

# 소실점과 신경회로망을 이용한 이동 로봇의 위치 결정

°이 효진\*, 이 기성\*

홍익대학교 전기·제어공학과,  
서울 특별시 마포구 상수동 72-1 홍익대학교

## Determining the Position of a Mobile Robot Using a Vanishing Point and Neural Networks

\*Hyojin Lee, Keeseong Lee\*

Dept. of Electrical & Control Engineering, HongIk University.  
72-1, Sangsu-dong, Mapo-gu, Seoul, Korea

**Abstract** : During the navigation of mobile robot, one of the essential task is to determine the absolute position of mobile robot. In this paper, a method to determine the position of the camera using a vanishing point and neural networks without landmark is proposed. In determining the position of the camera on the world coordinate, there are differences between the real value and the calculated value because of uncertainty in pixels, incorrect camera calibration and lens distortion etc. This paper describes the solution of the above problem using BPNN(Back Propagation Neural Network) and experimental results show the capability to adapt for a mobile robot.

### 1. 서론

자율 주행 로봇에서는 주로 모터의 엔코더를 이용하여 현재 위치를 알아내고 초음파센서에 의해 장애물의 외부적인 환경을 인식한다. 하지만 정밀하게 모터를 제어하고 초음파센서를 이용해 인식한다 해도 물리적인 오차가 생기고 이 오차가 누적되면 최종 목적지에서는 심각한 오차를 발생시킬 수 있다. 이런 이유에 의해서 비전 시스템에 의해 이동 로봇의 절대 좌표 값을 보장해 주어야 한다. CCD 카메라로부터 얻어진 영상 정보를 이용하여 카메라의 위치 정보를 구하고자 하는 노력이 많은 사람들을 통해 이루어져 왔다. 여러 가지 방법이 있었지만 가장 널리 쓰이는 방법은 반복 기법(Iterative techniques)을 통하여 카메라 모델의 비선형 방정식을 구하는 방법이다. Haralick[1]는 사각형의 표식을 이용하여 Perspective Projection으로부터 Camera Parameter를 계산하는 방법을 개발하였고, Harwood[5]는 삼각형 표식을 이용하여 3차원 상의 물체의 위치를 구하는 연구를 하였으며, 그 외에도 원형이나 줄무늬 등의 표식을 이용하여 절대 좌표를 구하는 연구 등이 있어 왔다. 하지만 렌드

마크를 이용한 이동 로봇의 위치 결정시에 긴 복도 주행의 경우에 렌드 마크 사이에 이동 로봇이 위치해 있는 경우 이동 로봇의 위치 결정이 불가능하기에 이에 대한 대책이 요구되었다. 본 논문에서는 이동 로봇의 긴 복도 주행 시에 소실점을 추출하고, 신경회로망의 오류역전파(BPN)알고리즘을 사용하여 이동 로봇의 절대 좌표를 구하는 방법을 제안하고 실험을 통해 실제 적용가능함을 보였다.

### 2. 소실점 추출 알고리즘

다음 그림1과 그림2는 각각 홍익 대학교 제1공학관 6층의 복도를 CCD 카메라를 이용하여 Image를 획득한 원영상과 Sobel Operator를 사용하여 원영상으로부터 Edge를 찾아낸 영상이다.



<그림 1> 복도 영상의 원영상



<그림 2> Edge 추출 영상

### 2.1 직선 성분 찾기

Edge 영상으로부터 복도의 벽면의 직선 성분들을 찾기 위한 알고리즘이 필요하다. 본 논문에서는 형태학적(Morphological)인 구성 요소로부터 직선 성분을 찾아내는 방법을 사용하였다. 먼저  $8 \times 8$ 인 구성 요소 Window를 만들고 각각의 각도가  $11.25^\circ$ 의 Window 16개를 만들었다. 윈도우의 픽셀수가 8개이기 때문에 Edge 영상에서 픽셀수가 8개 이하인 직선 Segment는 모두 Noise로 간주되고 제거해 나간다. 모두 16개의 구성 요소 윈도우를  $B_k(k=0,1,\dots,15)$ 라 정의하고 각각의  $B_k$ 는 한 방향 성분의 직선을 찾아내는데 사용된다. Edge 영상을  $F(i,j)$ , Edge Segment로부터 구성 요소 윈도우를 거쳐 찾아낸 직선 성분의 영상  $A_k(i,j)$ 는

$$A = \bigcup_{k=0}^{15} A_k = \bigcup_{k=0}^{15} F(i,j) \circ B_k$$

$$k = 0, 1, \dots, 15; i, j = 0, 1, \dots, n \quad (1)$$

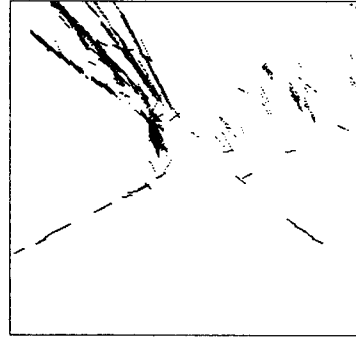
와 같이 정의하며 식(1)에서  $F(i,j) \circ B_k$ 는

$$F(i,j) \circ B_k = \sum \sum F(i+u, j+v) \cap B_k(u,v)$$

$$u = 0, 1, 2, \dots, 15; v = 0, 1, 2, \dots, 15 \quad (2)$$

와 같으며  $n$ 은 영상의 픽셀 갯수를 나타낸다.

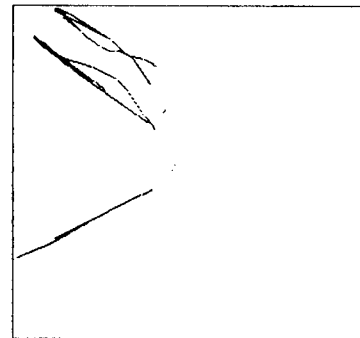
만약  $\sum \sum F(i+u, j+v) \cap B_k(u,v)$ 가 어떤 임계값보다 크면  $F(i+u, j+v) \cap B_k(u,v)$ 는 직선 성분 영상인  $A_k$ 로 옮겨지게 된다.



<그림 3> 직선 성분 추출 영상

### 2.2 Line Linking

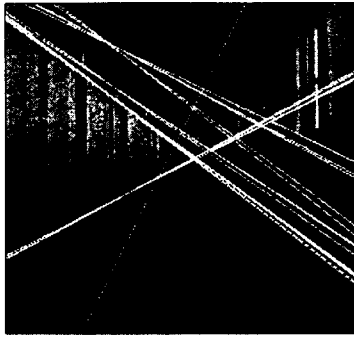
처음 직선 성분의 기울기를  $a_1$ 이라 하고 그 직선의 마지막 선분의 영상 면상의 좌표를  $p_1$ 이라 할 때, 다음 직선 성분은  $(p_1+1)$ 인 영상 면상의  $x$ 좌표에서 시작하여 아래로 Scan하며 또 다른 직선 성분을 찾는다. 같은 직선을 계속 찾는 것을 방지하기 위해 제한 조건을 만족하는 직선 성분의 처음 좌표를 구하게 된다. 두 번째로 찾은 직선 성분의 영상 좌표를  $p_2(i,j)$ 라 할 때 제한 조건은  $\{p_2(i-1,j) \neq 0, p_2(i-1,j-1) \neq 0, p_2(i-1,j+1) \neq 0\}$ 이어야 한다. 다음 직선 성분의 처음 좌표가  $p_2$ 라 할 때 선분  $\overline{p_1 p_2}$ 의 기울기  $a_2$ 를 구하여 처음 직선 성분의 기울기  $a_1$ 과 비교하여 어느 임계값보다 적으면 한 직선 성분으로 간주하게 되고,  $p_2$ 로부터 두 번째 직선 성분의 마지막 점을 Chain Code를 이용하여 찾아간다. 마지막 점의 좌표가  $p_3$ 라 할 때 선분  $\overline{p_2 p_3}$ 의 기울기  $a_3$ 와  $a_2$ 와 비교하여 임계값보다 적으면 선분  $\overline{p_1 p_2}$ 와 선분  $\overline{p_2 p_3}$ 는 한 직선 성분으로 Linking이 된다.



<그림 4> Line Linking 영상

### 2.3 Line Fitting

Line Linking을 하고 난 후에 각각의 직선들은 정확한 직선을 형성하지 못하고 오차를 동반하기에 Line Fitting과정이 필요하다. Line Fitting과정은 Least Square Estimation을 사용하여 각 직선들의 기울기를 구하였다. 다음은 Line Linking후에 Line Fitting을 하고 난 후의 영상에 대한 그림이다.



<그림 5> Line Fitting 영상

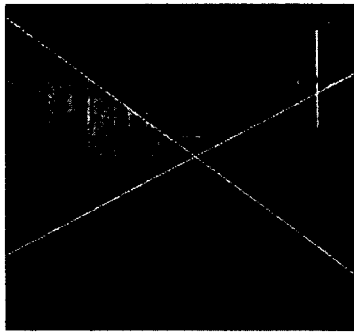
## 2.4 Line Verification

Line Fitting 후의 여러 직선들의 픽셀 집합을  $L_k^p(i,j)$  ( $P=1,2,3 \dots p$ ,  $K=1,2,3 \dots k$ )이라 정의하고, 이때  $k$ 는 한 직선을 이루는 픽셀의 수를 나타내고  $p$ 는 여러 직선들의 개수를 나타낸다. 2.1절의 Edge 영상  $F(i,j)$ 와 여러 직선  $L_k^p(i,j)$ 와의 Correlation 함수  $C_p$ 을 구하면

$$C_p = F(i,j) \otimes L_k^p = \sum F(i,j) \cap L_k^p(i,j), \quad (3)$$

$P = 0, 1, 2 \dots p$

이 되고 최대값  $\text{Max}(C_p)$ 를 구하면, 가장 정확한 소실점을 이루는 직선인  $L_k^p$ 을 구할 수 있다.



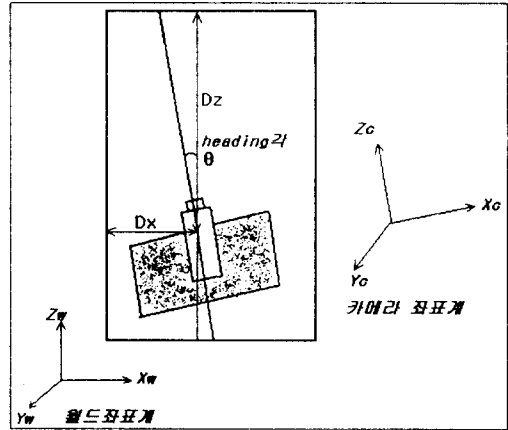
<그림 6> 소실점 추출 영상

## 3. 소실점을 이용한 위치결정 알고리즘

본 논문에서는 평지인 실내 복도를 주행하는 경우이고, 카메라가 이동 로봇에 고정되어 좌우로만 회전하는 경우만 고려를 하였으므로 카메라 Parameter중에서 월드 좌표 계의 원점에서 카메라 좌표 계의 원점까지 X축 상으로의 거리  $dx$ 와, 월드 좌표 계의 원점에서 카메라 좌표 계의 Z축 상의 거리  $dz$ , 월드 좌표 계의 Y축을 중심으로 카메라 좌표 계가 회전한 Pan각  $\theta$ 만 알면 월드 좌표 계에 대한 카메라의 절대 위치를 구할 수 있다.

### 3.1 Pan각 $\theta$ 결정

3차원 상에 평행한 두 직선을 영상 면에 투영하였을 때, 두 직선과 영상 면이 직각을 이루지 않는다면 영상



<그림 7> 복도에서의 월드좌표계와 카메라좌표계

면상의 투사된 두 직선은 한 점(소실점)에서 만나게 되고, 이 소실점은 평행한 직선의 어느 한 점의 위치에 상관없이 그 직선의 z축상의 방향 벡터와 X,Y축의 방향 벡터에 따라 변하게 된다. 만약 카메라 좌표 계와 월드 좌표 계의 회전각이 없다면 카메라의 위치에는 상관없이 소실점은 항상 영상 면상의 중앙에 위치하게 된다. 카메라의 시축인 Z축과 월드 좌표계의 Z축과의 각  $\theta$ 는

$$\theta = \cos^{-1} \left( \frac{f}{\sqrt{u_x^2 + v_x^2 + f^2}} \right) \quad (4)$$

와 같고 영상면상의 소실점인  $(u_x, v_x)$ 이 실제 컴퓨터 메모리 화면에 나타나는 점은 렌즈 왜곡과 영상 획득 과정에서 오차로 비선형 특성을 나타내게 된다. 이런 비선형특성을 본 논문에서는 BPN을 사용하여 보정을 하였다. 즉, 화면 중앙과 소실점과의 영상 면상의 X좌표 값의 차이를 입력으로 하고, 출력으로는 Pan각  $\theta$ 를 1:1 매핑하여 입력 뉴런은 1개, 은닉층 뉴런 20개, 출력 뉴런 1개의 신경망을 구성하였다.

### 3.2 입력 수정 알고리즘

BPN으로 학습을 시키는 경우, 실제의 출력과 원하는 출력 사이에서의 오차가, 많은 학습이 끝난 후에도 남게 되어 이것으로부터 최종 위치 결정에 대한 오차가 발생하게 되므로 원하는 출력을 출력으로 발생하는 최적의 원하는 입력을 구하여, 실제 입력을 원하는 입력으로 수정하여 결과를 출력하였다. BPN 학습 알고리즘으로 학습이 끝난 후에 실제 출력과 원하는 출력과의 출력층에서의 오차  $\delta_k$ 는

$$\delta_k = (t_k - o_k) f'(net_k) \quad (5)$$

로 계산되어지고, 이로부터 중간층에서의 오차  $\delta_j$ 는

$$\delta_j = f'(net_j) \sum \delta_k w_{jk} \quad (6)$$

와 같다. 이 중간층 오차로부터 입력 값을 원하는 출력에 근접하는 출력을 내도록 직접 수정하면 입력 값이 정규 값인 (-1,1)범위를 넘어갈 수 있으므로 tanh함수를 사용하여 새로운 입력층을 만든다. 기존의 입력  $I_j$ 로부터 새로운 입력값  $I_j^* = I_j / (1 + |I_j|) = \tanh(I_j)$  의 식

으로부터

$$I_i^* = f^{-1}(I_i) = \text{atanh}(I_i) \quad (7)$$

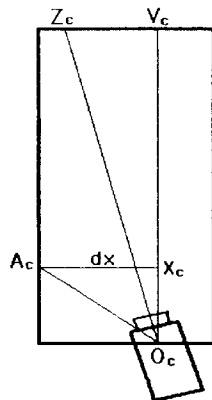
와 같이 구할 수 있다. 중간층 오차  $\delta_i$ 로부터 새로 수정된  $I_i^{*+1}$ 은

$$I_i^{*+1} = I_i^{*0} + \Delta I_i^*, \Delta I_i^* = c \cdot \delta_i \quad (8)$$

로부터 구해진다. 이때  $c$ 는 상수로써 입력 수정의 스텝 수를 결정하는 학습 계수이다. 몇 차례의 반복을 통해 원하는 출력과 실제 출력 사이의 오차가 만족스러운 수준으로 줄어들면 입력 수정 알고리즘이 끝나게 되고, 이 알고리즘에 의해 새로 생성된  $I_i$ 가 최적의 입력 값이 되고 각 입력 패턴에 대해서  $\Delta I_i^*$ 의 벡터는 저장된다. 실제 출력을 구할 때 어느 입력 값에 대해 최적의 입력 값에 대한 패턴에 대해 실제 입력 값과의 유클리디안 거리를 구하여 최소 거리에 해당되는 입력 패턴에 대해  $\Delta I_i^*$  벡터로 입력값을 수정하여 출력을 내게 된다.

### 3.3 X축의 거리 Dx 결정

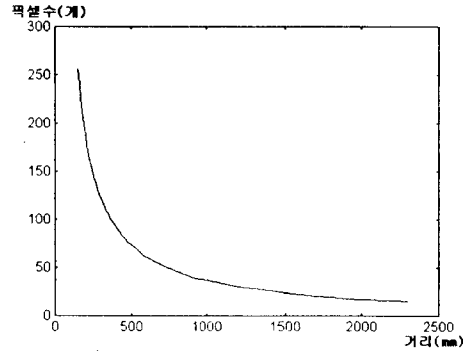
월드 좌표 계 원점으로부터 카메라 좌표 계 원점까지의 X축상로의 거리  $dx$ 를 구하기 위해서는 먼저 Pan각과 카메라 화면상의 맨 가장자리에 해당되는 곳으로의 거리 정보가 필요하다. 그림8과 같이 카메라가 복도의 어느 한 지점을 보고 있을 때의 시축을  $\overline{O_c Z_c}$  선분이라 할 때, 소실점이 위치해 있는 곳은 카메라가 복도와 평행으로 이루는 곳에 위치해 있으므로 카메라가 월드 좌표 계에 놓인 지점이 되며 이때 소실점의 X성분 위치를  $V_c$ 라 하면  $\angle \overline{O_c Z_c} \overline{O_c V_c}$ 가 월드 좌표 계와 카메라 좌표 계의 Y축 회전각인 Pan각이 된다. 카메라의 가지 각도는 그림에서처럼 카메라의 시축으로부터 화상으로 획득되는 각도로서  $\angle \overline{O_c A_c} \overline{O_c Z_c}$ 에 해당되며 실험에 의해 구한 렌즈의 가지 각도는  $10.52^\circ$ 이다.



<그림 8> 렌즈의 가지각

선분  $\overline{O_c A_c}$ 에 해당하는 실제 거리를 구하면  $dx$ 를 구하게 된다. 선분  $\overline{O_c A_c}$ 에 해당하는 실제 거리는 화상 메모리의 0번째 행의 Vanishing Line의 길이가 되며

LSE(Least Square Estimation)을 사용하여 구한 두 직선의 기울기의 y절편 값의 차가 된다. 화상 면상에 나타나는 길이와 실제 거리는 계산에 의해 구할 수도 있지만 비선형 특성의 렌즈 왜곡과 Grabber에서 영상 획득 시의 왜곡으로 인해 실제값과 다르게 나온다. 그러므로 이를 BPN을 사용하여 보정을 하였다. 먼저 좀더 정확한 값을 얻기 위해 길이 6cm의 정사각형 표식을 이용하여 화상 면상에 나타나는 길이와 실제 거리를 1:1 매핑을 하였다. 두 값 사이의 매핑관계는 그림9와 같다.



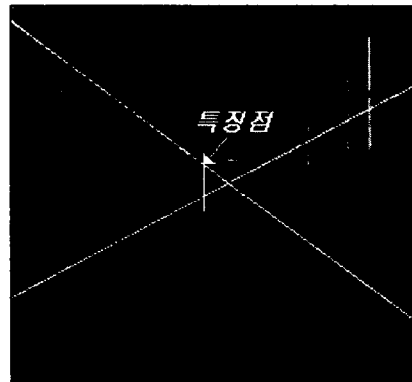
<그림 9> 영상의 픽셀수와 거리와의 특성 곡선

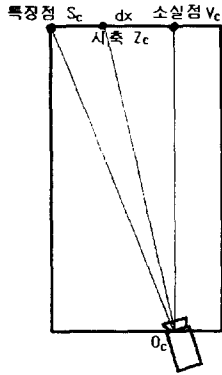
앞에서 구한  $\overline{O_c A_c}$ 와  $\angle A_c O_c V_c$ 로부터  $dx$ 를 구할 수 있다.

$$dx = \overline{O_c A_c} \cdot \sin(\pi - \angle A_c O_c V_c) \quad (9)$$

### 3.4 Z축의 거리 Dz 결정

그림10과 같이 복도의 끝에 해당되는 곳을 기준점으로 삼았다. 좌측 복도의 끝에 해당되는 지점은 항상 소실점의 좌측에 위치해 있으므로 소실점으로부터 좌측으로 Scan하며 Edge영상으로부터 얻은 영상과 Matching을 시켜 처음 나오는 수직 성분의 Edge를 찾으면 좌측 복도의 끝에 해당하는 특징 점을 찾을 수 있게 된다.





<그림 10> 복도의 특징점

특징점  $S_c$ 와 소실점  $V_c$ 사이의 거리  $dx$ 와  $\angle Z_c O_c V_c$ 를 알고 있으므로  $\angle S_c O_c Z_c$ 를 구하면  $dz$ 거리  $\overline{V_c O_c}$ 를 구할 수 있다.  $\angle S_c O_c Z_c$ ,  $\angle Z_c O_c V_c$ 와  $dx$ 정보로부터 복도 끝에서부터 카메라까지의 Z축상오차의 거리  $dz$ 를 구하면 식(10)과 같다.

$$dz = \frac{dx}{\tan(\angle S_c O_c Z_c + \angle Z_c O_c V_c)} \quad (10)$$

#### 4. 실험 결과

실험은 홍익 대학교 제1공학관 6층 실내복도 K609호 앞에서부터 K616호 앞까지 총거리 28m에 한해서 Cart 위에 카메라를 고정하고 1m씩 이동하며 영상을 획득하였으며 복도의 총길이는 58.3m이며 너비는 2.4m이다. 복도 벽면으로부터 각각 1m, 2m인 지점에서 영상을 획득하고, 총 56개의 Grid로부터 거리를 측정하였다. 표1과 표2는 복도 끝으로부터 58.3m 거리에서 1m씩 이동하면서 dx거리는 각각 1m, 2m인 지점에서 영상을 획득하였을 때의 실험 결과이다.

실험 결과 dx에 대한 오차는 대개 5% 미만으로 절대 오차가 5cm 이내로 발생하여 비교적 좋은 결과를 보인 것으로 보아 방향 각에 대한 오차는 크지 않으며, 소실점을 정확하게 찾았음을 알 수 있다. 하지만 dz에 대한 최대 절대 오차는 4m까지 발생하였다. 이 오차는 주로 센서 해상도에 대한 오차에 기인한다. 이는 결국 연속된 대상 물체의 모습을 잘게 나눔으로 인하여 경계선의 수치가 오차를 남기게 되는데 이는 결국 고해상도의 CCD 센서와 기억 소자가 개발되어 실용화되어야 해결될 수 있다.

| Z축 거리 (m) | X축 측정 거리  |        | Z축 측정 거리 |        |
|-----------|-----------|--------|----------|--------|
|           | 측정결과 (cm) | 오차 (%) | 측정결과 (m) | 오차 (%) |
| 58.3      | 101.38    | 1.38   | 58.10    | 0.34   |
| 57.3      | 102.77    | 2.77   | 53.02    | 7.46   |
| 56.3      | 97.52     | 2.48   | 52.34    | 7.03   |
| 55.3      | 100.87    | 0.87   | 52.87    | 4.39   |
| 54.3      | 99.46     | 0.54   | 53.69    | 0.73   |
| 53.3      | 102.52    | 2.52   | 54.50    | 2.25   |
| 52.3      | 95.34     | 4.66   | 48.24    | 7.76   |
| 51.3      | 99.87     | 0.13   | 53.30    | 3.89   |
| 50.3      | 104.73    | 4.73   | 52.33    | 4.03   |
| 49.3      | 99.24     | 0.76   | 49.95    | 1.31   |
| 48.3      | 96.06     | 3.94   | 47.85    | 0.93   |
| 47.3      | 106.25    | 6.25   | 48.38    | 2.28   |
| 46.3      | 95.74     | 4.26   | 48.23    | 4.00   |
| 45.3      | 102.62    | 2.62   | 43.37    | 4.26   |
| 44.3      | 100.47    | 0.47   | 41.97    | 5.55   |
| 43.3      | 102.49    | 2.49   | 42.46    | 1.93   |
| 42.3      | 105.85    | 5.85   | 43.85    | 3.66   |
| 41.3      | 96.32     | 3.68   | 44.72    | 8.28   |
| 40.3      | 97.40     | 2.68   | 40.65    | 0.86   |
| 39.3      | 108.50    | 8.50   | 39.05    | 0.63   |
| 38.3      | 98.17     | 1.83   | 38.41    | 0.28   |
| 37.3      | 102.63    | 2.63   | 36.73    | 1.52   |
| 36.3      | 103.54    | 3.54   | 37.13    | 3.11   |
| 35.3      | 103.01    | 3.01   | 36.16    | 2.43   |
| 34.3      | 94.21     | 5.79   | 38.39    | 11.92  |
| 32.3      | 102.35    | 2.35   | 34.31    | 6.22   |
| 31.3      | 105.46    | 5.46   | 29.87    | 4.56   |

<표1> X축상의 거리가 1m일때의 측정결과 및 오차

영상 정보가 Pixel 단위로 표현됨으로 인해 발생하는 dz 오차를 정량적으로 분석해 보면 표1에서 dz의 실제 거리가 55.3m, dx의 실제 거리가 1m일 때 측정 결과는 dx가 100.87cm로 비교적 정확한 결과가 나왔음에도

| Z축<br>거리<br>(m) | X축 측정 거리     |           | Z축 측정 거리    |           |
|-----------------|--------------|-----------|-------------|-----------|
|                 | 측정결과<br>(cm) | 오차<br>(%) | 측정결과<br>(m) | 오차<br>(%) |
| 58.3            | 200.77       | 0.38      | 60.79       | 4.27      |
| 57.3            | 198.16       | 0.92      | 56.29       | 1.76      |
| 56.3            | 197.70       | 1.15      | 57.75       | 2.57      |
| 55.3            | 205.02       | 2.51      | 53.57       | 3.18      |
| 54.3            | 199.29       | 0.35      | 56.92       | 4.82      |
| 53.3            | 205.35       | 2.67      | 55.61       | 4.33      |
| 52.3            | 200.54       | 0.27      | 49.84       | 4.70      |
| 51.3            | 196.76       | 1.62      | 50.16       | 2.22      |
| 50.3            | 202.39       | 1.19      | 51.23       | 1.84      |
| 49.3            | 201.22       | 0.61      | 47.52       | 3.61      |
| 48.3            | 200.04       | 0.02      | 46.97       | 2.75      |
| 47.3            | 198.08       | 0.96      | 46.34       | 2.03      |
| 46.3            | 198.62       | 0.69      | 49.54       | 6.99      |
| 45.3            | 200.91       | 0.45      | 42.75       | 5.62      |
| 44.3            | 199.07       | 0.46      | 40.87       | 7.74      |
| 43.3            | 208.34       | 4.17      | 42.61       | 1.59      |
| 42.3            | 203.47       | 1.73      | 45.74       | 8.13      |
| 41.3            | 196.84       | 1.58      | 43.53       | 5.39      |
| 40.3            | 204.34       | 2.17      | 40.08       | 0.54      |
| 39.3            | 202.38       | 1.19      | 42.85       | 9.03      |
| 38.3            | 201.87       | 0.93      | 40.35       | 5.35      |
| 37.3            | 199.14       | 0.43      | 34.37       | 7.85      |
| 36.3            | 198.52       | 0.74      | 38.24       | 5.34      |
| 35.3            | 201.78       | 0.89      | 35.94       | 1.81      |
| 34.3            | 205.92       | 2.96      | 33.74       | 1.63      |
| 32.3            | 204.18       | 2.09      | 31.81       | 1.51      |
| 31.3            | 197.95       | 1.02      | 32.02       | 2.30      |

표 2. X축상의 거리가 2m일때의 측정 결과 및 오차

불구하고 dz은 52.87m로 절대 오차가 2.43m가 발생하였다. 실험 측정 결과 방향 각은 0.657°가 나왔으며 영상면의 화면 중앙으로부터 특징 점인 복도 끝점에 해당되는 픽셀수는 5가 나왔는데 만약 실제의 값이 4.5라고 가정할 때 이에 대한 dz을 구하여 보면, 먼저 픽셀수가 4.5일 때의 각은 0.391°가 되며 이에 해당하는 dz은 55.09m가 나옴으로 실제 거리에 근접한 결과가 된다. 결국 픽셀 해상도의 오차로 인해 실제 거리에서 2.2m의 오차가 나왔음을 알 수 있다.

### 3. 결론

본 논문에서는 소실점과 신경회로망을 이용하여 복도, 주행중인 이동 로봇의 절대 위치를 찾는 문제에 대한 새

로운 방법을 제안하였다. 기존의 랜드 마크를 이용하여 위치 결정을 하는 경우 랜드 마크가 훼손되었을 경우나 혹은 이동 로봇이 랜드 마크를 보지 못하는 경우에는 위치 결정을 제대로 못하는 문제점이 존재한다. 이에 랜드 마크 없이 이동 로봇이 실내 복도를 주행할 경우에 소실점을 찾는 알고리즘과 소실점으로부터 위치 결정, 보정하는 알고리즘을 제안하고 실험을 통하여 위치 결정의 결과를 알아보았다. 실험 결과 X축의 거리 측정이나 방향각의 경우에는 비교적 정확한 결과를 내었으나 Z축의 거리 측정에서의 오차가 많이 났다. 이는 센서 해상도로 인한 오차에 기인하고 측정시 한 픽셀 차이에 따라 실제 Z축으로의 오차는 최대 3m이상 나는 결과를 초래하였다. 본 연구의 결과는 신경회로망을 이용하여 이동 로봇의 절대 위치를 찾는 데에 적용 가능성을 보여 주고 있다.

### 참고문헌

- [1] Robert M. Haralick, "Determining camera parameters from the perspective projection of a rectangle", *Pattern Recognition*, Vol.22 No.3, pp.225-230, 1989.
- [2] Zen Chen, Din-Chang Tseng and Jenn-Yee Lin, "A Simple Vision Algorithm for 3-D Position Determination using a Single Calibration Object", *Pattern Recognition*, Vol.22 No.2, pp.173-187, 1989.
- [3] Gideon P. Stein, "Internal Camera Calibration using Rotation and Geometric Shapes", *The Degree of Master of Science at the M.I.T.*, Febrary 1993.
- [4] Reimar K. Lenz, Roger Y. Tsai, "Techniques for Calibration of the Scale Factor and Image Center for High Accuracy 3D machine Vision Metrology", *IEEE Robotics and Automation*, pp. 68-75, 1987.
- [5] Robert M. Haralick, *Computer and Robot Vision II*, Addison-Wesley Publishing Company, 1993.
- [6] Xavier Lebégue "Significant Line Segments for an In door Mobile Robot", *IEEE Robotics and Automation*, Vol. 9, No. 6, pp. 801-815, 1993
- [7] Dorst, Smeulders, "Discrete Straight Line Segment", *Ph.D. Thesis, Technological Univ.*, Febrary, 1994.