

# 원 궤적 경로 기법을 이용한 이동로봇의 주행

## Mobile Robot Navigation Using Circular Path Planning Algorithm

한성민\*, 이강웅  
(Sung-Min Han and Kang Woong Lee)

**Abstract** : In this paper, we proposed a navigation algorithm of the mobile robot for obstacle avoidance using a circular path planning method. The proposed method makes circular paths in order to avoid obstacles in the front side of the mobile robot. An optimal path for approaching to the target is selected and the linear and angular speeds for stable moving of the mobile robot are controlled. Obstacles are detected by image processing which reduce image data obtained from a web camera. Performance of the proposed algorithm is shown by experiments with application to the Pioneer-2DX mobile robot.

**Keywords** : circular path planning, mobile robot navigation, obstacle avoidance, vision system

### I. 서론

미지의 환경에서 이동로봇이 장애물과 충돌하지 않으면서 원하는 목적지까지 이동하는 자율 주행 기법에 대한 많은 연구가 진행되고 있다. 이동로봇이 다양한 분야에서 활용됨에 따라 자율 주행 기법의 개발은 더욱 활발해지고 있다. 최근에는 장애물을 인지하기 위한 효과적인 센서로 카메라가 활용되고 있으며 염가의 카메라가 개발됨에 따라 실용성이 크게 증가되면서 카메라를 사용한 자율 주행 기법이 많이 개발되고 있다. 자율 주행 기법에서 해결해야 할 문제는 일반적으로 위치 측정(localization), 경로 계획(path planning), 모션 제어(motion control)로 나누어진다[1]. 그 중에서 주행 방향에 있는 장애물을 인지하고 최적의 경로를 선정하여 최단시간에 목적지에 도달하도록 하는 경로 계획 알고리즘의 개발이 중요한 요소이다.

지금까지 많은 자율 주행 알고리즘이 개발되었는데, APF (Artificial Potential Field) 주행기법은 이동로봇을 목적지로 이끄는 힘과 장애물로부터 밀어내는 힘을 적용하여 APF 지도를 만들고 이를 이용하여 이동경로를 생성하였다[2,3]. APF 주행 기법은 장애물이 많은 환경에서 목적지로 이동하는 최단 경로를 생성하지만 정면과 측면이 모두 막혀있는 U자 형태의 지형에서는 이동경로를 생성할 수 없다. Limit-cycle 주행기법은 장애물을 회피하는 매끄러운 경로를 생성할 수 있지만 여러 개의 장애물이 인접하여 있는 경우 Limit-cycle이 중첩되어 안정성이 제한될 수 있다[4,5]. 또한 ND (Nearness Diagram) 주행기법은 장애물이 많은 복잡한 환경에서 장애물을 회피할 수 있지만 장애물이 적은 단순한 환경에서는 효율적이지 못하다[6].

카메라에 의해 이동로봇 전방의 환경을 인지하는 경우 영상처리용 하드웨어 없이 염가의 카메라만을 부착하여 실시간 처리가 가능하도록 하면 구조가 간단해지고 비용이 절감되어 실용적이 될 수 있다. 따라서 영상처리 데이터를 줄이

는 기법이 요구된다. 영상처리 데이터를 축소시키면 영상처리 속도가 증가하여 실시간 획득이 가능할 뿐만 아니라 잡음이나 밝기 변화에 의한 영향도 개선시킬 수 있다. 한편 카메라에 의해 인지된 장애물을 회피하기 위한 경로의 생성은 목적지까지 최단거리가 되도록 하는 것이 우선되어야 하나 경로를 따라 이동하는 이동로봇의 주행안정성도 고려하여야 하며 신속하게 이동할 수 있는 경로이어야 한다.

본 논문에서는 영상처리를 신속하게 하기 위하여 영상정보를 축소시키는 기법을 적용하고 장애물을 회피하기 위한 원 궤적 경로를 제안하고자 한다. 또한 이동로봇이 생성된 원 궤적 경로를 따라 이동할 때 주행안정성을 보장하도록 속도를 제어한다. 원 궤적 경로는 장애물의 위치에 따라 원의 반경 또는 중심을 정하기만 하면 되므로 경로설정 과정이 간단하다. 또한 원의 반경을 적절히 선정하면 회피경로를 따라 이동하는 이동로봇의 안정성도 확보될 수 있을 뿐만 아니라 좁은 공간에서도 효과적으로 장애물을 회피할 수 있다.

본 논문의 II 장에서는 주변 환경 정보를 얻기 위한 카메라구조와 영상처리 방법을 설명하고 처리된 영상으로부터 장애물의 위치를 계산하는 방법을 설명한다. III 장에서는 장애물을 회피하기 위한 최적의 원 궤적 경로 설정 방법을 설명하고 IV 장에서는 원 궤적 경로를 따라 이동하기 위한 이동로봇의 속도 제어 방법에 대해 설명한다. V 장에서는 실험 결과를 통하여 제안하는 알고리즘의 성능을 입증하고 결론을 VI 장에서 내린다.

### II. 영상처리 시스템

장애물 정보를 얻기 위한 영상은 그림 1에서 보는 것과 같이 이동로봇 상단에 설치된 단일 카메라로부터 얻어진다. 카메라를 지면 방향으로 설치하여 이동로봇의 전면과 측면에 존재하는 장애물을 확인하는데 이 때 얻어지는 영상은 지면 가까이에 설치된 카메라로부터 얻는 영상보다 사각이 적고 정확한 장애물 정보를 가지게 된다. 그림 2는 그림 1과 같이 설치된 카메라로부터 얻어진 영상이다.

카메라로부터 얻어진 320x240 크기의 컬러 영상은 처리 시간을 단축하기 위해 흑백 영상으로 처리한다. 흑백영상은

\* 책임저자(Corresponding Author)

논문접수 : 2008. 8. 4., 채택확정 : 2008. 9. 12.

한성민, 이강웅 : 한국항공대학교 항공전자 및 정보통신공학부  
(maruji@kau.ac.kr/kwlce@kau.ac.kr)

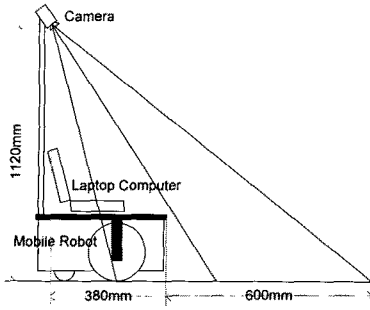


그림 1. 영상 시스템의 기하학적 구조.  
Fig. 1. Geometric structure of vision system.

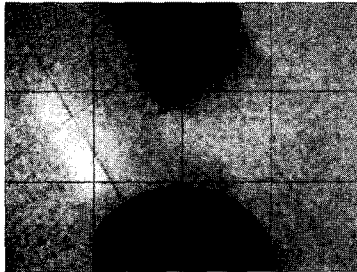


그림 2. 카메라에서 획득된 영상.  
Fig. 2. The image obtained by the camera.

인접하는 영상 픽셀을 그룹화하여 하나의 픽셀로 나타내는 영상 축소기법을 적용한다. 즉, 다음과 같이 인접하는  $n \times n$  픽셀 그룹을 하나의 픽셀로 나타낸다[7].

$$I(x, y) = \frac{1}{n^2} \sum_{k=0}^{n-1} \sum_{l=0}^{n-1} i(nx+k, ny+l) \quad (1)$$

여기서  $i(x, y)$ 는 원 영상에서 픽셀의 밝기 값,  $I(x, y)$ 은 줄어든 영상에서 픽셀의 밝기 값이며 0~255사이의 범위를 가진다.

축소된 영상은 원 영상의  $1/n^2$  크기이므로 장애물 정보 획득을 위한 영상처리속도가 매우 빠르며 평균값을 사용하여 축소하기 때문에 잡음과 빛의 변화 등에 둔감하게 된다. 지면의 색은 미리 알고 있다고 가정하고 축소된 영상으로부터 지면의 색과 대비하여 이진 영상으로 처리한다. 이진화된 영상에서 장애물의 양 끝점의 위치를 찾아 이미지 좌표계에서의 장애물 위치를 정하는데 장애물의 양 끝점만을 이용하여 이동경로를 생성하면 그림 3과 같이 장애물의 크기나 모양에 따라 충돌이 발생할 수 있다. 따라서 이동로봇과 최단

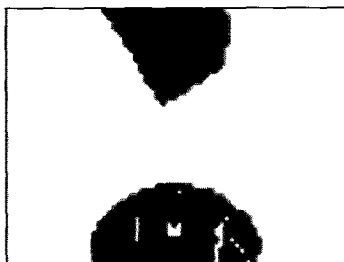


그림 3. 이진화 처리된 영상.  
Fig. 3. The image obtained by binary processing.

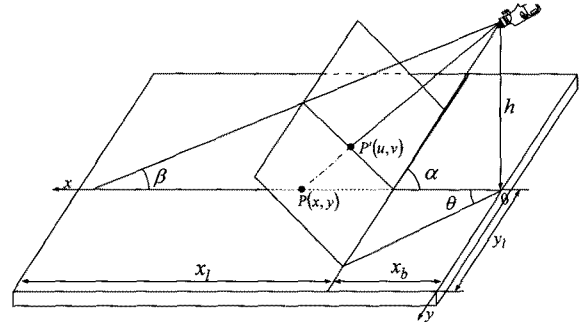


그림 4. 영상평면상의 좌표와 공간 좌표의 관계.  
Fig. 4. The relationship of the image frame and space frame.

거리에 있는 점을 찾고 최단거리의 점을 기준으로 장애물을 돌로 나눈 후 각 장애물의 양 끝점을 이용하여 경로를 생성한다.

이진화된 영상평면상의 장애물 위치를 로봇이 이동하는 공간상의 위치로 변환시켜야 한다. 한 대의 카메라를 사용하기 때문에 그림 4와 같은 영상평면상의 좌표와 3차원 공간 좌표 사이의 기하학 구조를 이용하여 실제 공간 좌표를 얻는다[8].

공간상에 있는 모든 물체는 지면에 접하고 있다고 가정하면 그림 4의 매개변수  $\alpha, \beta, \theta$ 는 각각 다음과 같이 정의된다.

$$\alpha = \tan^{-1} \left[ \frac{h}{x_b} \right] \quad (2)$$

$$\beta = \tan^{-1} \left[ \frac{h}{x_b + x_l} \right] \quad (3)$$

$$\theta = \tan^{-1} \left[ \frac{y_l}{x_b + x_l} \right] \quad (4)$$

여기서  $h$ 는 카메라의 높이,  $x_b$ 는 카메라의 앞부분에 존재하는 사각지역의 거리,  $x_l$ 과  $y_l$ 은 카메라에 보이는  $x$ 축 방향과  $y$ 축 방향의 최대 거리이다.

매개변수  $\alpha, \beta, \theta$ 를 사용하면 이동로봇의 공간 좌표상의 장애물 위치  $x, y$ 는 다음과 같이 표현된다.

$$x = h \times \tan \left[ (90 - \alpha) + \left[ \frac{u}{S_x} \right] (\alpha - \beta) \right] \quad (5)$$

$$y = x \times \tan \left[ \frac{2v}{S_y} \times \theta \right] \quad (6)$$

여기서  $S_x, S_y$ 는 영상 좌표상에서 수직방향의 총 픽셀 수와 수평방향의 총 픽셀 수이며  $u, v$ 는 영상 평면상의 중심에서 장애물까지의 수직 픽셀 수와 수평 픽셀 수이다.

### III. 장애물 회피 경로

이동로봇이 목표물까지 이동하기 위한 경로는 안정성을 유지하면서 최단시간에 도달할 수 있도록 설정되어야 한다. 또한 이동하는 경로 전방에서 장애물이 인지되어도 이동로

봇은 신속하고 안정적으로 장애물을 회피하여 목표물을 향하여야 한다. 따라서 장애물 회피 경로의 설정은 단순해야 하며 이동 안정성과 신속성이 보장되어야 한다. 이와 같은 목적에 적합한 경로로 원 궤적 경로를 제안하고자 한다. 원 궤적 경로는 장애물의 위치에 따라 원의 반경 또는 중심을 정하기만 하면 되므로 경로설정 과정이 간단하다. 또한 원의 반경을 적절히 선정하면 회피경로를 따라 이동하는 이동로봇의 안정성도 확보될 수 있을 뿐만 아니라 좁은 공간에서도 효과적으로 장애물을 회피할 수 있다.

카메라에 의해 이동 환경이 실시간으로 감시되고 인지된 장애물을 회피하기 위한 원 궤적 경로생성 알고리즘을 제시하기로 한다. 경로생성은 이동로봇의 방향과 장애물의 위치에 따른 원의 중심 또는 원의 반경을 정하는 것이다. 이동로봇은 항상 목적지를 향하여 직선적으로 이동하는 것이 최상의 경로선택이나 이동로봇이 목적지 방향과 다른 방향을 지향하고 있거나 장애물이 나타나는 경우 최상의 경로선택을 하도록 한다.

이동로봇이 목적지와 다른 방향을 향하고 있으면 목적지와 같은 방향과 일치할 때까지 원 궤적 경로를 따라 이동하도록 한 후 직선 경로를 따라 이동하도록 한다. 이 때 원 궤적 경로를 위한 원의 반경과 중심을 정해야 하는데 목적지까지의 거리가 멀고 이동로봇이 지향하는 방향과 목적지 방향과의 차가 크면 그림 5의 점선과 같이 지나치게 우회하는 원 궤적 경로가 형성된다. 이런 경우 이동로봇과 목적지 사이에 가상 목적지를 설정하면 근거리의 원 궤적이 형성되어 이동시간을 줄일 수 있다. 가상 목적지까지의 거리  $S$  는 이동로봇이 안정적으로 신속하게 회전하면서 이동할 수 있도록 선정한다.

이동로봇의 현재 위치를 원점으로 하는 평면 좌표계에서 가상 목적지의 좌표를  $p_{vg} = [x_{vg} \ y_{vg}]^T$  라 하고 원의 중심 좌표를  $r_{vg} = [0 \ y_{rvg}]^T$  라 하면 다음 식이 성립한다.

$$y_{rvg} = \frac{\|p_{vg}\|^2}{2y_{vg}} \quad (7)$$

가상 목적지에 의해 생성된 원 궤적 경로를 따라 이동하면 이동로봇의 방향과 목적지의 방향이 일치하게 되고 이동로봇은 목적지를 향해서 직선으로 이동할 수 있다.

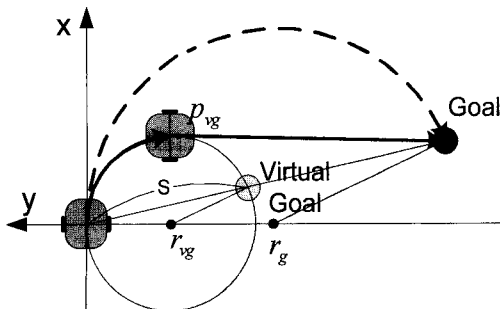


그림 5. 원 궤적 경로와 가상 목적지.  
Fig. 5. The circle trajectory and virtual goal.

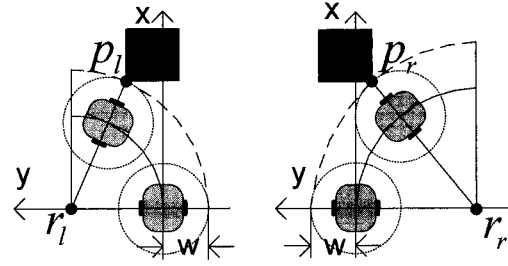


그림 6. 전방 장애물을 회피하기 위한 원 궤적.  
Fig. 6. The circle trajectory for avoidance of obstacles in the front.

한편 이동로봇 전방에 장애물이 존재하면 장애물의 위치에 따라 장애물의 전방에서 오른쪽 또는 왼쪽으로 회피하거나 장애물의 후방으로 오른쪽 또는 왼쪽으로 해야 하는데 이를 위한 원 궤적 경로 결정이 이루어져야 한다.

그림 6과 같이 이동로봇의 위치를 원점으로 하는 평면 좌표계에서 장애물의 왼쪽 끝 좌표를  $p_l = [x_{pl} \ y_{pl}]^T$  이라 하고 장애물의 오른쪽 끝 좌표를  $p_r = [x_{pr} \ y_{pr}]^T$  이라 하자. 그리고 왼쪽으로 회피하는 가장 큰 원의 중심을  $r_l = [0 \ y_{rl}]^T$  이라 하고 오른쪽으로 회피하는 가장 큰 원의 중심을  $r_r = [0 \ y_{rr}]^T$  라 하면 다음 식이 주어진다.

$$\|p_l - r_l\|^2 = |y_{rl} + W|^2 \quad (8)$$

$$\|p_r - r_r\|^2 = |y_{rr} - W|^2 \quad (9)$$

여기서  $W$  는 이동로봇의 폭이다.

식 (8)과 식 (9)로부터  $y_{rl}$  과  $y_{rr}$  은 각각 다음과 같다.

$$y_{rl} = \frac{1}{2} \frac{\|p_l\|^2 - W^2}{y_{pl} + W} \quad (10)$$

$$y_{rr} = \frac{1}{2} \frac{\|p_r\|^2 - W^2}{y_{pr} - W} \quad (11)$$

따라서 장애물의 왼쪽 전방으로 회피하기 위한 원 궤적 중심의  $y$ 좌표는 다음 범위 내에 있어야 한다.

$$y_{rl} \geq y_r \geq 0, \quad y_{rl} > 0 \quad (12)$$

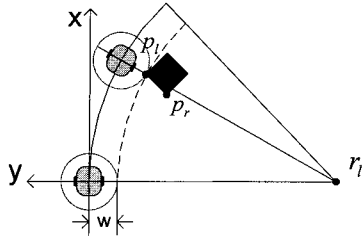
한편, 장애물의 오른쪽 전방으로 회피하기 위한 원 궤적 중심의  $y$ 좌표는 다음 범위 내에 있어야 한다.

$$0 \geq y_r \geq y_{rr}, \quad y_{rr} < 0 \quad (13)$$

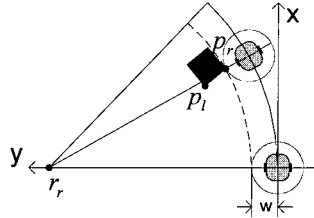
그러나 이동로봇의 우측에 있는 장애물의 크기를 인지할 수 있으면 그림 7(a)와 같이 장애물의 왼쪽 후방으로 회피할 수 있다. 이동로봇이 장애물의 왼쪽 후방으로 이동하기 위한 원 궤적 중심의  $y$ 좌표는 다음 조건을 만족시켜야 한다.

$$y_{rl} \geq y_r, \quad y_{rl} < 0 \quad (14)$$

반대로 그림 7(b)와 같이 이동로봇이 장애물의 오른쪽 후방으로 회피하기 위한 원 궤적 중심의  $y$ 좌표는 다음 조건을



(a) 장애물의 왼쪽 후방으로 회피하기 위한 원 궤적.



(b) 장애물의 오른쪽 후방으로 회피하기 위한 원 궤적.

그림 7. 장애물의 후방으로 회피하기 위한 원 궤적.

Fig. 7. The circle trajectory for avoidance of obstacles in the rear side.

만족시켜야 한다.

$$y_r \geq y_{rr}, \quad y_{rr} > 0 \quad (15)$$

이동로봇 전방에 있는 장애물이 존재하면 식 (12), (13), (14) 과 (15)의 조건을 만족시키는 장애물 회피 이동 경로군이 형성된다. 이와 같은 경로군 중에서 이동로봇이 충돌 없이 목적지로 이동하기 위해 최적의 경로를 찾아야 한다.

장애물을 고려하지 않고 목적지로 가기 위한 경로를 가상 목적지를 정하여 설정하였다면, 장애물이 존재하는 경우 장애물을 회피하기 위한 경로군에서 가상목적지를 정하여 설정한 경로와 유사한 경로가 최적의 경로가 된다. 이 경로 오차는 다음과 같이 반경의 역수 차로 나타낼 수 있다.

$$e = \left| \frac{1}{y_{rvg}} - \frac{1}{y_r} \right| \quad (16)$$

장애물을 회피하기 위한 원 궤적 경로 각각에 대해 식 (16)을 계산하여 최소가 되는 경로가 목적지로 향하는 최적의 원 궤적 경로가 된다. 그림 8과 같이 이동로봇의 전방에

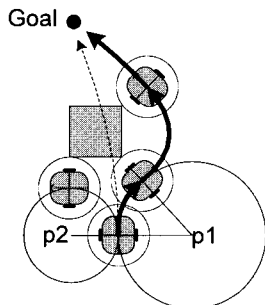


그림 8. 장애물 회피 경로의 선택.

Fig. 8. Selection of obstacle avoidance path.

장애물이 있는 경우 장애물을 회피하기 위한 원 궤적의 중심은 p1, p2가 되나 목적지에 도달하기 위한 경로와 유사한 원 궤적 경로는 p1이 되므로 장애물을 회피하면서 이동하기 위한 경로를 선택할 수 있다.

#### IV. 속도 제어

이동로봇이 그림 9와 같은 원 궤적 경로를 따라 이동할 때의 각속도와 선속도를 결정하여야 한다. 그림 9에서 보는 바와 같이 이동로봇이  $\theta$  만큼 이동하였다면 이동거리  $l$  은 다음과 같다.

$$l = r \cdot \theta \quad (17)$$

여기서  $r$  은 반경이고  $\theta$  는 회전 각도이다.

원 궤적을 선회하는 경우 일정한 선속도와 각속도를 유지하면 각각 다음 식으로 나타낼 수 있다.

$$l = v \cdot t \quad (18)$$

$$\theta = \omega \cdot t \quad (19)$$

두 식으로부터 시간 변수를 제거하면 선속도와 각속도 사이의 관계는 다음 식으로 표시된다.

$$\omega = \frac{v}{r} \quad (20)$$

이동로봇이 원 궤적을 선회할 때 발생하는 원심력은  $F = \frac{mv^2}{r}$  로 표시되므로 선속도는 다음과 같이 원의 반경 함수로 나타낼 수 있다.

$$v_f = \sqrt{\frac{F}{m} r} = \gamma \sqrt{r} \quad (21)$$

식 (21)으로부터 원의 반경에 따른 선속도가 정해지면 식 (20)을 이용하여 각속도를 정할 수 있다. 이 때 선속도는 최대속도를 넘지 않도록 제한한다.

$$v = \begin{cases} v_{\max} & , v_f \geq v_{\max} \\ v_f & , v_f < v_{\max} \end{cases} \quad (22)$$

여기서  $v_{\max}$  는 이동로봇의 최대 선속도이다.

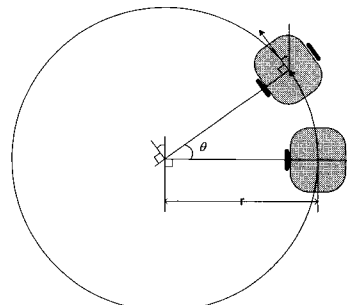


그림 9. 원 궤적 경로.

Fig. 9. Circular path.

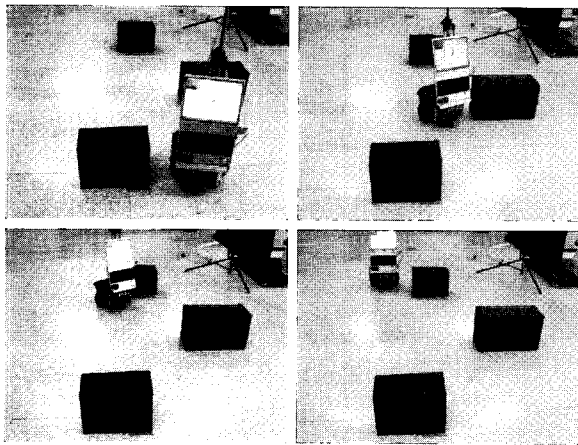
V. 실험

제안된 원 궤적 경로 설정 알고리즘을 Pinoer2-DX 이동로봇에 적용하여 전방에 있는 장애물을 회피하여 목적지로 이동하는 실험을 실시하였다. 영상 획득을 위한 카메라는 저가의 PC CAM을 사용하였으며 그림 1과 같은 구조로 설치하였다. 영상은 1/25 크기로 축소되도록 하였으며 최대 선속도는 300mm/s로 제한하였다. 실험은 7평 정도의 공간에서 진행되었으며 목적지는 출발지점에서 전방 5m지점으로 설정하였다. 영상처리 시간은 1프레임당 12ms이다. 가상 목적지의 거리  $S$ 는 주행환경을 고려하여 0.5m로 정하였으며, 속도 상수  $\gamma$ 는 20으로 정하였다. 그림 10은 넓은 공간에 3개의 장애물이 흩어져 있는 단순한 환경에서의 실험 결과로 (a)는 주행 과정을 나타낸 사진이고 (b)는 이동로봇의 주행경로, (c)는 선속

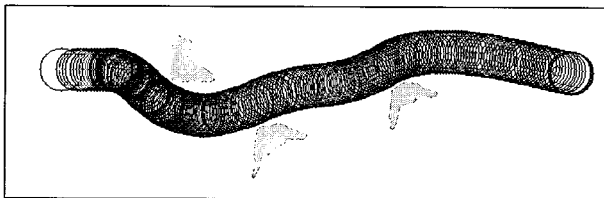
도, (d)는 각속도이다. 이동시간은 23초이며 대부분의 구간에서 200mm/s 이상의 속도를 유지하며 목적지까지 도달하였음을 확인할 수 있다.

그림 11은 장애물이 밀집되어 있는 복잡한 환경에서 실험한 결과로 (a)는 주행 과정을 나타낸 사진이고 (b)는 주행경로, (c)와 (d)는 각각 선속도와 각속도이다. 이동시간은 23초이며 첫 번째 실험과 거의 동일한 시간에 목적지까지 도달하였음을 확인할 수 있다. 이것은 이동로봇의 선속도가 장애물과의 거리와 관계없이 결정되기 때문이다.

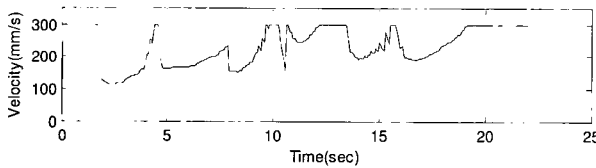
그림 11과 같은 환경에서 기존의 ND 방법을 적용하여 실험한 결과를 그림 12에 나타내었는데 장애물을 회피하여 벗어나는데 33초가 걸려 제안된 기법이 10초 정도 단축할 수 있음을 확인할 수 있다. 이것은 ND 방법이 이동경로가 복잡



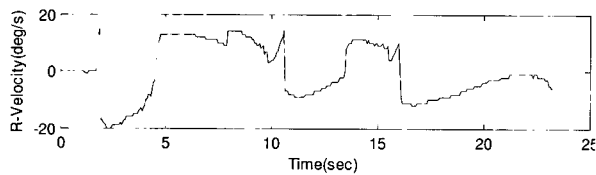
(a) 장애물 회피 주행.



(b) 이동경로.



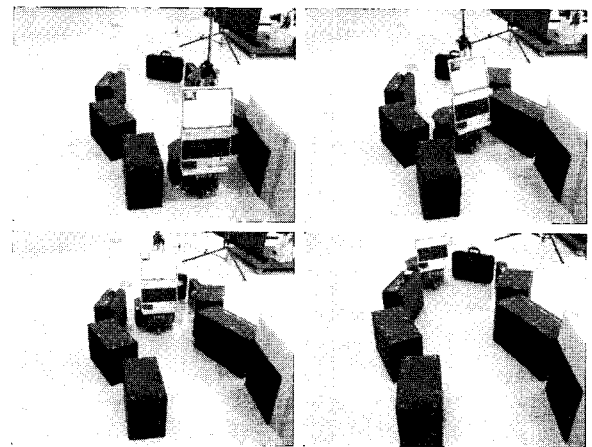
(c) 선속도.



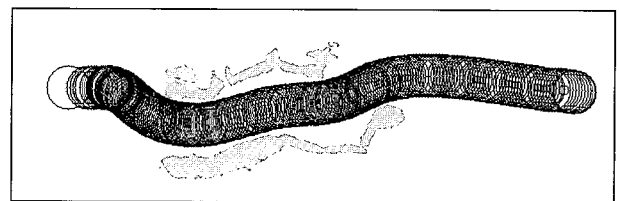
(d) 각속도.

그림 10. 단순한 장애물 회피실험 결과.

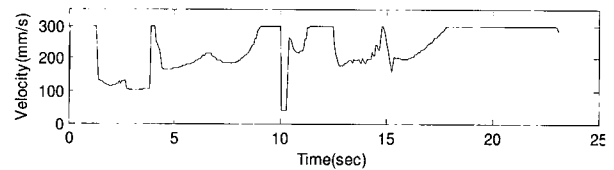
Fig. 10. Experimental result under the environment with simple obstacles.



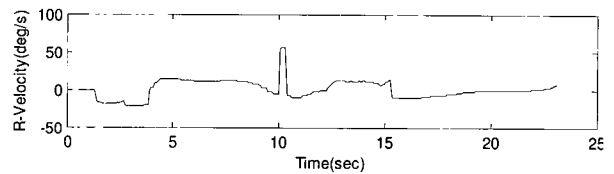
(a) 장애물 회피 주행.



(b) 이동경로.



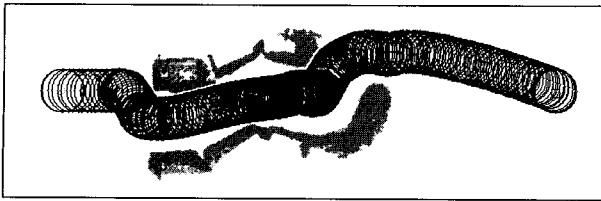
(c) 선속도.



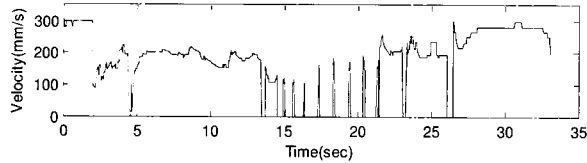
(d) 각속도.

그림 11. 복잡한 장애물 회피 실험 결과.

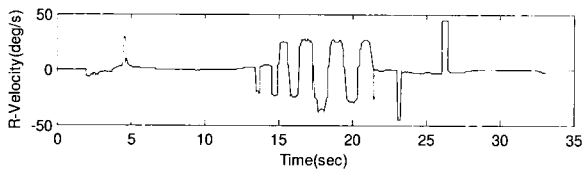
Fig. 11. Experimental result under the environment with complex obstacles.



(a) 이동경로.



(b) 선속도.



(c) 각속도.

그림 12. ND 항법을 이용한 장애물 회피 실험 결과.

Fig. 12. Experimental result using the ND navigation method.

하고 장애물의 거리에 따라 이동속도가 정해지기 때문이다.

## VI. 결론

본 논문은 단일 카메라를 사용하여 주변의 환경 정보를 얻고 인지된 장애물을 원 궤적 경로를 따라 이동하면서 목적지까지 이동하는 원 궤적 경로 장애물 회피 기법을 제안하였다. 카메라로부터 얻어진 영상은  $n \times n$  픽셀 단위로 평균값을 계산하여 영상 데이터를 축소한 후 영상처리를 통하여 장애물을 인지하도록 하였다. 인지된 장애물 군을 회피하기 위한 원 궤적 경로를 선정된 후 원 궤적 크기에 따른 속도제한을 하여 안정적으로 장애물을 회피하도록 하였다. 제안된 방



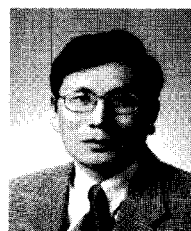
한성민

2007년 한국항공대학교 항공전자공학과 (공학사). 2007년~현재 한국항공대학교 대학원 항공전자공학과 석사과정 재학 중. 관심분야는 이동로봇 제어, 로봇 팔 제어.

법의 성능 평가는 Pioneer2-DX 이동로봇에 적용한 실험을 통하여 실시하였으며 단순하거나 복잡한 장애물 환경에서도 신속하게 장애물을 회피하여 목적지에 도달함을 확인하였다. 따라서 복잡한 환경하에서도 효과적으로 장애물을 회피하면서 신속하게 목적지에 도달하는 성능이 보장되어 실용성이 높은 것으로 기대된다.

## 참고문헌

- [1] N. Sariff and N. Buniyamin, "An Overview of Autonomous Mobile Robot Path Planning Algorithms," *SCOReD 4th Student Conf. Research and Development*, pp. 183-188, June, 2006.
- [2] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," *Proc. of IEEE Conf. Robotics and Automation Sacramento*, pp. 1398-1404, April, 1991.
- [3] S. S. Ge and Y. J. Cui, "New Potential Functions for Mobile Robot Path Planning," *IEEE Trans. Robotics and Automation*, vol. 16, no. 5, October 2000.
- [4] D.-H. Kim, J.-H. Park, and J.-H. Kim, "Limit-cycle navigation Intelligence, Robotics and Autonomous Systems, 2001.
- [5] R. Grech and S. G. Fabri, "Trajectory Tracking in the Presence of Obstacles Using the Limit Cycle Navigation Method," *Proc. of the 13th Conf. Control and Automation*, pp. 101-106, June, 2005.
- [6] J. Minguez and L. Montano, "Nearness diagram (ND) navigation-collision avoidance in troublesome scenarios," *IEEE Trans. of Robotics and Automation*, vol. 20, pp. 45-59, Feb. 2004.
- [7] S. Vitabile, G. Pilato, F. Pullara, and F. Sorbello, "A Navigation System For Vision-Guided Mobile Robots," *Proc. of International Conf. Image Analysis and Processing*, pp. 566-571, Sept. 1999.
- [8] G. Cheng and A. Zelinsky, "Real-Time Visual Behaviours for Navigating a Mobile Robot," *Proc. of IEEE/RSJ International Conf. Intelligent Robotics and Systems*, vol. 2, pp. 973-980, Nov. 1996.



이강웅

1980년 한국항공대학교 항공전자공학과 (공학사). 1982년 서울대학교 전자공학과(공학석사). 1983년~1984년 삼성전자 컴퓨터개발부. 1989년 서울대학교 전자공학과(공학박사). 1994년~1995년 미시건주립대학교 방문교수. 1989년~현재 한국항공대학교 항공전자 및 정보통신공학부 교수. 관심분야는 로봇제어, 비선형제어.