

# 스테레오 비전을 이용한 6 족 로봇의 장애물 회피를 위한 국소맵 빌딩 및 경로생성에 관한 연구

## A Study on Stereo Vision-based Local Map Building and Path Generation for Obstacle Avoidance of the Hexapod Robot

노경곤<sup>1,✉</sup>, 김진걸<sup>1</sup>  
Gyung-Gon Noh<sup>1,✉</sup> and Jin-Geol Kim<sup>1</sup>

<sup>1</sup> 인하대학교 IT 공과대학 전기공학부 (School of Electrical Engineering, Inha Univ.)  
✉ Corresponding author: as900515@hotmail.com, Tel: 032-860-8923

Manuscript received: 2009.7.29 / Revised: 2009.12.2 / Accepted: 2010.1.29

*This paper is concerned with stereo vision-based approach to detect obstacles and to generate the path of destination from the start. The hexapod robot in the experiment is cable of walking by legs and driving by wheels simultaneously. The hexapod robot operates under the driving mode normally, and it changes driving mode to walking mode to overcome obstacles using its legs. Disparity map, which is the correlation between two images taken by stereo camera, is employed for calculation of the distance between the robot and obstacles. When the obstacles information is extracted from the disparity map, the potential field algorithm is applied to create the obstacle-avoidance path. Simulator, based on OpenGL, is developed to generate the graphical path, and the experimental results are shown for the verification of the proposed algorithm.*

Key Words: Stereo Vision (스테레오 비전), Hexapod Robot (6 족 로봇), Local Map Building (국소맵 빌딩)

### 1. 서론

현재 이동로봇은 다리를 가진 보행 로봇과 바퀴가 달린 모바일 로봇으로 크게 나눌 수 있다. 국내에서 연구중인 4 족 로봇으로는 FRAMIX<sup>1</sup> 가 있고 일본에서 연구중인 4 족 로봇은 AIBO,<sup>2</sup> Tekken series<sup>3</sup> 그리고 TITAN series<sup>4</sup> 등이 있다. 위의 로봇들은 주행이 불가능한 설계로 인하여 로봇 구동에 필요한 전력수급이 원활하지 못한 것이 문제점으로 지적되고 있다. 바퀴 달린 로봇들은 장애물을 회피하는 등의 원활한 이동성을 기초로 개발되어 있다. 하지만 주행타입의 로봇들은 이동하는 경로에 지면의 높이가 불연속적인 턱과 같은 장애물을 만나게 되면 턱을 회피하거나 극복하기

가 어렵게 되므로 로봇의 이동에 제약을 받게 된다.

따라서 본 논문에서 제안하는 6 족 로봇은 보행과 주행이 동시에 가능한 형태로 설계하여 로봇이 진행하기 힘든 지형을 극복할 수 있고 장애물 또한 넘을 수 있는 구조로 되어있다. 본 로봇은 기본적으로 6 족의 다리를 가지고 있으며, 외곽의 4 개의 다리의 끝에 바퀴 모듈이 장착되어 있다. 또한 조향이 가능하도록 바퀴 모듈 위에 모터가 장착되어 있다.

지형지물을 판단하기 위한 방법으로는 초음파 센서를 이용한 거리측정,<sup>5</sup> 레이저 스캐너를 이용한 거리측정,<sup>6</sup> 영상처리를 이용하는 방법<sup>7</sup> 등이 있는데 본 논문에서는 부가적인 센서의 사용을 최소화하고 장애물의 다양한 정보를 효율적인 획득을 위

하여 스테레오 카메라를 이용하였다. 거리는 한 점에서 다른 점까지의 길이를 나타내는 것으로 인간의 양안에 의한 거리 감각과 같이 카메라를 수평으로 일정 거리의 차이를 두어 시차가 생기게 하여 장애물과의 거리를 측정할 수 있게 된다. 시차 정보를 추출하기 위하여 카메라 보정을 하여 카메라의 특성 왜곡을 보정하고 두 이미지 사이의 픽셀이 동일한 축에 있어야 하므로 영상의 평행화 과정<sup>8</sup>을 수행하였다. 이렇게 수행된 이미지 사이의 시차를 명암도 영상으로 나타낸 시차지도<sup>9</sup>를 이용하여 장애물의 위치 및 크기를 검출하여 경로 계획에 사용하였다. 본 논문에서는 스테레오 카메라를 사용하여 출발점과 목표점에 위치한 장애물의 위치와 크기를 검출하고 이를 회피 또는 극복하기 위한 국소맵 상의 궤적을 생성하며 실험을 통하여 이를 검증한다.

2. 스테레오 시각시스템

2.1 스테레오 영상 시스템 기구

스테레오 비전을 사용하여 수평적으로 위치한 서로 다른 2 대의 카메라로부터 같은 물체의 이미지를 얻어내어 3 차원 영상이 2 차원 스크린으로 투영될 때 소멸되는 물체의 거리정보를 추출하였다. 그러므로 영상 시스템은 인간의 시각 구조와 같은 수평시차 분만 가지도록 설계되었으며 카메라 사이의 간격을 조절할 수 있도록 제작하였다. 카메라는 IEEE-1394 방식의 디지털 카메라로 추가적인 Grabber 보드가 필요없어 기존의 아날로그 카메라보다 영상 시스템 구축이 간편해졌다. 그림 1은 스테레오 영상 시스템의 모델링과 실제 모습을 보여준다.

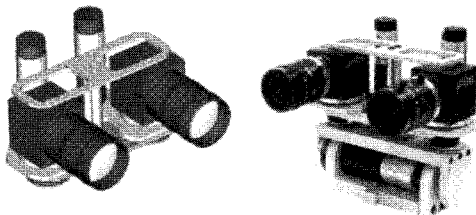


Fig. 1 Stereo vision system

2.2 카메라 변수

카메라 보정(camera calibration)은 회전요소, 이동요소, 크기요소 값을 이용하여 한 영상을 변형 시킴으로써 두 영상간의 부분적으로 동일한 영역

이 오버랩 되도록 하는 작업을 말하며 영상처리의 전처리 과정 중 하나이다. 카메라 행렬은 내부 행렬과 외부 행렬로 나타낼 수 있는데 내부 행렬은 초점거리(focal length), 주점(principal point), 렌즈 왜곡(lens distortion)등의 정보를 담고 있으며 외부 행렬은 회전, 이동, 투영의 정보를 담고 있다.

카메라 변수는 불완전한 카메라 렌즈에 의해 디지털 이미지 좌표계에 투영되는 영상과 전역 좌표계간의 관계를 정의하는 변수이다. 그림 2는 3 차원의 데이터가 카메라 변수 행렬에 의해 2 차원의 이미지로 변환되는 관계를 행렬로 나타내며 X, Y, Z는 3 차원 좌표이며, u, v는 카메라에 의해 획득된 2 차원 좌표이다.

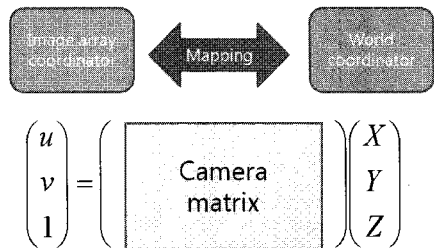


Fig. 2 Relation between image coordinate and world coordinate

2.3 영상의 평행화

스테레오 비전의 구조는 이미 잘 알려져 있는 에피폴라(Epipolar) 기하학으로 간단히 표현할 수 있다. 그림 3은 두 카메라의 투영중점  $O_r, O_l$ 과 이미지 평면  $\pi_r, \pi_l$ 을 보여주고 있다.  $P_r = [x_r, y_r, z_r]$ ,  $P_l = [x_l, y_l, z_l]$ 은 카메라 좌표계에서 3 차원상에 한 점  $p$ 를 가리키는 벡터이고  $\bar{P}_r = [\bar{x}_r, \bar{y}_r, \bar{z}_r]$ ,  $\bar{P}_l = [\bar{x}_l, \bar{y}_l, \bar{z}_l]$ 는 점  $p$ 를 가리키는 좌우 이미지 안의 좌표를 나타낸다.

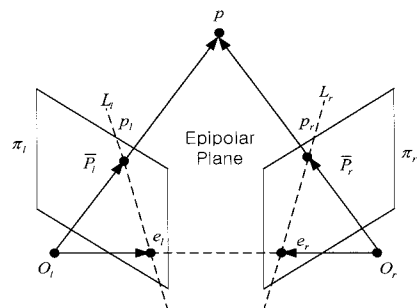


Fig. 3 Epipolar geometry

좌측 이미지 평면  $\pi_l$ 의 한 점  $\bar{p}_l$ 에 대해서 우측 이미지에 대응하는 점  $\bar{p}_r$ 과  $e_l$ 를 연결하는 선 위에 존재한다. 이 선을 에피폴라 선이라고 하고  $L_r, L_l$ 로 정의한다. 에피폴라 선은 두 카메라의 투영중점  $O_r, O_l$ 과  $p$ 로 정의되는 에피폴라 평면과 이미지 평면  $\pi_r, \pi_l$ 의 교선이고, 한 이미지의 모든 점에 에피폴라 라인은 공통적으로  $e_l, e_r$ 을 지난다. 이를 에피폴라 이라고 하고,  $O_r, O_l$ 과 이미지 평면  $\pi_r, \pi_l$ 과 교점을 이룬다. 만약  $\bar{p}_l, \bar{p}_r$ 이 3 차원 공간상에서 한 점  $p$ 와 각각 대응되는 점이라면 한 이미지평면의 점은 다른 이미지 평면상의 에피폴라 라인 위에 존재함으로 대응하는 점을 찾기 위한 영역이 2 차원에서 1 차원으로 줄일 수 있다. 이를 에피폴라 제약조건이라고 한다.

카메라 좌표계 안에서 두 카메라의 관계는 외부 매개변수인 이동벡터 ( $T=O_r - O_l$ )와 회전벡터  $R$ 로 표현될 수 있다. 공간상에 주어진  $p$ 에 대하여  $p_l$ 와  $p_r$ 의 관계는 다음의 식(1)과 같다.

$$P_r = R(P_l - T) \tag{1}$$

또한 픽셀 좌표계에서의  $\bar{p}_l, \bar{p}_r$ 과 카메라 좌표계의  $p_l, p_r$ 의 관계는 다음의 식(2)와 식(3)으로 표현 할 수 있고  $f_l, f_r$ 은 각각 카메라의 초점거리를 나타낸다.

$$\bar{P}_l = \frac{f_l}{z_l} p_l \tag{2}$$

$$\bar{P}_r = \frac{f_r}{z_r} p_r \tag{3}$$

### 2.4 카메라 평행화 결과

아래의 그림 4는 체스보드의 edge를 구하여 실 세계 좌표를 구하여 보정과 평행화한 결과이다. 2대의 좌우 스테레오 카메라의 동기화 보정을 하기 위하여 동일 목표물의 10 가지 경우(정상 상태, 단 방향 회전 4 경우, 복합회전 5 경우)를 적용하여 대응점을 구하였다.

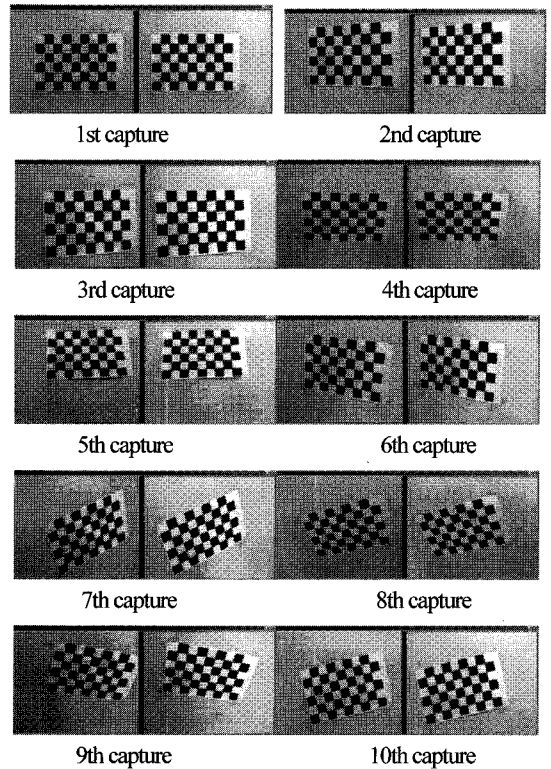


Fig. 4 Extract corresponding points and rectification

입력 목표물의 크기는 1000 x 1000 mm 이며, 획득 이미지의 크기는 640 x 480 화소이다. 표 1은 10 가지의 경우에 대하여 보정된 대응점을 추출하여 구해진 각 카메라의 내부 변수 행렬과 기초 행렬이다. 내부 변수 행렬의 대각값은 카메라의 수평과 수직 초점거리를, 3 열은 중심 위치를 나타내며, 기초행렬은 두 행렬의 보정 기본 값을 나타낸다.

Table 1 Calculated camera parameters

Camera_1	1072.51	0.00	326.30
intrinsic	0.00	1072.37	227.81
parameter	0.00	0.00	1.00
Camera_2	1066.11	0.00	321.30
intrinsic	0.00	1067.29	211.95
parameter	0.00	0.00	1.00
Fundamental	0.00	0.00	0.00009
matrix	0.00	0.00	-0.08590
	-0.00022	0.08632	1.00

### 3. 로봇구조

#### 3.1 기구적 구조

다양한 환경내에서 여러 장애물을 극복 내지 회피하기 위하여 다리 구조와 바퀴 구조가 융합된 6 족 로봇을 자체 개발하였다. 로봇의 이동성을 극대화 하기 위하여 본체는 알루미늄 재질을 사용하였고, 다양한 환경에 순응하기 위하여 각 다리는 5 자유도를 부가하였다. 그림 5 는 변형 가능한 6 족 로봇의 실제 모습과 각 관절의 회전 축을 나타낸다. 6 개의 각 다리는 동일한 구조로 이루어져 있으며, 각 다리는 3 차원 위치 추종을 위한 3 자유도와 바퀴 구동시 방향 전환을 위한 1 자유도 그리고 바퀴의 구동을 위한 1 자유도로 구성된다.

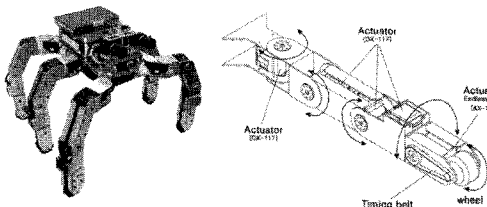


Fig. 5 Transformable hexapod robot

#### 3.2 로봇의 제어기 구성

제어시스템은 크게 세 부분으로 나눌 수 있다. 영상 처리 로봇의 보행계획 생성을 담당하는 주 제어기와 30 개의 모터 제어기에 보행 계획 데이터를 전송하는 메시지 제어기로 나뉜다. 각각의 서보모터는 자체의 서보 제어기능을 내장한 소형 및 모듈화된 RC 서보모터를 사용하였다. 영상 카메라는 두 개의 소형 IEEE-1394 디지털 카메라를 사용하였으며, 중앙 처리기로는 소형의 Micro-ITX 메인보드와 펜티엄 4 CPU 그리고 Windows-XP 운영체제로 영상처리 성능과 프로그램의 이식성을 높였다. 또한 사용자와 로봇과의 인터페이스는 Blue-Tooth 무선통신<sup>13</sup>을 사용하였다. 메시지 제어기는 다양한 인터페이스 환경을 구축할 수 있는 AT-mega128 CPU<sup>14</sup>를 사용하였다. 그림 6 은 6 족 로봇의 시스템을 나타낸 것이다.

### 4. 변위지도 생성과 장애물 검출

장애물의 위치와 거리를 판별하고 변위지도를 생성하기 위해 이미지 데이터를 가공하는 단계가 요구된다.

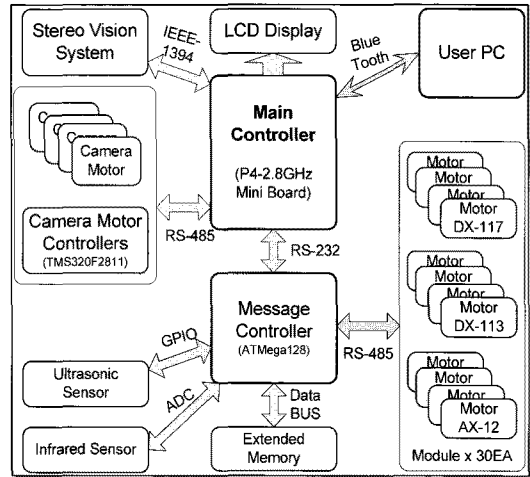


Fig. 6 Robot system configuration

따라서 초기 입력된 이미지를 처리하여 기본적인 거리 데이터를 획득하는 전처리 과정과, 이에 배경과 장애물을 구별하여 이미지상의 불필요한 노이즈를 제거하는 후처리 과정이 필요하다. 또한 이렇게 얻어진 좌우 한 쌍의 이미지로부터 장애물의 거리와 크기 정보를 추출하고 마지막으로 장애물을 배경으로부터 완전히 분리하는 과정을 거치게 된다. 다음은 이러한 과정을 자세히 기술한다.

#### 4.1 전처리 과정

전처리 과정은 크게 3 부분으로 나뉜다. 첫 번째는 과정은 이미지의 크기를 줄이는 과정이다. 이미지 처리의 연산시간을 줄이기 위해 Inter-cubic 방식을 사용하여 320 x 240 크기로 변환하였다. 두 번째 과정은 영상의 밝기 변화나 광원의 영향을 최소화하고, 빠른 연산을 하기 위해 LoG (Laplacian of Gaussian filter)<sup>17</sup>를 이용한다. 그러나 특징이 영상전체에 고루 분포되어 있지 않을 경우에는 추출할 수 있는 특징정보가 부족하기 때문에 정합의 정확도가 저하된다. 이를 극복하기 위하여 마지막으로 미디언 필터 (Median filter)<sup>17</sup>를 사용하였다. 미디언 필터는 입력영상의 주변 픽셀 값들을 오름차순 또는 내림차순으로 정렬하여 그 중앙에 있는 픽셀 값을 사용하는 필터로써 노이즈 충격<sup>17</sup>을 제거한다. 아래의 그림 7 은 전처리 과정을 거치고 SSD(Sum of Squared Differences)로 정합된 변위지도를 나타낸다. 흰색에 가까울수록 카메라와 가깝고 검은색에 가까울수록 카메라와의 거리가 먼 것이다.

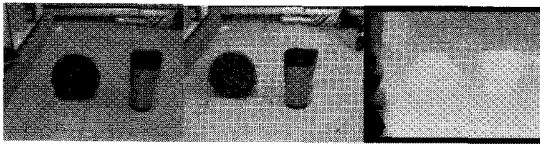


Fig. 7 Rectified images and disparity map

**4.2 후처리 과정**

후처리 과정은 크게 2 가지로 나뉜다. 하나는 모폴로지컬 필터 (Morphological filter)<sup>17</sup> 이고, 다른 하나는 스파이크 제거이다. 모폴로지컬 필터 연산은 침식(erosion) 연산 후 팽창(dilation) 연산을 하게 되는데 이를 닫힘(closing) 연산이라 한다. 닫힘 연산을 하게 되면 원래 영상의 형태는 유지하게 되고, 작은 노이즈는 사라져 버리기 때문에 노이즈를 제거하는 효과를 볼 수 있다. 마지막으로 스파이크 제거 알고리즘<sup>18</sup> 은 영역 기반 정합 방식에서 깊이가 불연속적으로 변하거나 경사가 급격하게 변하는 경우에 발생하며, 국소적으로 나타나는 특징이 있다. 따라서 국소적으로 나타난다는 특징을 이용하여 이를 제거할 수 있다. 변위지도에서 스파이크가 존재할 경우, 스파이크가 존재하는 열에서 스파이크 값은 상대적으로 큰 값을 갖는다. 따라서 스파이크 값과 진짜 값은 일정한 차이가 발생한다. 실험적으로 명도차이가 10~12 라는 것을 확인 하였으며 이 값이 발생하였을 경우 이를 스파이크로 간주하고 스파이크 전의 값을 이용하여 보정을 한다.

**4.3 이미지 상관관계**

변위는 오래 전부터 많은 사람들에 의해 연구되어 온 분야이다. 변위 알고리즘 중 가장 간단한 방법이 바로 SSD 를 이용한 방법이다. 이 알고리즘은 두 이미지 사이의 명도차이를 이용한 방법이다. 본 논문에서는 한 이미지를 기준으로 다른 이미지의 명도 값의 차이를 특정영역에서 계산하고 이 차이 값이 최소가 되는 부분이 해당 픽셀의 대응점이라고 생각하고 그때의 인덱스의 위치가 변위 값이 된다. 이와 같은 작업을 각 이미지들을 기준으로 왼쪽과 오른쪽에 해당하는 변위지도를 각각 계산해 준다. 하지만 SSD 만을 이용하여 정확한 변위 값을 구하기 힘들다. 그러므로 전처리 과정과 후처리 과정을 통하여 이미지의 품질을 향상시켰다. 아래의 그림 8 은 대응구간을 이용한 상관관계에 대한 그림이다.

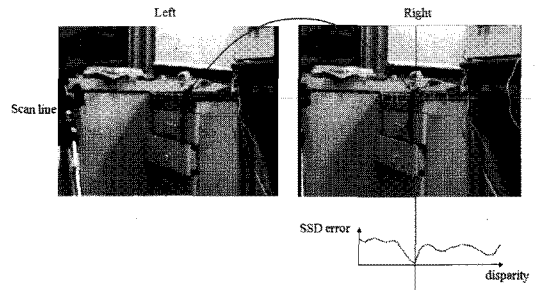


Fig. 8 Corresponding points correlation

변위 값을 구하기 위한 SSD 함수는 식(4)와 같고 여기서  $I_L, I_R$  은 왼쪽 이미지와 오른쪽 이미지의 명암 도를 나타내며  $u, v$  는 식(5)에 의하여 정의된 윈도우의 크기이다.

$$C_r(x, y, d) = \sum_{(u,v) \in W_m(x,y)} [I_L(u, v) - I_R(u - d, v)]^2 \quad (4)$$

$$W_m(x, y) = \left\{ u, v \left| \begin{array}{l} x - \frac{m}{2} \leq u \leq x + \frac{m}{2} \\ y - \frac{m}{2} \leq v \leq y + \frac{m}{2} \end{array} \right. \right\} \quad (5)$$

**4.4 장애물 검출**

계산된 변위지도에서 장애물은 바닥과 다르게 흰색을 띄게 된다. 그림 9 는 계산된 변위지도의 기울기 차이를 보여준다.

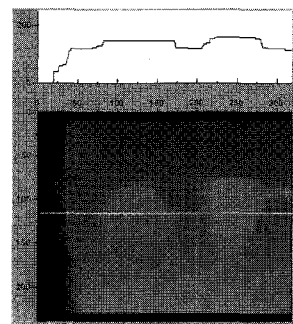


Fig. 9 Difference of gradient in disparity map

이런 특징을 살려 수평라인을 스캔하여 장애물의 테두리를 검출하게 된다. 또한 자연히 배경과 장애물은 분리가 된다. 아래의 표 2 는 장애물을 분리하는 과정을 나타낸다.

Table 2 A process of obstacle separation

```

Algorithm extract object (X,Y,I)
  For all Y do
    Sum =  $\sum_{n=1}^X I$ 
    Average =  $\frac{SUM}{X}$ 
    If Average < I, then
      I = 0
  End For
  Return new_value
    
```

X, Y 는 이미지의 수평, 수직크기이고 I 는 이미지의 명도를 나타낸다. 수평축 라인의 평균을 구한 후 평균보다 작은 값들은 명도를 0 으로 하여 배경을 분리하였다. 그림 10 은 제안된 알고리즘으로 추출한 장애물이다.



Fig. 10 Calculated obstacles by proposed algorithm

장애물들은 배경과 분리되어 하나의 덩어리 (Blob)으로 변환하였다. 이러한 덩어리들을 각각의 Blob labeling<sup>19</sup> 하기 위하여 OpenCV<sup>20</sup> 에서 제공하는 Blob labeling 함수를 사용하였다. Blob coloring 라이브러리는 표 3 과 같이 동작한다. Blob labeling 알고리즘은 각 화소의 상단과 왼쪽 라벨을 검색하여 현재 화소의 라벨을 결정한다. 이 두 이웃 화소들은 라벨링 과정에서 이미 주사된 화소이다. 결정 과정에서는 세 가지의 서로 다른 경우가 발생할 수 있는데 만약 두 이웃 화소들이 모두 전경 화소가 아닌 경우에는 현재 화소에 새로운 라벨값을 할당하고 두 화소 중 하나의 화소만 전경 화소인 경우에는 그 화소의 라벨값을 현재 화소에 할당한다. 그리고 두 화소가 모두 전경 화소이면서 같은 라벨값을 갖는 경우에는 동일한 라벨값을 현재 화소에 할당한다. 그러나, 두 화소가 전경 화소지만 서로 다른 라벨값을 갖는 경우에는 이 두 영

역은 현재 화소에 의해 서로 연결되는 영역이므로 동일한 라벨값으로 병합되어야 한다.

Table 3 Blob coloring algorithm

```

Algorithm blob labeling
  Let the initial Color[k]=0
  Step1.
  Scan the original image from left to right and top to bottom using L-shape template for k=imageSize
  if image[C]=255 then
    if image[U]=255 and image[L]=0 then
      Color[C]=Color[U]
    if image[U]=0 and image[L]=255 then
      Color[C]=Color[L]
    if image[U]=0 and image[L]=0 then
      Color[C]=K++, new color
  repeat
    
```

따라서 더 두 화소의 라벨값 중 더 작은 값을 현재 화소의 라벨값으로 할당하고 각 라벨은 동치 라벨로 등록된다. 그림 11 은 blob labeling 으로 구해진 장애물의 중심위치와 변위 값들이다.

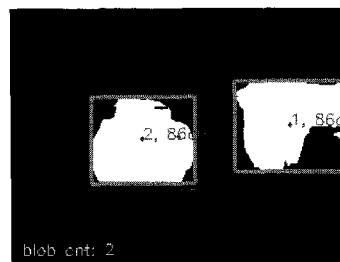


Fig. 11 Labeled obstacles

이로써 라벨링 된 장애물들은 각각의 좌표와 크기 및 거리를 알 수 있다.

### 5. 로봇의 경로생성

그림 12 는 경로생성의 방법을 선택하는 순서를 나타낸다.

로봇이 장애물을 판단하여 경로를 생성하기 위하여 첫 번째 단계로는 포텐셜 필드(Potential Field) 방법<sup>21</sup> 을 이용하여 장애물을 회피하는 경로를 생성하고 두 번째는 시작점과 도착점 사이에 최단 거리인 가상의 직선을 그렸을 때 장애물이 검출되면 로봇의 역기구학을 이용하여 로봇이 넘어가는 단계이며, 마지막으로 앞선 두 개의 방법을 혼합

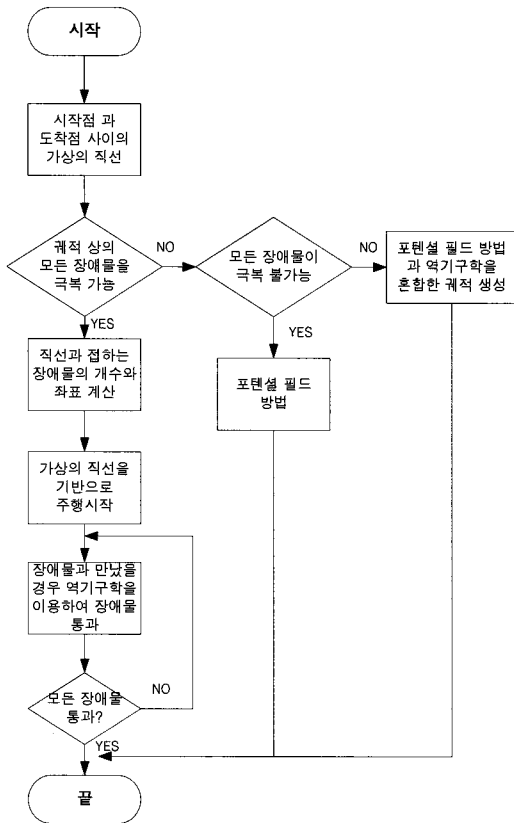


Fig. 12 Selection of path plan method

하여 사용하는 방법으로 경로를 생성할 수 있다.

먼저 포텐셜 필드를 이용한 방법은 장애물의 높이가 로봇이 통과하지 못하는 경우 적용된다. 로봇은 최대 150mm 까지 다리를 올릴 수 있으나 로봇의 동작이 불안정해지는 이유로 50mm 로 제한을 하였다. 즉 장애물의 높이가 50mm 이상일 경우 로봇은 회피동작을 하게 된다. 포텐셜 필드 방법을 사용하였을 경우 커다란 오목(concave) 형상의 장애물에 대하여 포텐셜 필드의 접근 방법이 제한된 경우가 발생 가능하므로 이러한 경우를 줄이고 빠른 연산을 하기 위하여 그림 13 과 같이 장애물의 모든 모서리를 포함 할 수 있는 원으로 근사화 하였으며, 시뮬레이션과 실제 실험에 대해서는 볼록(convex) 형상과 비교적 작은 크기(10cm x 10cm)의 장애물로 제한하였다.

### 5.1 포텐셜 필드를 이용한 경로계획

포텐셜 필드 방법은 자율 이동로봇의 경로 계획이나, 작업공간 내에서 로봇 매니퓰레이터가 충돌없이 목적지까지 이동시키는 문제에 주로 적용되었다.

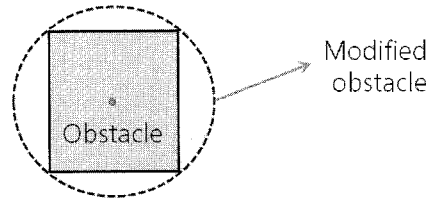


Fig. 13 Approximated obstacle as a circle

돌없이 목적지까지 이동시키는 문제에 주로 적용되었다. 포텐셜 필드 방법은 로봇을 포텐셜 함수의 영향을 받는 물체로 취급하는 것이다. 장애물은 척력의 포텐셜 (Repulsive potential)에 의하여 모델링 되고, 목표점은 인력의 포텐셜 (Attractive potential)에 의하여 모델링 된다. 따라서 모든 작업 공간은 위의 두 전위 함수의 합에 의해 표현된다. 식 (6)은 전체 포텐셜 필드를 나타낸다. 여기서  $U_{xd}(x)$ 는 인력의 포텐셜을 나타내며 식 (7)과 같이 정의된다. 식 (7)에서  $k_p$ 는 포지션의 게인 값을 나타내며,  $x$ 는 현재 포지션 값,  $x_d$ 는 목표의 좌표를 나타낸다. 식 (8)은 척력의 포텐셜이며 다음과 같다.

$$U(x) = U_{xd}(x) + U_0(x) \quad (6)$$

$$U_{xd}(x) = \frac{1}{2} k_p (x - x_d)^2 \quad (7)$$

$$U_0(x) = \begin{cases} \frac{1}{2} (\frac{1}{\rho} - \frac{1}{\rho_0})^2 & \text{if } \rho < \rho_0 \\ 0 & \text{if } \rho \geq \rho_0 \end{cases} \quad (8)$$

인력과 척력의 전위장에 착안점을 둔 이 방법은 복잡한 형상공간의 모델링이 필요 없고 비교적 간단한 계산으로 로봇의 경로계획이 가능하다. 그림 14는 좌표의 원점 즉 (0, 0)에서 출발하여 (90, 90)으로 경로가 생성되는 것을 볼 수 있다. 본 연구에서는 장애물의 형태가 각이진 형태가 아닌 원의 모양이 국부 최소점에 강인한 면모를 보인다는 점을 이용하여 장애물의 형태를 원으로 근사화 하였다.

생성된 궤적을 기반으로 시뮬레이션 상에서 로봇의 움직임을 구현하기 위하여 경유점 간의 각도와 거리를 알아야 한다. 아래의 그림 15는 각 경유점간의 각도와 거리를 알 수 있는 방법을 제시한 것이다. 여기서  $\alpha$ 는 현재의 포인트와 다음 포

인트간의 각을 나타내며 다음 식 (9)와 같이 표현 된다.

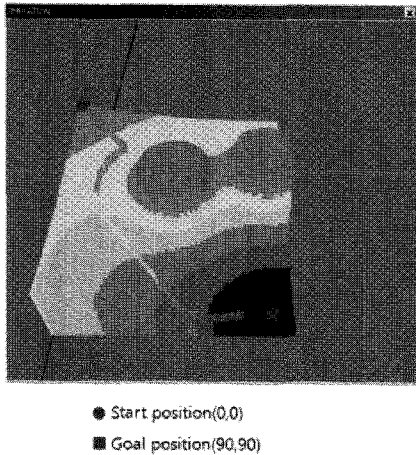


Fig. 14 Path generation and obstacle expression using a potential field

때, 포텐셜 필드를 이용하여 궤적을 생성한 시뮬레이션의 결과를 보여준다.

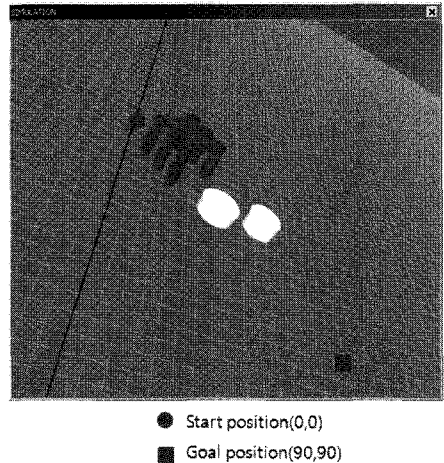


Fig. 16 Driving simulation result

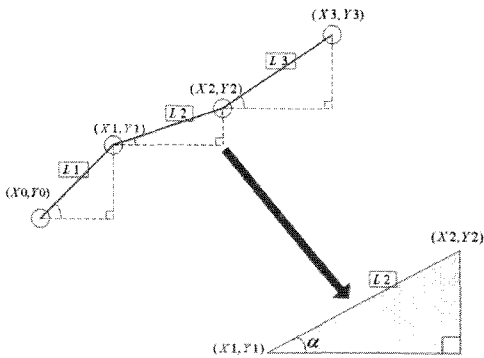


Fig. 15 Angles and distances between way points

$$\alpha = \begin{cases} \arctan\left(\frac{Y_1 - Y_0}{X_1 - X_0}\right) \times \frac{180}{\pi} & \text{if } X_0 < X_1 \\ -\arctan\left(\frac{Y_1 - Y_0}{X_1 - X_0}\right) \times \frac{180}{\pi} & \text{if } X_0 \geq X_1 \end{cases} \quad (9)$$

$\alpha$  는 두 개의 조건으로 나타낼 수 있으며 로봇의 방향을 위의 조건으로 판단할 수 있다. 또한  $L$  은 두 점 간의 거리를 나타내며 로봇이 이동하는 거리를 나타낸다. 로봇은 시작점 (0, 0) 에서부터 목표점 (90, 90)까지 106 개의 경유점을 식 (9)를 사용하여 추종하게 되며 오차 없이 목표점까지 추종하였다. 아래의 그림 16 은 장애물의 높이를 100mm 로 설정하여 로봇이 이를 극복할 수 없을

### 5.2 장애물을 넘을 수 있는 경우의 경로계획

장애물을 딛고 넘기 위해서는 장애물까지의 위치와 장애물의 크기, 높이, 형태 그리고 딛고 넘는 표면의 마찰력등과 같은 다양한 정보가 요구된다. 그러나 본 논문에서는 장애물의 형태와 크기가 일정하고, 딛고 넘는 표면의 마찰력이 로봇의 바퀴가 회전하여 이동할 수 있을 만큼 충분하다고 가정하여, 최소한의 기본정보만으로 실제 로봇의 실험을 적용할 수 있게 제한하였다. 로봇이 장애물을 통과 할 수 있을 때의 경우로 시작점과 도착점의 최단거리인 가상의 직선을 그리며, 진행 도중 장애물을 만났을 경우 장애물을 넘어가는 방법이다. 장애물은 원으로 근사화 되어있기 때문에 장애물 좌표와 반지름의 크기를 기반으로 장애물의 위치와 크기를 알 수 있다. 먼저 시작점부터 목적지까지의 경로 중에 인접하는 장애물이 있는지 확인한다. 그림 17 과 같이 직선의 방정식과 원의 방정식을 연립방정식을 풀어 교점을 판별한다.

로봇의 중심점을 기준으로 움직이기 때문에 중심으로부터 양쪽 다리를 기준으로 하여 2 개의 직선이 생기므로 이 직선의 방정식을 대입하여 풀면 어떤 다리에 장애물이 걸리는지 알 수 있다. 여기서  $d$  는 원의 중심과 직선의 거리이고  $r$  는 반지름의 길이라고 가정하면  $d < r$  의 조건이면 두 점에서 만나고  $d = r$  이면 한 점에서 만나고  $d > r$  이면 만나지 않는다. 두 점이나 한 점에서 만나는 경우에



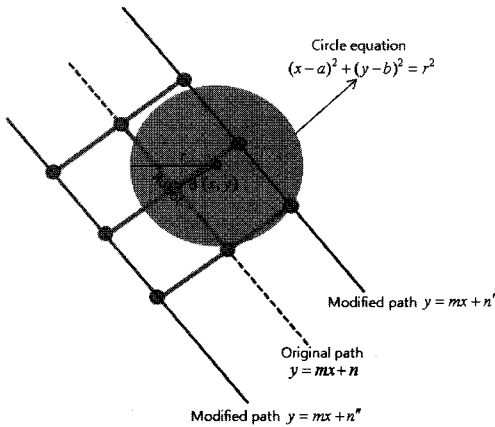


Fig. 17 Intersection between circle and line

장애물의 높이를 판별한 후 로봇이 어떤 방식으로 진행할 것인지에 대한 판단을 하게 된다. 이러한 방법으로 두 개의 직선을 대입하여 방정식을 풀어 내면 로봇의 어떤 다리에 어떤 장애물이 걸리는지를 알 수 있으며 장애물과의 교점 거리를 구하여 로봇의 궤적 상의 장애물의 길이를 알 수 있다. 여기서 장애물의 높이는 일정하다고 가정한다. 로봇의 역기구학을 풀어 로봇이 넘어 가는 동작을 하게 된다.

로봇이 장애물을 넘기 위하여 그에 상응하는 걸음새를 구현해야 한다. 역기구학을 사용하여 다리의 끝점을 입력하면 각 관절의 각도 값을 알 수 있다. 역기구학은 표 4 에 나타난 방법에 따라 각 관절 값을 구하였다.

로봇은 한 스텝을 가기 위하여 총 9 개의 세부 동작을 하게 되며 한 스텝을 진행하였을 경우 로봇은 100mm 를 이동한다. 로봇발의 끝점 좌표는 X, Y, Z 로 3 차원으로 구성이 되어있으며, 이 끝점 정보를 이용하여 3 개의 관절 값을 알 수 있다. 로봇의 다리는 6 개가 동일하기 때문에 1 개의 다리만 역기구학을 풀어 모든 다리에 적용을 하였다.

6. 실험결과

본 논문의 실험은 실내의 400cm x 400cm 의 공간에서 실험을 하였다. 실험은 로봇이 장애물을 회피할 수 있는 경우의 궤적생성과 생성된 궤적에 대한 추종에 대한 실험과, 로봇이 장애물 회피하지 않는 경우, 즉 로봇이 판단을 하여 장애물을 넘어가는 동작 그리고 마지막으로 혼합동작으로

나뉜다. 이에 각 실험은 포텐셜 필드를 이용한 장애물회피 주행실험, 로봇의 걸음새를 이용한 장애물 극복실험, 2 개를 혼합한 실험 총 3 가지 실험이며 각 실험당 2 에서 3 번씩 실시하였다.

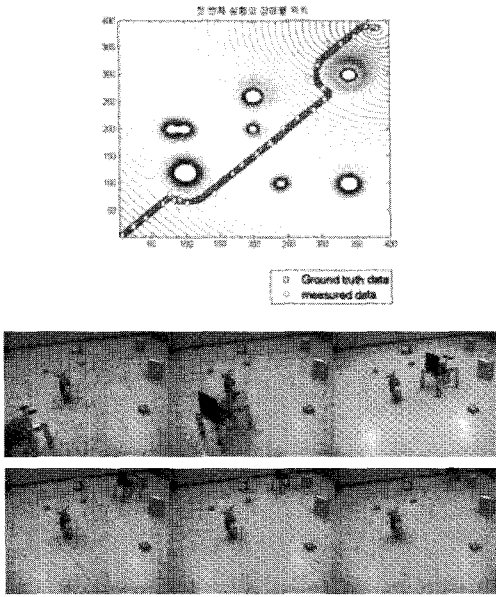
Table 4 Kinematics of leg

Forward kinematics	
$T1 = \begin{bmatrix} C1 & 0 & -S1 & L1*C1 \\ S1 & 0 & C1 & L1*S1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T2 = \begin{bmatrix} C2 & -S2 & 0 & L2*C2 \\ S2 & C2 & 0 & L2*S2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$T3 = \begin{bmatrix} C3 & -S3 & 0 & L3*C3 \\ S3 & C3 & 0 & L3*S3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$NSAP = \begin{bmatrix} Nx & Sx & Ax & Px \\ Ny & Sy & Ay & Py \\ Nz & Sz & Az & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$T123 = \begin{bmatrix} C1*C23 & -C1*S23 & -S1 & C1*(C2*L2+C23*L3)+L1*S1 \\ S1*C23 & -S1*S23 & C1 & C1*L1+S1*(C2*L2+C23*L3) \\ -S23 & -C23 & 0 & -L2*S2-L3*S23 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	
Inverse Kinematics	
$T123(1,3) = NSAR(1,3), T123(2,3) = NSAR(2,3) \quad -S1 = Ax, C1 = Ay$ $\therefore \theta1 = \arctan2(-Ax, Ay)$	
$T123(3,1) = NSAR(3,1), T123(3,2) = NSAR(3,2) \quad -S23 = Nz, -C23 = Sz$ $\therefore \theta23 = \arctan2(-Nz, -Sz)$	
$T12 = NSAP * T3^{-1}$	
$T12 = \begin{bmatrix} C1*C2 & -C1*S2 & -S1 & L1*S1+L2*C1*C2 \\ S1*C2 & -S1*S2 & C1 & L1*C1+L2*S1*C2 \\ -S2 & -C2 & 0 & -L2*S2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	
$NSAP * T3^{-1} = \begin{bmatrix} Nx*C3-Sx*S3 & Sx*C3+Nx*S3 & Ax & Px-L3*Nx \\ Ny*C3-Sy*S3 & Sy*C3+Ny*S3 & Ay & Py-L3*Ny \\ Nz*C3-Sz*S3 & Sz*C3+Nz*S3 & Az & Pz-L3*Nz \\ 0 & 0 & 0 & 1 \end{bmatrix}$	
$T12(3,4) = NSAR(3,4) \quad -L2*S2 = Px-L3*Nz$ $\therefore \theta2 = \pm \arctan2\left(\sqrt{1-\left(\frac{Px-L3*Nz}{-L3}\right)^2}, \frac{Px-L3*Nz}{-L3}\right)$	
$\therefore \theta3 = \theta23 - \theta3$	

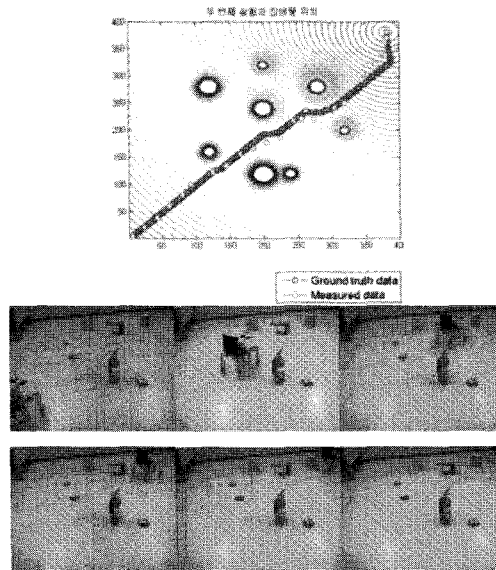
6.1 포텐셜 필드 방법을 사용한 경로추정실험

본 실험은 로봇이 장애물을 넘어가지 못하는 경우 포텐셜필드 방법에 의해 생성된 궤적을 나타낸 것이다. 실내의 400cm x 400cm 의 공간에서 각기 다른 위치에 장애물을 설치한 후 실험이 진행되었다.

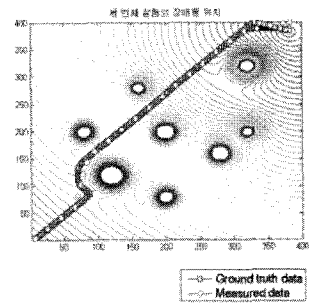
그림 18 은 포텐셜 필드를 사용한 경로추정 실험으로써 스테레오 카메라를 이용하여 획득한 이미지와 시차지도를 나타낸 것이며 총 8 개의 장애물과 시차지도를 포함하고 있다. 시차지도를 사용하여 장애물의 위치 및 크기를 추정할 수 있으며 이를 기반으로 경로를 생성하여 로봇이 진행방향을 결정할 수 있다. 이에 생성된 경로와 실제 실험



(a) 1<sup>st</sup> Experiment



(b) 2<sup>nd</sup> Experiment



(C) 3<sup>rd</sup> Experiment

Fig. 18 Experiments of obstacles avoidance

후 이동한 경로를 지면에 표시 및 비교하여 궤적 추정 오차를 비교하였다. 첫 번째 궤적 (a)의 평균 정확도는 93.49 %이며 끝점에서의 정확도는 90.98 %의 정확도를 보이고, 두 번째 궤적 (b)의 평균정확도는 93.13 %이며 끝점의 정확도는 97.23 %의 정확도를 보인다. 마지막으로 세 번째 궤적 (c)의 평균 정확도는 95.8 %이며 끝점의 정확도는 93.1 %이다.

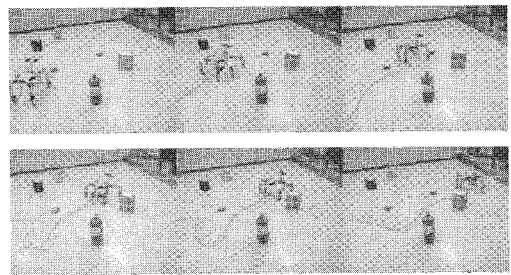
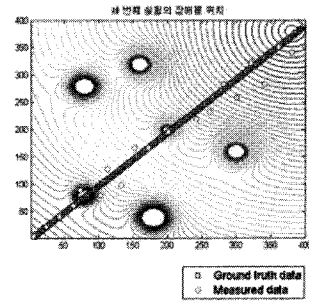
## 6.2 로봇의 역기구학을 이용한 장애물 넘기 동작실험

다음 실험은 로봇이 장애물을 넘을 수 있는 경우의 생성된 궤적을 나타낸 것이다. 첫 번째 실험과 동일한 조건인 실내의 400cm x 400cm의 공간에서 각기 다른 위치에 장애물을 설치한 후 실험을 하였다.

그림 19 는 로봇의 걸음새를 이용한 넘기 방법에 의하여 생성된 궤적을 보여준다. 각각의 실험 환경은 장애물의 경로 상 위치와 개수가 궤적 및 끝점의 추정 정확도에 주는 영향을 비교하기 위하여 (a) 경로의 목표 부근에 장애물이 있는 경우, (b) 초기 부근에 있는 경우 그리고 (c) 경로 상 2 개의 장애물이 있는 경우를 고려하였다.

장애물을 넘는 동작은 스테레오 카메라에 의해

구해진 장애물의 높이를, 로봇의 역기구학을 이용하여 순차적으로 윗부분을 닫고 서행한다. 그림 19의 첫 번째 궤적의 평균 정확도는 94.45%이며 끝점에서의 정확도는 92.12%의 정확도를 보인다. 두 번째 궤적의 평균 정확도는 91.13%이며 끝점의 정확도는 89.24%의 정확도를 보인다. 세 번째 궤적의 평균 정확도는 91.9%이며 끝점의 정확도는 90.4%이다.

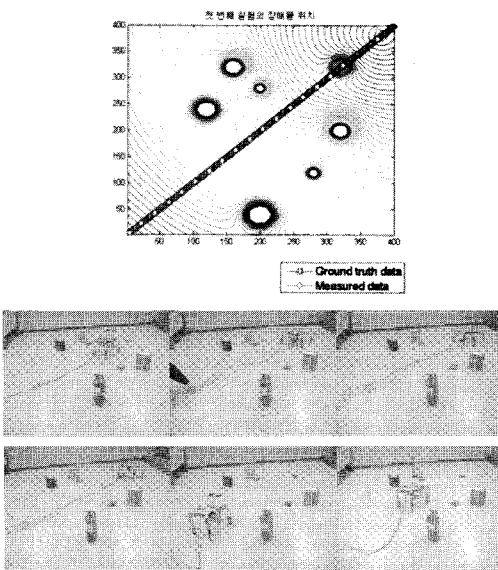


(c) 3rd Experiment

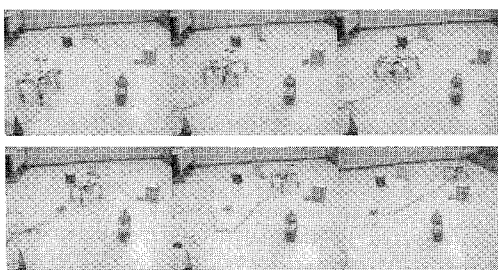
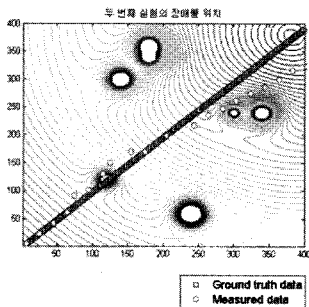
Fig. 19 Experiments of step over obstacles

**6.3 포텐셜 필드 방법과 넘기 동작의 혼합방법**

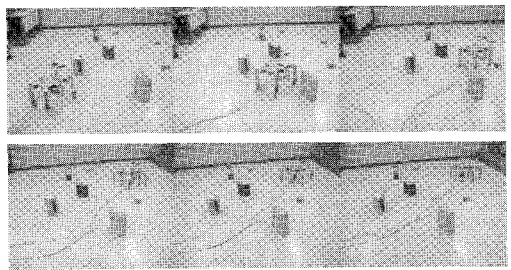
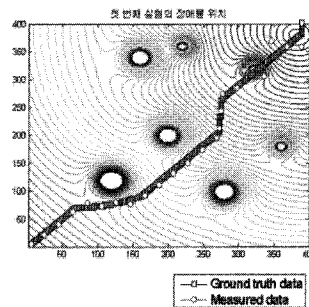
장애물을 넘을 수 있는 경우와 포텐셜필드 방법을 혼합하여 생성된 궤적을 실험한다. 이전 실험과 동일한 조건으로 각기 다른 위치의 장애물을 설치한 후 실험이 진행되었다. 그림 20은 로봇의 걸음새를 이용한 넘기 방법에 의하여 생성된 궤적을 보여준다.



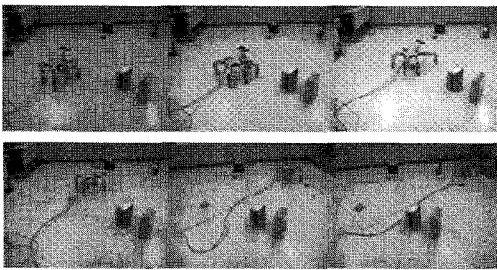
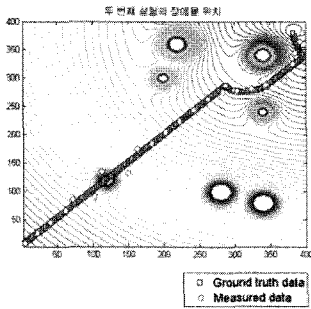
(a) 1<sup>st</sup> Experiment



(b) 2<sup>nd</sup> Experiment



(a) 1<sup>st</sup> Experiment



(b) 2nd Experiment

Fig. 20 Experiments with hybrid methods

첫 번째 궤적의 평균 정확도는 92.45 %이며 끝점에서의 정확도는 91.12 %의 정확도를 보인다. 두 번째 궤적의 평균정확도는 93.13 %이며 끝점의 정확도는 92.24 %의 정확도를 보인다.

### 7. 결론

본 논문에서는 스테레오 카메라를 이용하여 로봇이 장애물을 모델링하고 이 정보를 이용하여 장애물의 넘을 수 있는지에 대한 판단과 그에 대하여 궤적을 생성하는 실험을 하였다. 첫 번째 실험인 회피동작에서는 포텐셜 필드 알고리즘을 사용하였다. 이 알고리즘은 국부 최소점에 빠질 수 있기 때문에 이에 대한 예외처리를 해주어야 한다. 본 실험에서는 원의 모양으로 장애물을 근사화 해주었기 때문에 국부 최소점에 빠지지 않았다. 또한, 포텐셜필드 방법으로 생성된 궤적으로 로봇이 부드럽게 움직이는 것을 볼 수 있었다. 두 번째 실험은 로봇이 장애물을 넘어가는 경우로서, 장애물의 높이를 정확히 측정용 하여 로봇이 자연스럽게 넘어갈 수 있다. 장애물의 높이를 측정하기 위해서는 카메라의 보정결과가 정확해야 하므로 본 실험에서는 카메라 보정을 반복하여 정확성을 기하였다. 로봇의 역기구학과 보행 패턴을 이용하여 로봇이 장애물을 정확하게 넘을 수 있었다. 세 번

째 실험은 위의 두 가지 방법을 혼합하여 실험을 하였으며 각 부분의 장점만을 사용하여 적용을 하였다.

로봇이 장애물을 검출하고 궤적을 생성하여 추종하는데 있어서 92 %이상의 정확도를 보였다. 그러나 절대적인 위치를 판단할 수 없기 때문에 향후에는 시각적 표식(Visual marker)을 이용하여 절대 위치를 알아내야 할 것이다. 향후 보완해야 할 점은 스테레오 비전을 사용하는데 있어서 외부 조명에 의한 영향이다. 외부 광원의 위치가 바뀌면 장애물의 위치도 바뀌는 경우가 생기고, 이러한 문제를 해결하기 위하여 조명의 영향을 최소화 하는 알고리즘을 사용하여야 할 것이다. 또한 로봇이 장애물에 대하여 회피해갈 것인지 아니면 넘어갈 것인지에 대하여 단순한 수학적 판단보다는 퍼지 알고리즘 등과 같은 인공지능 알고리즘의 추가가 요구된다. 더불어 지형 및 장애물 인식의 오차에 의한 궤적 추종 오차를 보상하고 주행 중 갑자기 나타나는 동적 장애물을 감지하고 이를 극복하기 위한 실시간 처리가 차후 연구과제로 요구된다.

### 후 기

이 논문은 인하대학교의 지원에 의하여 연구되었음.

### 참고문헌

1. Kim, T. H., Choi, K. H. and Kim, Y. T., "A study on the static gait of small-scale quadruped robot," Journal of the Frontier Technology Innovation Center, Vol. 10, No. 1, pp. 81-98, 2001.
2. Fujita, M., "On Activating Human Communications with Pet-Type Robot AIBO," Proceedings of the IEEE, Vol. 92, No. 11, pp. 1804-1813, 2004.
3. Fukuoka, Y., Kimura, H., Hada, Y. and Takase, K., "Adaptive Dynamic Walking of a Quadruped Robot 'Tekken' on Irregular Terrain Using a Neural System Model," Proceedings of Robotics and Automation IEEE, Vol. 2, pp. 2037-2042, 2003.
4. Hodoshima, R., Doi, T., Fukuda, Y., Hirose, S., Okamoto, T. and Mori, J., "Development of TITAN XI: a quadruped walking robot to work on slopes," Proceedings of Intelligent Robots and Systems IEEE, Vol. 1, pp. 792-797, 2004.

5. Lee, C.-Y., Choi, H.-G., Park, J.-S., Park, K.-Y. and Lee, S.-R., "Collision Avoidance by the Fusion of Different Beam-width Ultrasonic Sensors," Proceedings of Sensors IEEE, pp. 985-988, 2007.
6. Jia, S., Yasuda, A., Chugo, D. and Takase, K., "LRF-Based Self-Localization of Mobile Robot Using Extended Kalman Filter," SCIE Annual Conference, pp. 2295-2298, 2008.
7. Gao, L.-F., Gai, Y.-X. and Fu, S., "Simultaneous Localization and Mapping for Autonomous Mobile Robots Using Binocular Stereo Vision System," International Conference on Mechatronics and Automation, pp. 326-330, 2007.
8. Zhu, M., Ge, Y., Huang, S. and Chen, W., "Stereo Vision Rectification Based on Epipolar Lines Match and Three Variables Projective Matrix," International Conference on Integration Technology IEEE, pp. 133-138, 2007.
9. Lim, H. S. and Park, H. W., "A Dense Disparity Estimation Method using Color Segmentation and Energy Minimization," International Conference on Image Processing IEEE, pp. 1033-1036, 2006.
10. Tsai, R., "A versatile camera calibration technique for high accuracy 3D machine vision metrology using off the shelf TV cameras and lenses," Journal of Robotics and Automation IEEE, Vol. RA-3, No. 4, pp. 323-344, 1987.
11. Faugeras, O. D. and Toscani, G., "The Calibration Problem for Stereo," Conference on Computer Vision and Pattern Recognition IEEE, pp. 15-20, 1986.
12. <http://www.ptgrey.com>
13. <http://www.bluetooth.com>
14. <http://www.atmel.com>
15. <http://www.robotis.com>
16. Fu, K. S., Gonzalez, R. C. and Lee, C. S. G., "Robotics," McGRAW-HILL, 1987.
17. Jung, S. H. and Lee, M. H., "Learning OpenCV: Computer Vision with the OpenCV Library," HongRung Publishing Company, 2007.
18. Kim, J. H., Choi, C. H., Song, J. B., Park, S. K. and Kim, M., "Building of Occupancy Grid Map of an Autonomous Mobile Robot Based on Stereo Vision," J. of KSPE, Vol. 19, No. 5, pp. 36-42, 2002.
19. Park, J. M., Looney, C. G. and Chen, H. C., "Fast Connected Component Labeling Algorithm Using A Divide and Conquer Technique," Conference on Computer and Their Applications, pp. 373-376, 2000.
20. <http://sourceforge.net/projects/opencvlibrary/>
21. Barraquand, J., Langlois, B. and Latombe, J., "Numerical potential field techniques for robot path planning," IEEE Transactions on Systems, Man and Cybernetics, Vol. 22, No. 2, pp. 224-241, 1992.