

이동로봇의 실시간 위치결정을 위한 수직선과 점 대응 알고리즘

Fast Algorithm for Location Determination of Mobile Robot: Vertical Line to Point Correspondences

김재희 조형석

한국과학기술원 생산공학과
(Dept. of Production Eng. KAIST)

ABSTRACT

It is one of the essential task to determine the absolute location of mobile robot during its navigation. In this paper we propose an algorithm to calculate the distance and orientation of camera from landmark through the visual image of stripe typed landmark. Exact closed form solution of camera location is obtained with the correspondences from vertical line on mark plane to the intersection point of projected line with horizontal axis of image plane. It needs only one line image information, so that location determination can be processed in real time.

I. 서론

근세기에 이르러 로봇은 고정된 위치에서 주어진 과업을 수행하는데서 진일보하여 이동기능이 추가됨으로서 그 활동영역과 수행작업이 확대되었다. 특히, 공장자동화를 위한 무인반송차 (AGV, Automated Guidance Vehicle), office 빌딩내에서의 문서전달 로봇, 병원에서의 간호 로봇 (Nursing Robot) 및 안내로봇 등 여러 응용분야에서 이동로봇의 연구개발이 이루어지고 있다.

이동로봇을 원하는 경로대로 주행시키기 위해서는 여러가지 제어 기술이 필요하지만 그중에서 특히, vehicle의 현 위치를 판단하는 것이 필수적인 요소기술 중의 하나이다. 이와같은 vehicle의 주행을 원만히 수행하기 위하여 궤도식방법을 사용할수있다. 즉, 원하는 궤도상에 선로를 매설하여 전차기를 이용하여 경로를 유도할수있고[1] 반사레이프를 경로에 따라 부착하고 광학적으로 반사광을 기용하여 궤도 주행을 할수도 있다[2]. 그러나 이 경우 궤도를 변경할 때마다 유도선을 새로 배치해야하고 바닥에설치된 유도선은 반복 주행으로 인하여 손상받기 쉬운 단점을 가지고 있다. 반면 무궤도식(free ranging)으로 자유주행할 경우 원하는 경로를 쉽게 바꿀수 있고 유연하게 주행할수 있지만 vehicle의 현 위치를 정확히 알고있어야 하는 전제조건을 가지고 있다. 이와 같은 자유주행 로봇의 위치를 알아내기위하여 wheel encoder를 이용한 dead reckoning 방식[3], 등대항법시스템[4], 자이로를 이용한 관성항법 시스템[5] 등이 연구되었으나 부정확하거나 가격이 비싸거나 군사용이라서 일반적으로 실용화 되지 못하였다.

최근에 들어 정해진 위치에 랜드마크를 간헐적으로 설치하고 이 마크를 카메라를 통하여 얻은 영상으로 부터 로봇의 위치 정보를 얻어내는 방법을 많이 연구하고 있다. 본 연구에서는 마크평면상의 수직선이 영상평면에 투영되어, 이 투영된 직선이 영상좌표계의 수평축과 만나는 교점과 마크 수직선간의 대응성을 이용하여 로봇의 위치정보를 폐형식의 해 (closed form solution)로 도출하는 알고리즘을 고안하였다.

카메라의 영상을 이용하여 위치를 결정하는 연구는 이동로봇, 항공기 위치결정, 로봇 캘리브레이션 등의 분야에서 오랫동안 연구되어 왔다. 1981년 Fischer & Bolles [6] 가 3차원 좌표가 알려진 m개의 점에 대한 카메라 위치를 기용하여 카메라의 위치를 결정하는 문제(camera location determination problem)를 정의한 이래 Keller & Twinkel [7] 은 비선형 최소자승법으로 이문제를 장황하게 풀었으며, anapathy [8] 는 기준 좌표계상에서 실제 object 포인트와 영상면상의 포인트간의 좌표변환 행렬이 실험적으로 결정되어

주어졌을때 카메라의 이동벡터와 회전각을 각각 closed form으로 구하는 방법을 제안 하였다. 또한 Lenz & Tsai [9] 는 로봇 팔을 한 조인트씩 움직이면서 로봇 calibration 을 수행하는 방법을 제안하였고 1990년 Liu & Huang [10] 는 line matching을 이용하여 카메라의 위치를 결정하는 방법을 제시하였다.

그러나 이와 같은 방법은 이동로봇의 위치를 산출하는 방법으로는 적합하지 못하다. 즉, 이동로봇의 위치 산출은 매우 빠르게 수행되어 로봇 주행제어에 반영되어야 하는 제약 조건이 있고, 컴퓨터가 로봇 위에 탑재되어야 하므로 중형컴퓨터가 아닌 PC급 컴퓨터로 실시간 판정이 가능 하여야 하기 때문이다. 또한 평지를 주행하는 이동로봇의 위치 정보는 거리좌표 (x,y)와 로봇의 heading angle 만 필요하므로 모든 정보를 다 구할 필요가 없다. 이동로봇의 실시간 위치 산출을 위하여 1987년 Kabuka & Arenas [11] 는 마크평면에 원을 그리고 이것을 영상처리 하였다. 즉, 마크와 카메라는 동일한 높이에 위치한다고 가정하고, 영상에 맺힌 타원의 장축과 단축의 길이를 하드웨어를 사용하여 빠른 계산을 수행하였다. 그런데 영상면에 투사된 찌그러진 타원을 표준타원으로 가정 하였고, 마크식별을 위하여 마크상에 별도의 bar code를 추가하였다. 또한 마크와 카메라는 항상 동일한 높이에 있어야 한다는 제한조건이 따른다.

본 연구에서는 줄무늬 형태의 마크를 카메라를 통하여 얻은 영상으로 부터 마크와 카메라 간의 거리 벡터와 방위각을 산출하는 알고리즘을 제안하였다. 이 알고리즘은 마크평면상의 수직선이 영상평면에 투영되어 영상면의 가로축과의 교점을 구함으로써 위치정보를 얻을수 있는데, 이 위치정보가 폐형식(closed form)으로 도출될 수 있어서 계산시간이 매우 짧고 영상대이터도 중심행 한줄만 읽어서 처리하므로 영상처리도 실시간으로 가능하다. 뿐만 아니라 마크형상이 수직선 형태로 되어있어 교차비 (cross ratio)를 이용하여 마크의 식별이 가능하고, 카메라의 높이 편차에 무관하게 원하는 위치정보를 얻을 수 있다.

II. 시스템 개요

이동로봇이 주행할 경우 로봇 시스템, 카메라시스템과 랜드마크가 그 구성요소로 되어있으며 로봇의 위치 산출을 위해서는 각각의 좌표계가 정의되어야 한다. 우선 본연구에서 사용되는 좌표계를 정의 하기로 한다. world 좌표계 { W }는 로봇 이동 공간에서 기준이 되는 절대좌표계이며 vehicle좌표계 { V }는 vehicle의 몸체에 고정된 것이고 camera 좌표계 { C }는 카메라 렌즈의 중심에 원점을 두고있다.

mark 좌표계 { M }는 마크를 설치한 곳에서 마크평면에 고정된 좌표계이다.

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \quad (3)$$

공장 자동화를 위한 무인 반송차를 비롯하여 안내로봇 등 일반적인 이동로봇은 현 시점의 로봇의 절대좌표와 방향각(heading angle)을 알아야만 원하는 경로를 정확하게 주행할 수 있다. 그림1에서 보는 바와 같이 원좌표계 { W }로부터 로봇의 좌표계 { V }간의 변환행렬을 알아내는 것이 연구의 목표이다. 이를 성취하기 위하여 정해진 위치에 마크를 설치하고 로봇위에 탑재된 카메라를 통하여 얻은 영상으로 부터 마크와 카메라 좌표계 간의 변환행렬을 도출하여 이미 알고있는 카메라와 로봇 몸체간의 변환행렬을 도입하여 결국 로봇의 절대좌표와 방향각을 구할수가 있다. 즉

$${}^W T = {}^W T ({}^V T \begin{matrix} C \\ M \end{matrix} T)^{-1} = {}^W T \begin{matrix} C \\ M \end{matrix} T \begin{matrix} V \\ T \end{matrix} \quad (1)$$

이된다 여기서

T: world 좌표계 {W}와 vehicle 좌표계 {V} 간의 변환행렬
 T: world 좌표계 {W}와 mark 좌표계 {M} 간의 변환행렬
 T: vehicle 좌표계 {V}와 camera 좌표계 {C} 간의 변환행렬
 T: camera 좌표계 {C}와 mark 좌표계 {M}간의 변환행렬이다
 그런데 표식마크의 절대좌표와 방위각 (orientation)은 알고 있는 값이므로 변환행렬 ${}^W T$ 는 정해지고, 로봇 위에서 카메라의 위치와 자세 또한 설계시 이미 정해진 것이기 때문에 변환행렬 ${}^C T$ 도 정해진 값이다. 따라서 카메라 좌표계와 마크좌표계간의 변환행렬 ${}^M T$ 만 다음장에서 제안하는 방법으로 구해내면 로봇의 위치와 방위각은 산술적으로 계산된다. 한편, 공장내부와 건물 내부등 바닥이 고르지 않음으로 주행하는 이동 로봇은 수직방향의 상하이동이 없으므로 로봇의 위치좌표(x,y)와 방향각(heading angle)만 알면 원하는 경로를 주행할 수 있다. 그래서 본 연구에서는 이 세가지 정보를 알수있는 보다 빠른 알고리즘을 제시하였다.

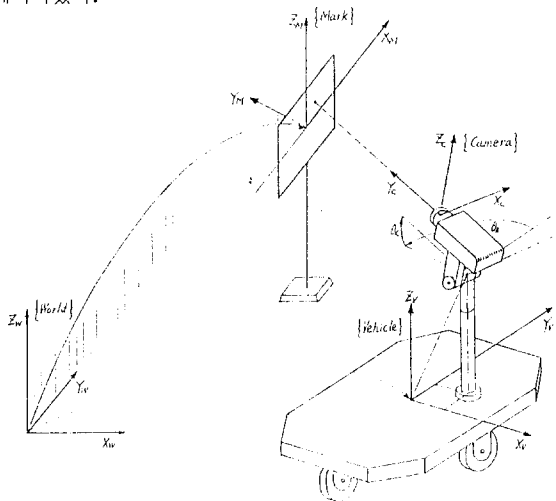


그림 1. 이동 로봇의 좌표계

이며, 여기서 P는 투시변환행렬 (perspective transformation matrix)이고 ${}^i V$ 와 ${}^i V_p$ 는 각각 영상좌표계를 기준으로 본점의 좌표이다. 또한 image plane를 기준으로 본 임의의 점 ${}^i V$ 와 마크 평면 좌표계 { M }를 기준으로 본 점 ${}^M V$ 와의 관계식은

$${}^i V = {}^i M T \begin{matrix} M \\ V \end{matrix} \quad (4)$$

이므로

$${}^i V_p = P \begin{matrix} i \\ M \end{matrix} T \begin{matrix} M \\ V \end{matrix} \quad (5)$$

가 된다. 여기서 ${}^i M T$ 는 영상좌표계 { I }와 마크좌표계 { M }간의 좌표변환 행렬로서 다음과 같이 정한다.

$${}^i M T = \text{TRANS}(D_x, D_y - f, D_z) \text{ROT}(Z_i, \phi_z) \text{ROT}(Y_i, \phi_y) \text{ROT}(X_i, \phi_x) \quad (6)$$

여기서 (D_x, D_y, D_z) 는 카메라 좌표 원점에서 마크좌표 원점까지의 거리 벡터이고, ϕ_z 는 pan angle, ϕ_x 는 tilt angle, ϕ_y 는 roll angle이다. 마크 평면상의 object point V가

$${}^M V = (X_m, 0, Z_m)^T \quad (7)$$

로 정해지면 식(3), (6)과 (7)을 식(5)에 대입하여 영상면상의 image point V_p 의 성분을 구하면

$$X_p = f \frac{C_{\phi_y} C_{\phi_z} X_m + (C_{\phi_x} S_{\phi_y} C_{\phi_z} + S_{\phi_x} S_{\phi_z}) X_m + D_x}{C_{\phi_y} S_{\phi_z} X_m + (C_{\phi_x} S_{\phi_y} S_{\phi_z} - S_{\phi_x} C_{\phi_z}) Z_m + D_y} \quad (8)$$

$$Z_p = f \frac{-S_{\phi_y} X_m + C_{\phi_x} C_{\phi_y} Z_m + D_z}{C_{\phi_y} S_{\phi_z} X_m + (C_{\phi_x} S_{\phi_y} S_{\phi_z} - S_{\phi_x} C_{\phi_z}) Z_m + D_y}$$

가 된다.

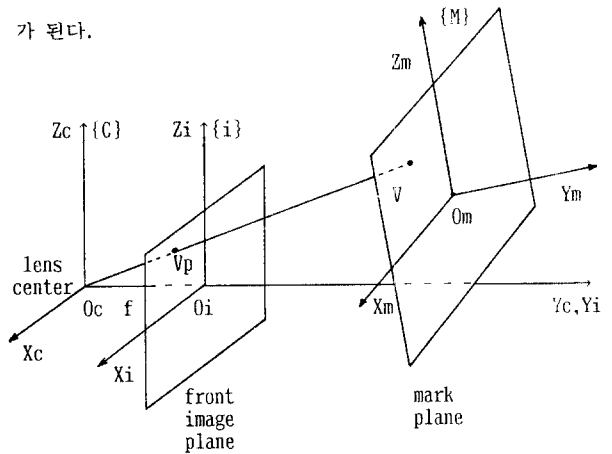


그림 2. 카메라 좌표 모델

B. 수직선과 점과의 대응

object와 image와의 맵핑을 이용하기 위하여 3차원 직선과 영상면의 2차원 직선의 대응이나 3차원 점과 2차원 점의 대응 등은 이미 많이 사용되고 있는 방법이다. 본 연구에서는 수직선과 점과의 대응관계를 통하여 이동로봇의 위치 정보를 얻고자 한다. 즉 마크평면상에 수직선을 표시하고 이것이 영상평면 상에서 수평중심축 (X_p 축)과의 교점을 구하여 마크 평면의 수직선과 영상 평면의 교점을 대응시킨다는 것이다. 마크평면상에서 수직선 ($X_m = X_{mi}$)을 생각해보자 이직선은 영상평면상에서

$$Z_p = A X_p + B \quad (9)$$

인 사선으로 나타낸다. 여기서 A 와 B 는 $D_x, D_y, D_z, \phi_x, \phi_y, \phi_z$ 의 함수로 표현된다. 이 사선이 영상 평면의 수평 중심축 (X_p)과 만나는 교점을 구하면

$$X_{p1} = f \frac{(C\phi_x S\phi_y C\phi_z + S\phi_x S\phi_z)(-S\phi_y X_{mi} + D_z) - (C\phi_y C\phi_z X_{mi} + D_x) C\phi_x C\phi_y}{(C\phi_x S\phi_y S\phi_z - S\phi_x C\phi_z)(-S\phi_y X_{mi} + D_z) - (C\phi_y S\phi_z X_{mi} + D_y) C\phi_x C\phi_y} \quad (10)$$

가 된다. 그런데 일반적으로 평지를 주행하는 이동로봇의 위치결정을 위하여 필수적으로 필요한 정보는 거리벡터(D_x, D_y)와 방위각(ϕ_z)뿐이며 이동로봇가 수직상방으로 미소이동하거나 좌우 전후로 기울어 지는 정보는 부수적이며 함해에 소용되지 않는다. 이와 같은 평지주행 이동로봇의 환경조건이 있기 때문에 마크는 지면과 수직으로 설치하고 이동로봇상의 카메라는 팬각(pan angle, ϕ_z)과 틸트각(tilt angle, ϕ_x)을 움직이는 장치에 장착한다. 그러면 롤각(roll angle, ϕ_y)는 영으로 간주할 수 있으며 틸트각 ϕ_x 는 팬틸트 장치의 인코더에서 측정할 수 있다. 또한 마크원점이 카메라 원점과 같은 높이에 있도록 설정하면 수직거리 D_z 는 $D_x, D_y, \phi_x, \phi_y, \phi_z$ 의 함수로 표시될 수 있다.

$$D_x(C\phi_x S\phi_y C\phi_z + S\phi_x S\phi_z) + D_y(C\phi_x S\phi_y S\phi_z - S\phi_x C\phi_z) + D_z C\phi_x C\phi_y = 0 \quad (11)$$

결과적으로 방정식 (10)에서 미지수는 D_x, D_y, ϕ_z 세개가 되므로 마크평면상의 세개의 수직선과 이에 상응하는 영상평면상의 세 교점의 값을 사용하면 미지수 D_x, D_y, ϕ_z 의 해를 구할수 있다. 그런데 다행히 서두에서 언급한 바와 같이 이 세 정보를 완벽한 폐형식의 해(exact closed form solution)로 구할 수 있음으로 해서 이동로봇가 주행하는 도중 실시간 함해가 가능하다는 것이 이 알고리즘의 장점이다. 수직선의 X좌표와 영상면상의 교점 세쌍(X_{m1}, X_{p1}), (X_{m2}, X_{p2}), (X_{m3}, X_{p3})을 각각 방정식(10)에 대입하여 얻은 세개의 방정식과 방정식 (11)을 연립하여 풀면 상당량의 계산을 거쳐서 다음과 같은 최종 결과를 얻을 수 있다. 여기서 계산을 줄이기 위하여 X_{m1} 을 영으로 둔다. 즉, 세 수직선중 하나를 마크평면의 Z_m 축과 일치시킨다.

$$\phi_z = \text{atan} \left[\frac{q_2 S\phi_x S\phi_y - q_1 f C\phi_x}{q_2 C\phi_x + q_1 f S\phi_x S\phi_y} \right] \quad (12)$$

$$D_x = \frac{A}{q_2 C\phi_x C\phi_y} [q_3 f B C + B^2 - 1] \quad (13)$$

$$D_y = \frac{A}{q_2 C\phi_x C\phi_y} [B C + q_3 f (C^2 - 1)] \quad (14)$$

$$D_z = \frac{A}{q_2} [B + q_3 f C] \quad (15)$$

여기서

$$A = S\phi_x S\phi_y S\phi_z + C\phi_x C\phi_z$$

$$B = C\phi_x S\phi_y C\phi_z + S\phi_x S\phi_z$$

$$C = C\phi_x S\phi_y S\phi_z - S\phi_x C\phi_z$$

$$q_1 = \{ X_{m2} (X_{p3} - X_{p1}) + X_{m3} (X_{p1} - X_{p2}) \} / \text{Det}$$

$$q_2 = \{ X_{m2} X_{p2} (X_{p1} - X_{p3}) + X_{m3} X_{p3} (X_{p2} - X_{p1}) \} / \text{Det}$$

$$q_3 = X_{m2} X_{p2} (X_{p2} - X_{p3}) / \text{Det}$$

$$\text{Det} = X_{m2} X_{m3} X_{p1} (X_{p2} - X_{p3}) \text{ 이다.}$$

이상의 결과를 실제 이동로봇에 응용할때 보다 단순화시킬 수도 있다. 즉, 마크와 카메라를 동일한 높이에 위치시키고 마크를 수평으로 응시함으로써 틸트각 ϕ_x 와 롤각

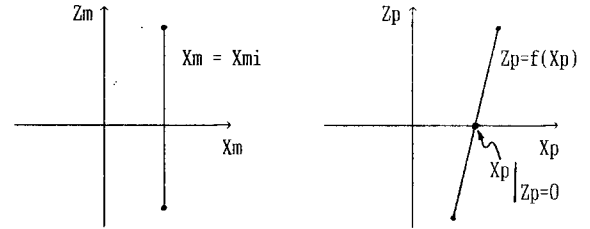
ϕ_y 를 영으로 둘 수있고 따라서 D_z 는 영이된다. 결국

$$\phi_z = \text{atan} \left[f \frac{X_{m2} (X_{p3} - X_{p1}) + X_{m3} (X_{p1} - X_{p2})}{X_{m2} X_{p2} (X_{p1} - X_{p3}) + X_{m3} X_{p3} (X_{p2} - X_{p1})} \right] \quad (16)$$

$$D_y = f \frac{X_{m2} X_{m3} (X_{p3} - X_{p2})}{X_{m2} (X_{p3} - X_{p1}) + X_{m3} (X_{p1} - X_{p2})} \text{Cos} \phi_z \quad (17)$$

$$D_x = \frac{X_{p1}}{f} D_y \quad (18)$$

로 단순화 된다. 따라서 마크영상 중 중심행 한 줄에서 세개의 경계점 (X_{p1}, X_{p2}, X_{p3})를 찾아서 위 식에 대입하면 쉽게 ϕ_z, D_x, D_y 를 구해 낼 수있다.



(a) 마크평면상의 수직선 (b) 영상평면상의 사선

그림 3. 마크평면상의 수직선과 영상평면상의 교점과의 대응

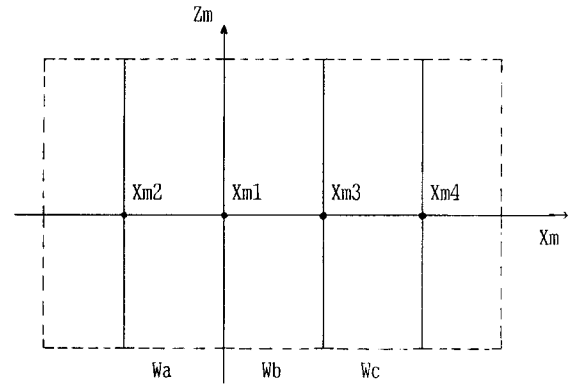


그림 4. 제안된 마크의 형태

C. 마크의 식별

이동로봇가 정해진 위치에 고정되어 있는 랜드마크로부터 거리 및 방향 정보를 얻는것도 중요하지만 임의의 대상 물체가 카메라 영상에 투시될때 이것이 마크인지 아닌지를 식별하는 것도 이에 못지 않게 필수적이다. 본 연구에서는 그림4에서 보는 바와같이 마크 평면위에 위치정보를 얻기위한 세개의 수직선 외에 한개의 수직선을 추가하여 그 네직선의 교차비(cross ratio)활용함으로써[12] 마크의 식별을 가능하게 하였다.

$$CR = \frac{(X_{m3} - X_{m2})(X_{m4} - X_{m1})}{(X_{m1} - X_{m2})(X_{m4} - X_{m3})} \quad (19)$$

따라서 별도의 마크식별 장치가 필요없이 위치 측정에 사용된 영상을 활용 할 수 있어 마크의 단순화와 계산시간의 단축을 도모할 수가 있다.

IV. 오차 해석

전장에서 제시한 알고리즘을 사용하여 실제로 거리 및 방향각을 계산함에 있어서 필연적으로 수반되는 오차로서는 카메라센서의 해상도에 의한 오차와 바닥의 불균일성으로 인한 카메라 높이 편차에 의한 오차, 카메라 미소 틸트각에 의한 오차등이 생길 수 있다. 본장에서는 이와 같은 요인으로 인해 발생하는 오차의 크기를 예측해보고 이 결과를 토대로 전장에 제시한 알고리즘이 최적으로 적용될 수 있는 설치조건을 유도한다.

A. 센서 해상도 오차

디지털카메라에 맺힌 영상은 결국 픽셀 단위의 이산 정보를 제공해 주며 결국 연속된 대상물체의 모습이 잘게 나누어짐으로 인하여 경계선의 수치에 오차를 남기게 된다. 이 오차는 여러가지 오차 원인중 가장 으뜸 (dominant)이 되며 결국 고해상도의 CCD센서와 기억 소자가 개발되어 실용화 되어야만 해결될 수 있다. 그러면 먼저 영상 정보가 pixel단위로 표현됨으로 인하여 발생하는 오차를 정량적으로 분석해보기로 한다. 계산의 단순화를 위하여 $\phi_x = \phi_y = D_z = 0$ 인 경우를 모델로 삼는다.

그림 5에서 보는 바와 같이 마크평면상에서 세계의 수직선의 좌표를 X_{m1}, X_{m2}, X_{m3} 라 두고 중간에 있는 수직선의 편이상 마크좌표계의 원점에 일치시킨다. 그리고 각 수직선 간의 거리를 W 로 두자. 카메라 영상이 맺히는 CCD 센서의 pixel수를 N_{xmax} 라 두고 픽셀의 크기를 Δx 라 하며 렌즈의 초점거리를 f 라 고한다. 그리고 오차의 정량적 분석을 위하여 마크의 세 수직선이 완전히 보일 수 있는 최소한의 기준거리 D_{min} 을 정의하고 카메라로부터 마크에 이르는 거리를 D_{min} 으로 나눈 값을 배율 M_{dis} 로 정의한다. 즉,

$$D_{min} = \frac{f(2W)}{N_{xmax} \Delta x} \quad (20)$$

$$M_{dis} = \frac{D_y}{D_{min}} \quad (21)$$

단 여기서 D_{min} 은 방위각 ϕ_z 가 영이고 X 방향거리 D_x 가 영인 경우에 Y 방향의 최소거리이며 실제로 ϕ_z 가 증가함에 따라 D_{min} 의 값은 변화한다.

한편 그림 6에서 보는 바와 같이 대상물체의 한점이 영상면의 점 P 에 투사되었다고 할때 연속적인 영상좌표는 X_p 이지만 실제로는 N_p 번째 cell이 그 값을 대신하게 된다. 즉, 다음구간의 X_p 는 모두 N_p 라는 픽셀 값으로 대치된다.

$$(N_p - N_r) \Delta x - \frac{\Delta x}{2} < X_p < (N_p - N_r) \Delta x + \frac{\Delta x}{2} \quad (22)$$

여기서 $N_r = (N_{xmax} - 1) / 2$

이다. 따라서 $X_p = (N_p - N_r) \Delta x$ 로 두고 전장에서 유도한 식 (16) (17) (18)을 구현하면 아래식 (23) (24) (25)와 같이 되는데 이 과정에서 X_p 는 최대 $\pm \Delta x / 2$ 만큼의 오차를 포함하게 된다.

$$\bar{\phi}_z = \text{atan} \left[\frac{f}{\Delta x} \frac{X_{m2}(N_{p1} - N_{p3}) + X_{m3}(N_{p2} - N_{p1})}{X_{m2}(N_{p2} - N_r)(N_{p1} - N_{p3}) + X_{m3}(N_{p3} - N_r)(N_{p2} - N_{p1})} \right] \quad (23)$$

$$\bar{D}_y = \frac{f}{\Delta x} \frac{X_{m2} X_{m3} (N_{p3} - N_{p2})}{X_{m2}(N_{p2} - N_r)(N_{p1} - N_{p3}) + X_{m3}(N_{p3} - N_r)(N_{p2} - N_{p1})} C_{\phi z} \quad (24)$$

$$\bar{D}_x = \frac{\Delta x}{f} (N_{p1} - N_r) D_y \quad (25)$$

따라서 실제 오차가 포함된 $\bar{\phi}_z, \bar{D}_x, \bar{D}_y$ 는 식 (16) (17) (18)에 $N_{pi} \rightarrow N_{pi} \pm \frac{1}{2}$ 를 대입 함으로써 얻을 수 있으며, 오차가 가장 큰 경우는 $N_{p1} \rightarrow N_{p1} + \frac{1}{2}$, $N_{p2} \rightarrow N_{p2} - \frac{1}{2}$, $N_{p3} \rightarrow N_{p3} + \frac{1}{2}$ 을 대입했을때임을 묵시적으로 알 수 있으며

계산결과로 입증된다. 절대 오차와 퍼센트 오차는 다음식으로 표현된다

$$|e|_{\phi z} = |\phi_z - \bar{\phi}_z|, |e|_{\phi z} = 100 |1 - \bar{\phi}_z / \phi_z| \quad (26)$$

$$|e|_{D_y} = |D_y - \bar{D}_y|, |e|_{D_y} = 100 |1 - \bar{D}_y / D_y| \quad (27)$$

$$|e|_{D_x} = |D_x - \bar{D}_x|, |e|_{D_x} = 100 |1 - \bar{D}_x / D_x| \quad (28)$$

그림 7에서는 거리배율 M_{dis} 가 1과 2인 경우에 대하여 방위각 ϕ_z 의 변화에 따른 각각의 절대오차를 도시하였다. 그 결과 $\phi_z = 0$ 인 근처에서 ϕ_z 의 오차가 가장 작고, 각도가 증가하면 ϕ_z 의 오차는 완만하게 증가함을 알 수 있고, D_x 및 D_y 의 오차는 다소 증가함을 알 수 있다.

B. 센서 틸팅 오차

본 알고리즘의 전개 과정에서 카메라의 틸트각 ϕ_x 는 카메라 팬틸트 장치를 능동적으로 조정하면서 그 틸트각을 알고 있다고 생각하였다. 즉, 이동로보트가 주행하는 floor가 평면이라고 가정하고 모든 이론을 전개하였다. 그러나 실제 이동로보트가 주행할 경우 floor에는 미소한 굴곡이 있을 수 있으며 이것을 경사측정계(inclinometer)로 별도로 계측하여 보상하지 않으면 오차가 발생하게 된다. 이와같이 미소틸트로 인한 오차의 특성을 알아보기 위하여 $\phi_x = \sigma$ 이고 $\phi_y = D_x = D_z = 0$ 인 경우에 대하여 방위각 ϕ_z 와 거리 D_y 의 오차를 계산하였다. 방정식 (12) (13) (14)로부터 이 경우 ϕ_z, D_x, D_y 를 다음과 같이 구할 수 있다.

$$\bar{\phi}_z = \text{atan} \left[-\frac{q_1}{q_2} f \right] \quad (29)$$

$$\bar{D}_y = \frac{C_{\phi z}}{q_2} [-q_3 f S_{\phi}^2 S_{\phi z} C_{\phi z} + S_{\phi}^2 S_{\phi z}^2 - 1] \quad (30)$$

$$\bar{D}_x = \frac{C_{\phi z}}{q_2} [S_{\phi}^2 S_{\phi z} C_{\phi z} + q_3 f (S_{\phi}^2 C_{\phi z} - 1)] \quad (31)$$

위 식 (29)에서 보는 바와 같이 방위각 ϕ_z 는 tilt angle ϕ_x 에 전혀 영향을 받지 않음을 알 수 있다.

C. 높이 편차에 의한 오차

주행 평면의 높낮이가 다소변화할 경우 구하고자하는 ϕ_z, D_y, D_x 는 전혀 영향을 받지 않는다. 본 알고리즘은 마크상에 수직선을 배열하여 그 영상으로부터 위치정보를 얻도록 고안되어 있어서 카메라의 수직방향 변위가 발생하더라도 이것에 대하여 무관하게 결과를 얻을 수 있는 장점을 가지고 있다.

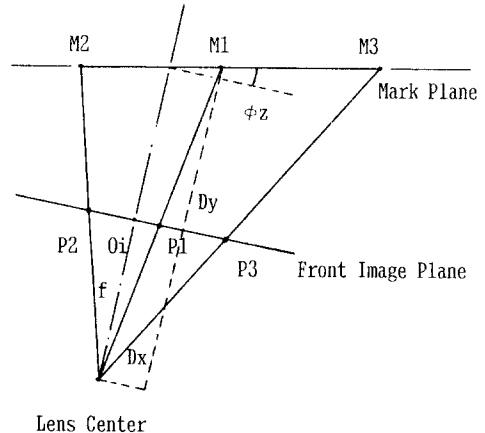


그림 5. 마크와 이미지 센서의 기하학적 관계

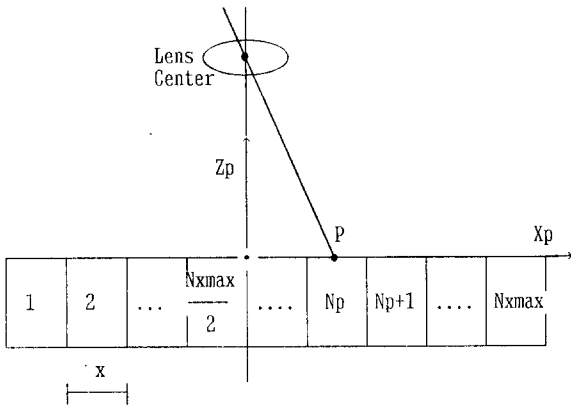


그림 6. 영상 소자의 불확실성

V. 실험 및 고찰

본 연구에서 제시한 위치측정 알고리즘과 그 오차해석을 실험적으로 확인하기 위하여 비교적 간단한 경우에 대하여 실험을 수행하였다. 알고리즘의 주요 특성을 알아보기 위하여 카메라와 마크를 동일한 높이에 위치시키고 카메라의 위치와 오리엔테이션을 변화시키면서 거리 Dx , Dy 와 방위각 ϕz 를 계산하여 그 오차를 산출하였다.

A. 실험 장치 구성

사진에서 보는 바와 같이 마크의 위치와 방위각을 변화시키기 위하여 카메라 렌즈 중심축과 나란하게 길이 1.5m의 이송레일을 설치하고 그레일 위에 X축 변위 Dx 와 방위각 ϕz 를 움직일 수 있는 X- θ 테이블을 설치하고 그 위에 마크를 고정시켰다. 이송레일은 서보모터와 컨트롤러가 부착되어 있으며 IBM-PC에 의해 제어되므로 반복수행이 수월하게 되어있고 X- θ 테이블은 수동으로 움직이게 설치되어 있다. 마크의 맞은편에 설치되어 있는 카메라는 마크가 설치된 회전 테이블의 중심을 향하도록 고정되어 있으며 렌즈 초점은 수동으로 조작된다. 이 카메라는 SONY사의 XC-77 모델의 CCD Video 모듈이며 CCD 유효화소수는 768(H) X 493(V)이며 실제 영상처리 가능한 해상도는 756(H) X 485(V)이다. 렌즈는 기본 초점거리 16 mm이며 렌즈 왜곡(distortion)이 다소있다. image processing board는 Image Technology사의 ITEX100을 사용하였는데 IBM-PC에 호환성이 있으며 grey level은 8비트 (256레벨)이고 해상도는 512 X 512이다. 영상처리를 위한 주 컴퓨터는 IBM-AT (CPU 80286 과 80287, 클럭12MHz)를 사용하였고 영상표시용 모니터와 프로그램용 모니터가 연결되어 있다. 렌즈마크는 줄무늬 형태의 검은띠를 그리고 검은색에서 흰색으로 변하는 경계선을 위치 산출을 위한 수직선으로 생각하였다. 수직선간의 간격은 60 mm 로 하였으며 두번째 수직선을 마크 좌표계의 Z축과 일치 시켰다.

B. 실험 방법 및 절차

렌즈의 중심축과 이송대 및 X- θ 테이블의 좌표계를 정확하게 일치시킨 후 calibration을 수행하여 거리 및 각도의 기준을 재조정한다. 마크의 거리 Dy 를 300 mm부터 100 mm 간격으로 증가시키고, 매 구간마다 Dx 를 0 에서 60 mm 까지 20 mm 간격으로, ϕz 를 0° 에서 80° 까지 10° 간격으로 변화시키면서 위치 산출 알고리즘을 거쳐 Dy , Dz , ϕz 를 계산하였다. 영상처리 및 위치계산을 위한 프로그램은 하드웨어 초기화 및 영상채취(SNAP), 경계선 검출, 마크식별, 위치정보계산의 순서로 구성되어 있으며 C-language를 사용하였다. SNAP명령에 의해 CCD cell에 잡힌 영상정보가 image processing board 에 기록되고 BRHLINE 명령에 의해 한행의 영상 정보가 image processing board에서 프로그램 수행 영역으로 입수된다.

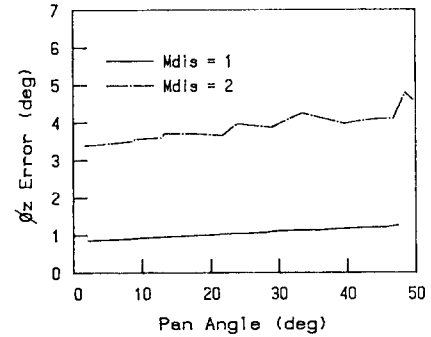
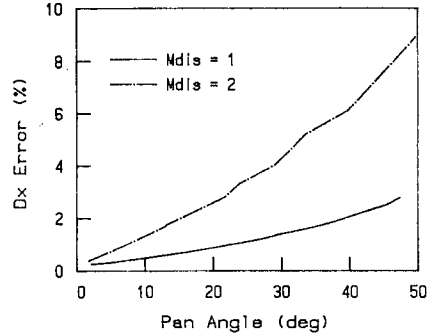
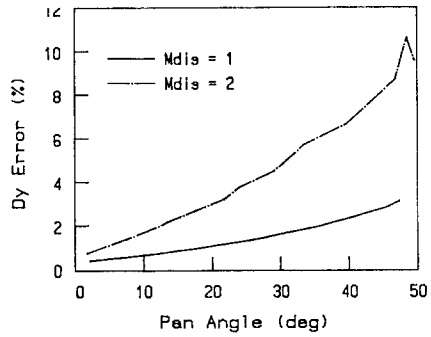
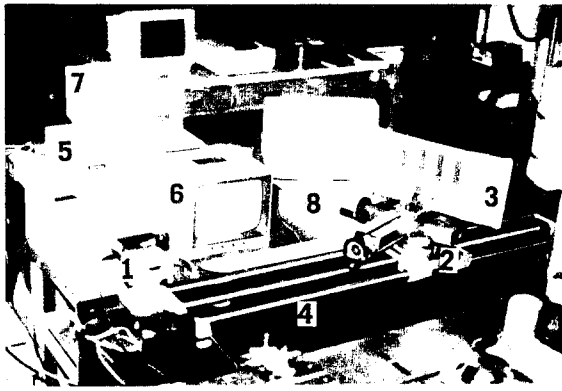


그림 7. 센서 해상도에 의한 오차

경계선 검출은 흑색에서 백색으로 변하는 경계점을 찾기 위한 것으로 gradient 방법을 사용하였다. 마크의 배경에서 나타난 불필요한 경계선들을 제거하기 위하여 첫번째 부터 네번째 경계선의 교차비를 구하여 마크의 교차비와 다르면 첫번째 경계선 부터 하나씩 버리면서 실제 마크의 경계선만 구별해 내었다. 이와 같은 방법으로 마크입이 확인되면 수평형 영상 정보를 다섯줄 뽑아서 각각 경계점을 찾은 뒤 평균하여 실제 위치 정보 ϕz , Dx , Dy 를 계산하였다.

C. 실험 결과 및 고찰

그림 8에서 보는 바와 같이 카메라와 마크와의 거리 Dy 가 증가할수록 ϕz , Dx , Dy 의 오차는 증가함을 알수있다. 이것은 마크가 카메라에 맺히는 영상의 크기와 직접관계되며 거리가 멀수록 마크영상은 작아지고 따라서 한 pixel의 오차로 인한 영향이 크게 작용한다 그러나 이 오차는 zoom렌즈를 장착한 카메라를 사용하여 마크 영상을 항상 크게 함으로써 해결될 수 있고 초점거리가 고정된 카메라를 사용할 경우 주로 특정하려는 거리의 범위를 미리 정하고 이에 맞게 렌즈의 초점거리와 마크의 크기를 결정함으로써 극복될 수 있다. 또한 방위각 ϕz 가 클수록 ϕz 의 퍼센트 오차는

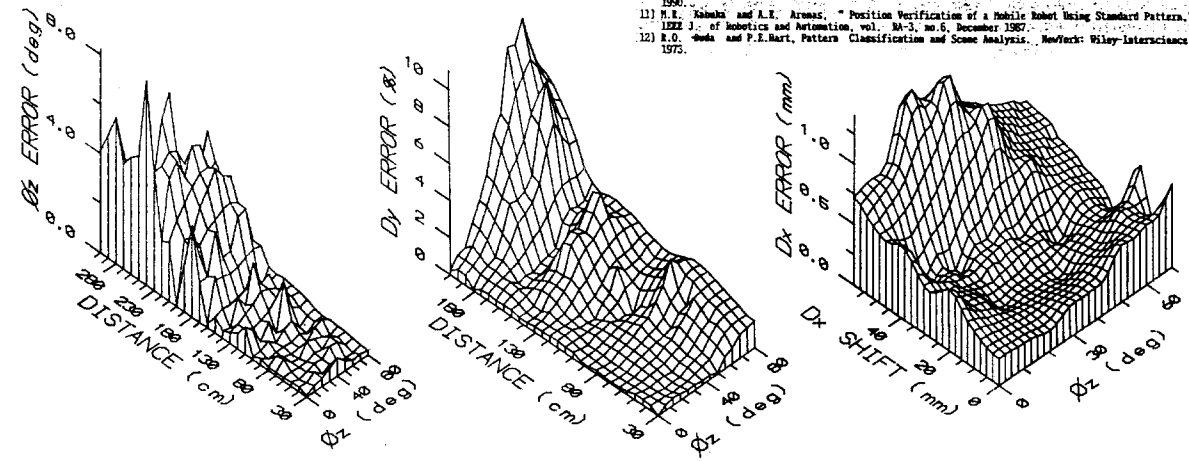


1. CCD Camera (SONY XC-77)
2. X- θ Table
3. Land Mark
4. Linear Rail
5. IBM-PC (image processing)
6. Video Monitor
7. Monitor for programming
8. Linear Rail Controller

사진 1. 실험 장치의 구성

줄어듦을 알 수 있다. 이것도 주로 pixel resolution에 기인하는 것이며 전장의 오차해석에서 분석한 결과와 같은 정도를 보여준다. ϕz 가 70° 이상의 큰 경우에서는 이론적으로 오차가 매우 적어야 하나 영상에 맺히는 마크의 크기가 매우 작아지고 따라서 줄무늬의 폭이 매우 작아져서 조명의 영향을 받아 경계선 검출이 어려워진다. 거리 Dy의 오차는 ϕz 가 40° 이상이 되면 많이 증가한다. 이것은 본 알고리즘의 특성에 기인한다.

그림 8 (c)는 거리 Dy와 방위각 ϕz 에 따른 횡방향 거리 Dx의 오차를 나타낸 것이다. Dy, ϕz 의 변화에 대해 민감하여 그 오차의 크기 또한 작은 편이나 Dx가 커질수록 Dx의 오차는 매우 커짐을 그림에서 볼 수 있다. 이것은 이변실험에서 렌즈왜곡을 보상해 주더라도 마크의 영상은 카메라 중심에서 잡는 것이 가장 정확한 위치정보를 제공할 수 있음을 말해준다. 이 실험의 결과로 볼때 카메라가 약 0°-40° 범위의 pan angle을 가지고 마크를 응시하는 것이 Dy 및 Dx의 정보가 적은 오차로 계산될 수 있으며 가능한한 마크의 중심을 렌즈 중심선 가까이 놓고 영상을 잡는 것이 좋음을 알 수 있다.



(a) ϕz 오차 (Dx=0)

(b) Dy 오차 (Dx=0)

(c) Dx 오차 (Dy=500mm)

그림 8. 위치 정보의 실험 오차

이 실험의 수행은 IBM-AT(80286 과 80287)와 image Technology사의 ITEX 100 image processing system을 사용하여 수행하였는데 카메라의 영상을 전달받아 Image board에 저장 (SNAP)하는데 66 msec, 이 이미지 영상으로부터 한줄을 가져다가 경계점을 추출하고 마크를 식별하는데 6 msec, 추출된 경계점을 가지고 위치 정보를 산출하는데 1 msec가 소요되었다. 따라서 이동로봇가 항해중 이 결과를 자신의 항해 방향 및 위치 수정에 실시간으로 활용하는데 무리가 없다.

VI. 결론

본 연구에서는 줄무늬 형태의 마크를 카메라를 통하여 얻은 영상으로부터 마크와 카메라간의 거리벡터와 방위각을 실시간으로 산출하는 알고리즘을 제시하였는데 본 연구를 통하여 얻어진 결과를 정리하면 다음과 같다.

첫째, 이 알고리즘의 전개에 있어서 최종적으로 얻는 위치정보 즉 거리벡터 (Dy, Dx)와 pan angle ϕz 는 폐형식의 해 (closed form solution)로 유도해 낼 수 있어서 매우 빠른 시간내에 결과를 얻을 수 있으며 이로 인하여 이동로봇의 항해에 실시간으로 적용 가능하다는 장점을 가지고 있다.

둘째, sensor의 해상도에 기인한 오차의 경우 방위각 ϕz 의 오차가 비교적 적기 때문에 이동로봇의 항해에서 매우 중요한 heading angle의 결정에 매우 유용하다. 뿐만 아니라 주행평면의 불균일로 인한 틸트각 (ϕx) 때문에 생기는 오차의 경우에도 방위각 ϕz 는 전혀 영향을 받지 않으며, camera의 높이 방향 편차가 발생하여도 위치정보 모두 영향을 받지 않기 때문에 이동로봇의 위치 측정 알고리즘으로 적합하다.

셋째, 마크의 식별을 위하여 이미 제작된 마크의 수직선 네개에 대한 교차비 (cross ratio)를 가지고 마크를 식별함으로써 별도의 마크식별을 위한 장치가 필요없으며 마크배정의 원하지 않는 수직선의 경우도 교차비를 이용하여 제거할 수 있어 경제적이다.

참고 문헌

- 1) N.H.E. Lavromb, "Tracking Stability of Wire Guided Vehicles," in Proc. Int. Conf. Auto. Guided Veh. Syst., pp.137-144, June 1981.
- 2) E.S. McVey, K.C. Drake, and R.H. Inigo, "Range Measurements by a Mobile Robot using a Navigation Line," IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, pp. 105-109, Jan. 1986.
- 3) T. Tomura, "A System for Measuring Current Position and/or Heading of Vehicle," Bull. JSME, vol. 25, no.203, 1982.
- 4) T. Arai, "Development of the Measuring Equipment for Location and Direction Using Ultrasonic Waves and its Evaluation," Trans. SICE, vol.18, no.10, pp. 1021-1020, Oct. 1982.
- 5) K. Fujisawa, "Guideless Robot Vehicle," Material on Vehicle Automation, no.18, pp.10-17, Dec. 1981.
- 6) M.A. Fischler and R.C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," Comm. ACM, vol.24, no.6, pp. 381-395, June 1981.
- 7) H. Weller and G. C. Tewinkel, "Space Resection in Photogrammetry," U.S. Coast and Geodetic Survey, ESSA Tech. Rep. 32, 1956. F.R. Wolf, Elements of Photogrammetry. New York: McGraw-Hill, 1974.
- 8) S. Ganapathy, "Decomposition of Transformation Matrices for Robot Vision," Proc. Int. Conf. Robotics, pp. 137-139, 1984.
- 9) R.K. Beaz and R.Y. Tsai, "Calibrating a Cartesian Robot with eye-hand Configuration Independent of Eye-to-hand Relationship," in Proc. Computer Vision and Pattern Recognition, Ann Arbor, MI, pp. 67-75, June 5-9, 1988.
- 10) Y. Liu and T.S. Saeng, "Determination of Camera Location from 2-D to 3-D Line and Point Correspondences," IEEE Trans. Pattern Analysis Mach. Intell., vol. PAMI-12, no.1, pp.28-37, Jan. 1990.
- 11) H.K. Kameki and A.E. Aronson, "Position Verification of a Mobile Robot Using Standard Patterns," IEEE J. of Robotics and Automation, vol. RA-3, no.6, December 1987.
- 12) R.O. Suda and F.E. Hart, Pattern Classification and Scene Analysis. NewYork: Wiley-Interscience, 1975.