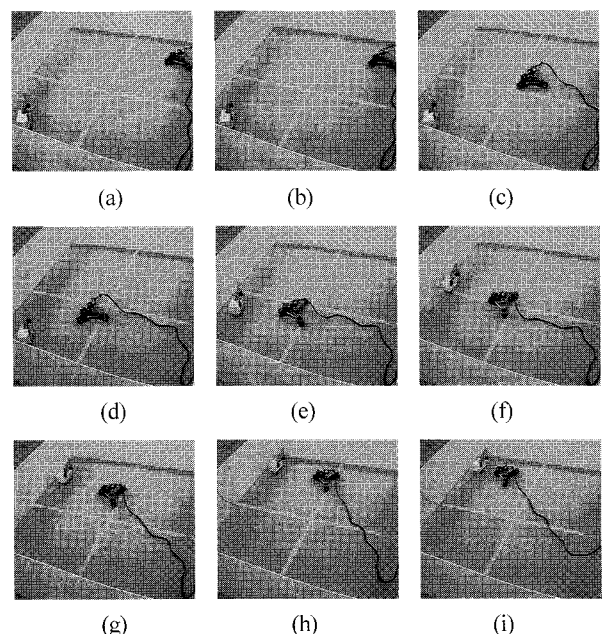


그림 15. 전방향 구동 로봇의 이동 물체 추적과정.  
Fig. 15. Procedures of tracking a moving target for omni wheel robot.

여 전 방향 구동로봇을 구동할 수 있는 프로그램을 나타내며, 매뉴얼 구동이 가능하도록 프로그래밍 되어있다.

앞서 소개된 그림 10은 본 논문에서 구현한 전 방향 구동 로봇의 스트레오비전 시스템 프로그램이다. 앞서 이야기한 바와 같이 정합된 점은 파랑색 십(+)자선으로 나타내었고, 대상 물체의 중심점은 빨강색 십(+)자선으로 나타내도록 프로그래밍 되어있다. 중앙의 녹색 십(+)자선은 이미지의 중심을 표시하는 선이며, 빨강색으로 나타난 직선은 왼쪽과 오른쪽의 카메라 높이가 일치하는지 알아보기 위해서 표현된 대상 물체의 최 하단을 표시하는 직선이다. 실험 결과 주변의 밝기에 따라 임계값을 적절히 조정하게 함으로써 대상 물체를 찾는데 강한 특징을 나타냈다.

## VI. 결론

본 논문에서는 전 방향 구동 로봇에 쌍안 카메라 시스템을 장착하여 이동 물체를 효과적으로 추적하는 알고리즘을 제시하였으며, 전 방향 구동 로봇의 설계 방법과 제어 방법을 제시하였다. 또한 쌍안 카메라 시스템에서 카메라의 보정 방법, 스테레오 이미지의 정합 방법, 3차원 좌표 복원 방법을 제시하였다. 이미지의 정합 과정에서 영상의 임계값을 이용한 이진화 기법과 윤곽선 검출을 이용하여 대상 물체의 경계선 추적을 추적하고 대상 물체를 인식하게 하였다. 3차원 복원 데이터의 오차를 측정할 결과  $x$ 좌표의 측정 오차는 -1mm~22mm로 나타났으며  $y$ 좌표의 측정 오차는 4mm~18mm로 나타났다. 대상 물체의 깊이 정보인  $z$ 좌표의 측정 오차는 5mm~120mm로 나타났으며,  $z$ 좌표의 경우 0~800mm일 때, 측정 오차가  $\pm 50$ mm 이내로 측정되어 이 구간을 물체를 추적하기 위한 신뢰성 구간으로 설정하여 사용하였다. 카메라와 대상 물체 사이의 방향 정보를 측정하기 위한  $\theta$  측정 오차는  $0.08^{\circ}$ ~ $3.0^{\circ}$ 로 나타났다.

실험 결과 측정 오차의 발생 원인은 측정시의 발생하는 오차와 렌즈의 왜곡에 의한 오차, 변위의 오차 때문인 것으로 사료된다.

기존의 연구들은 3차원 좌표복원에 있어 왼쪽과 오른쪽 카메라를 평행하도록 구축한 후 좌표를 계산하는 삼각측정 기법(simple stereo system)을 적용하였다[12,13]. 이러한 기법은 연산에 있어서 계산량을 현저히 줄이는 장점을 지닌 반면 양쪽의 카메라가 평행하지 않을 시에는 적용할 수 없다는 한계를 지니고 있다.

본 논문에서 제시한 3차원 좌표복원에 관한 연구는 왼쪽 카메라와 오른쪽 카메라의 공통 FOV(Field of View)가 존재할 때(스테레오 영역이 존재할 때), 기존의 연구와 차별되도록 왼쪽과 오른쪽 카메라의 위치나 자세를 구속하지 않으면서 한 점에 대하여 3차원 좌표를 복원하는 방법을 제시했음에 큰 의의를 들 수 있다.

전 방향 구동 방식을 채택함으로써 자율 주행 로봇이 이동 물체를 추적 할 때, 방향 전환이 필요 없는 빠른 움직임과 이동을 가능하게 함으로써 기존의 구동 방식과 차별화된 움직임이 가능하게 하였다. 또한 일반적인 구동부 구조를 지닌 자율주행로봇에 비하여 기동성이 뛰어난을 검증하였다.

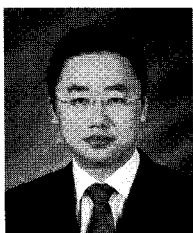
후후에는 이동 물체의 인식, 3차원 좌표 복원에 있어서 오차를 줄이기 위한 전처리 단계인 렌즈의 왜곡을 제거하는 기법, 연산 시간을 단축시키는 알고리즘의 개선과 왼쪽과 오른쪽 카메라의 모든 점들을 매칭하는 방법(스테레오 매칭)을 적용함으로써 보다 인간의 시각과 유사한 성능을 발휘하는 기법에 대한 연구가 필요할 것이다.

## 참고문헌

- [1] B. Carlisle, "An omnidirectional mobile robot," development in robotics, Kempston, pp. 79-87, 1983.
- [2] F. Pin, and S. M. Killough, "A new family of omnidirectional and holonomicwheeled platforms for mobile robot," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 6, pp. 978-989, 1999.
- [3] M. West, and H. Asada, "Design of a holonomic omnidirectional vehicle," *IEEE International Conference on Robotics and Automation*. pp. 97-103, 1992.
- [4] 변경석, 송재복 "가변 바퀴 배치 기구부를 갖는 4륜 전방향 이동로봇의 무단변속기 제어," 대한기계학회, 대한기계학회 춘추학술대회 대한기계학회 2002년 춘계학술대회논문집, pp. 871-876, 2002. 5.
- [5] U. R. Dhond and J. K. Aggarwal, "Structure from stereo-a review," *IEEE Trans.Syst., Man, Cybern.*, vol. SMC-19. no. 6, pp. 1489-150, 1989.
- [6] 최우영, 박래홍, "Stereo Vision 및 응용," 대한전자공학회, 전자공학회지 제 21 권 제 10 호, pp. 1010-1020, 1994. 10.
- [7] 김한성, 손광훈, "다중 스테레오 카메라를 이용한 3차원 모델링 시스템," 대한전자공학회, 전자공학회논문지, 제 44 권 제 1 호, pp. 1-9, 2007. 1.
- [8] 김수인, 남궁재찬, "스테레오비전 시스템에서의 이동 객체 추출 및 거리 측정," 멀티미디어학회, 멀티미디어학회논문지 제 5 권 제 3 호, 2002. 6.
- [9] 김도윤, "예제로 배우는 제어용 DSP" 성안당, 2003.
- [10] M. Xie, *Fundamentals of robotics: linking perception to action*, World Scientific, 2003.
- [11] E. Trucco, and A. Verri "Introductory techniques for 3-D Computer vision," 1998.
- [12] 조현, 진광식, 윤태성, 이용중, "스테레오 비전과 레이저포인터를 이용한 거리 측정 시스템" 산기연논문집, vol. 13, 1999.
- [13] 이주신, "단일 이동물체 추적을 위한 능동적인 스테레오 비전 시스템 구현" 산업과학연구, vol. 14, 1996.
- [14] H. P. Moravec, "Towards automatic visual obstacle avoidance," *Proc. Int. Joint Conf. Artif. Intell.*, Cambridge, MA, p. 584, Aug. 1977.
- [15] K. Mori, M. Kidode, and H. Asada, "An iterative prediction and correction method for automatic stereo comparison," *Computer Graphics Image Process*, vol. 2. no. 3/4, pp. 393-401, Dec. 1973.

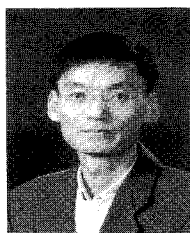


- [16] L. H. Quam, "Hierarchical warp stereo," *Proc. DARPA Image Understanding Workshop*, New Orleans, LA, pp. 149-155, 1984.
- [17] K. Ikeuchi, "Region-based stereo on needle maps," *Proc. ICAR*, pp. 207-214, 1985.
- [18] D. Marr and T. Poggio, "A computational theory of human stereo vision," *Proc. Royal Soc. London*, vol. B204, pp. 301-328, 1979.
- [19] W.E.L. Grimson, "From images to surfaces." MIT Press, Cambridge, MA, 1981.
- [20] H. H. Baker and T. O. Binford, "Depth from edge and intensity based stereo," *Proc. Int. Joint Conf. Artif. Intell.*, Vancouver, B.C., Canada, pp. 631-636, Aug. 1981.
- [21] Y. Ohta and T. Kanade, "Stereo by intra and inter scanline search using dynamic programming," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-7, no. 2, pp. 139-154, Mar. 1985.
- [22] S. A. Lloyd, E. R. Haddow, and J. F. Boyce, "A parallel binocular stereo algorithm utilizing dynamic programming and relaxation labelling," *Computer. Vision. Graphics. Image Process*, vol. 39, no. 2, pp. 202-225, Aug. 1987.



김 산

2005년 8월 서울산업대학교 기계설계 자동화공학부 졸업. 2008년 2월 서울산업대학원 메카트로닉스공학과 석사 졸업. 관심분야는 스테레오비전, 로봇틱스, 자동 제어 및 최적 제어.



김 동 환

1986년 서울대 기계설계학과 졸업. 1988년 동 대학원 석사졸업. 1995년 Georgia Institute of Technology 박사. 현재 서울산업대학교 기계설계자동화공학부 부교수. 관심분야는 메카트로닉스, 로봇틱스.