

단일곡률궤적을 이용한 이동물체의 포획 알고리즘

A Capturing Algorithm of Moving Object using Single Curvature Trajectory

최 병 석, 이 장 명*
(Byoung-Suk Choi and Jang-Myung Lee)

Abstract : An optimal capturing trajectory for a moving object is proposed in this paper based on the observation that a single-curvature path is more accurate than double- or triple-curvature paths. Moving distance, moving time, and trajectory error are major factors considered in deciding an optimal path for capturing the moving object. That is, the moving time and distance are minimized while the trajectory error is maintained as small as possible. The three major factors are compared for the single and the double curvature trajectories to show superiority of the single curvature trajectory. Based upon the single curvature trajectory, a kinematics model of a mobile robot is proposed to follow and capture the moving object, in this paper. A capturing scenario can be summarized as follows: 1. Motion of the moving object has been captured by a CCD camera., 2. Position of the moving object has been estimated using the image frames, and 3. The mobile robot tries to follow the moving object along the single curvature trajectory which matches positions and orientations of the moving object and the mobile robot at the final moment. Effectiveness of the single curvature trajectory modeling and capturing algorithm has been proved, through simulations and real experiments using a 2-DOF wheel-based mobile robot.

Keywords: single curvature trajectory, mobile robot, capturing, moving object

I. 서론

‘로봇(robot)’이라는 용어가 우리 생활에서 인식되기 시작한 것은 1970년대부터이다. 1970년대 마이크로프로세서(micro-processor)의 등장으로 인해 정확하고 편리하게 로봇을 제어할 수 있게 되었고, 80~90년대에 개발된 고성능 프로세서는 로봇의 발전 속도를 한층 빠르게 만들었다[1,2]. 집적회로, 센서, 인공지능, 영상처리, 컴퓨터 기술의 발달로 로봇은 과거보다 훨씬 더 정밀하고 신속한 제어가 가능해졌고, 오늘날에는 주위환경을 인식하여 환경변화에 따라서 스스로 동작형태를 결정하는 이른바 지능형 로봇(intelligent robot)이 등장하기에 이르렀다[3].

로봇은 형태에 따라서 크게 로봇 매니플레이터(manipulator)와 모바일(mobile) 로봇으로 나눌 수 있다. 고정된 작업공간을 갖는 로봇 매니플레이터에 비해 모바일 로봇은 이동이 가능하기 때문에 작업공간상의 한계는 없지만, 로봇 자체가 이동하므로 로봇의 위치추정, 속도 및 가속도, 장애물인식, 이동경로설정등 이동에 의해 발생하는 사항들을 고려해야 한다. 따라서 모바일 로봇에 대한 연구는 로봇 자체의 주행에 관련한 경로 계획, 위치추정, 주행제어로 크게 구분할 수 있다[4,5]. 로봇의 주행경로계획은 초기위치에서 목표지점까지 최적의 경로를 설계하는데 그 목적이 있다. 일반적으로 최적의 경로를 설계할 때 최소주행시간, 최소주행거리, 최소주행오차를 고려하게 되는데, 이 요소들은 독립적이지 않고, 서로 상관관계를 가지고 있다. 예컨대, 등속도로 주행하는 로봇의 주행거리와 주행시간은 비례관계에 있으며, 주행거리가 일정

할 때는 주행속도와 주행시간은 반비례 관계에 놓인다. 주행시간을 줄이기 위해서 주행속도를 높이면, 일반적으로 주행오차가 증가한다. 주행오차의 증가는 주행중 자세 및 위치보정 횟수의 증가로 이어져서 전체적으로는 주행시간과 주행거리가 오히려 늘어나는 현상을 보인다. 그러므로 경로계획은 그림 1에 나타난 것처럼 여러가지 고려사항을 토대로 가장 적절한 경로를 선정하는 것이 목표가 된다[6].

이렇듯 모바일 로봇의 주행에 있어서 경로계획은 기본이 되는 중요한 부분이기 때문에, 주행오차나 주행거리 및 주행시간을 줄일 수 있는 주행궤적에 대한 연구가 필요하다. 기존의 모바일 로봇에 대한 연구는 로봇에 장착된 센서와 추정기(칼만필터)를 통한 위치추정이나, 주행중의 오차를 보정하는 제어기법들(신경회로망, 적응제어, 예측제어)에 대한 내용이 대부분이었다[7].

따라서 본 논문에서는 모바일 로봇의 효율적인 주행궤적에 초점을 맞추었다. 로봇의 주행궤적으로 단일곡률궤적을 제시하고, 이동물체를 추적 및 포획하는 궤적계획에 실제 적용시켜 보았다. 시뮬레이션과 실험을 통해서 단일곡률궤적이 주행시간, 주행거리, 주행오차에 대해 어떠한 특성을 보이는지 알아보았다. 모바일 로봇에 장착된 CCD카메라를 이용하

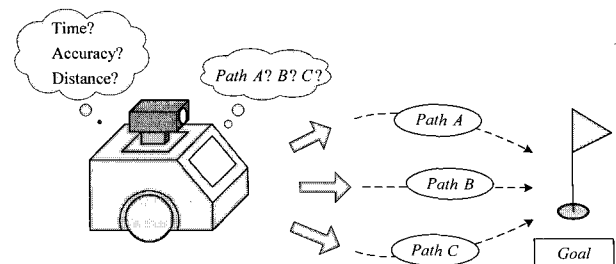


그림 1. 모바일 로봇의 주행경로계획.
Fig. 1. Trajectory planning for a mobile robot.

* 책임저자(Corresponding Author)

논문접수 : 2004. 9. 30., 채택확정 : 2005. 9. 29.

최병석, 이장명 : 부산대학교 전자공학과

(woodc@pusan.ac.kr/jmlee@pusan.ac.kr)

※ 본 연구는 정보통신부 대학 IT 연구센터(ITRC) 육성, 지원 사업의 연구결과로 수행되었음.

여 이동물체의 예상위치를 추정한 다음 단일곡률궤적을 따라서 모바일 로봇을 주행시켜 이동물체를 포획하도록 구성였다. 2장에서 모바일 로봇의 기구학 해석과 주행특성을 살펴보고, 3장에서는 단일곡률궤적과 더블곡률궤적을 소개하였다. 4장에서는 단일곡률궤적을 이용한 이동 물체의 포획 알고리즘을 소개하였다. 단일곡률궤적과 더블곡률궤적 비교실험 및 이동물체의 포획 실험 내용은 5장에 담았으며, 6장에서 본 논문의 결론을 맺었다.

II. 로봇의 기구학적 모델링 및 주행특성

모바일 로봇의 이동경로를 분석하고 궤적을 설정하기 위해서는 로봇의 제어변수와 로봇의 위치 및 속도와의 관계를 알아보는 기구학 해석이 선행되어야 한다. 기구학 해석을 바탕으로 차륜구동(differential driving) 형태를 가지는 모바일 로봇의 주행특성을 살펴본다.

1. 모바일 로봇의 기구학 해석

차륜구동형태의 모바일로봇은 그림 2(a)와 같이 2개의 바퀴가 같은 축상에 위치해 있으며, 각 바퀴는 모터에 의해서 독립적으로 제어된다. 여기서 v_L 은 왼쪽 바퀴의 속도, v_R 은 오른쪽 바퀴의 속도를 나타내며, 양 바퀴사이의 길이는 l 로 표시한다. 로봇의 움직임은 두 바퀴의 속도, v_L 및 v_R 에 의해서 결정되며, 바퀴의 속도값과 로봇의 선속도, v_1 , 및 회전각속도, v_2 ,와의 관계를 (1), (2)에서 볼 수 있다[8].

$$v_1 = \frac{v_R + v_L}{2} \tag{1}$$

$$v_2 = \frac{2(v_R - v_L)}{l} \tag{2}$$

차륜구동형태의 모바일 로봇의 기구학 모델은 그림 2(b)를 바탕으로 살펴볼 수 있다.

2차원 $X - Y$ 직교좌표계(cartesian coordinate)상에서 로봇의 위치는 $x_R(t)$ 와 $y_R(t)$ 로 표시하며, 방향은 $\theta_R(t)$ 로 나타낸다. $\dot{x}_R(t)$ 와 $\dot{y}_R(t)$ 은 로봇의 선형속도 성분이 되며, $\dot{\theta}_R(t)$ 은 회전각속도이다. 직교좌표공간의 위치 및 방향을 P 로 정의하면,

$$P = [x_R \quad y_R \quad \theta_R]^T \tag{3}$$

로 표시된다. 그러면, 직교 좌표상에서의 속도는

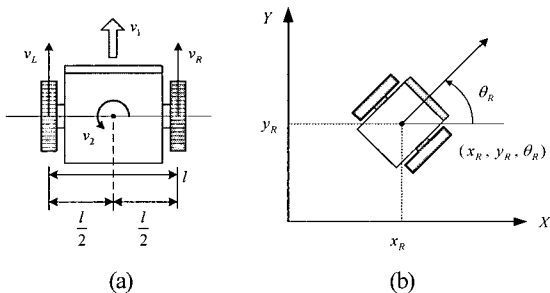


그림 2. 모바일 로봇의 기구학 모델.
Fig. 2. Kinematics model of mobile robot.

$$\dot{P} = [\dot{x}_R \quad \dot{y}_R \quad \dot{\theta}_R]^T \tag{4}$$

로 나타내어지며, $\dot{x}_R(t) = \cos\theta_R v_1$, $\dot{y}_R(t) = \sin\theta_R v_1$, 및 $\dot{\theta}_R(t) = v_2$ 로 정의된다. (3)과 (4)를 바탕으로 모바일 로봇의 기구학 모델은 (5)로 표시된다[9].

$$\begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} = \begin{bmatrix} \cos\theta_R & 0 \\ \sin\theta_R & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{5}$$

기구학 모델을 바탕으로 로봇의 양쪽 바퀴에 적절한 속도값을 주어서 로봇이 우리가 원하는 방향과 위치로 이동하게 할 수 있다[9].

2. 모바일 로봇의 주행원리

차륜구동형태의 모바일 로봇은 양쪽 바퀴 속도값에 의해 운동상태가 달라진다. 예를 들어 양쪽 바퀴속도가 1m/s 라면 (1), (2)에 의해 로봇은 1m/s 속도로 앞으로 직진한다($v_1 = 1$, $v_2 = 0$). 또 오른쪽 바퀴속도가 1m/s 이고 왼쪽 바퀴속도가 (-)1m/s 이면, 로봇은 제자리에서 회전하게 된다 ($v_1 = 0$, $v_2 = 4/l$). 로봇이 회전운동을 하는 경우, 회전운동의 중심점이 존재하는데, 이 점이 순간회전중심(ICC:Instantaneous Center of Curvature)이다. 그림 3(a)에 보인 것처럼 순간회전중심은 바퀴회전축의 연장선들이 만나는 점이 된다[8]. 그림 3(b)에서와 같이 차륜구동형태의 모바일 로봇은 두 바퀴축이 동일 선상에 있으므로, ICC는 바퀴축의 어느 곳이나 위치할 수 있다. 이 경우 ICC 위치는 양바퀴의 속도비에 의해 결정된다.

그림 4는 ICC와 로봇의 속도 및 위치를 표시하였다. 바퀴의 속도값과 로봇에서 ICC까지 거리와의 관계를 살펴보면 (6)과 같이 비례식으로 나타낼 수 있다.

$$v_L : v_R = R - \frac{l}{2} : R + \frac{l}{2} \tag{6}$$

(6)은 다음과 같이 정리될 수 있다.

$$R = \frac{l}{2} \left(\frac{v_R + v_L}{v_R - v_L} \right) \tag{7}$$

즉, 모바일 로봇의 회전반경은 좌우바퀴속도값에 의해서 결정된다. 위의 식을 통해서 로봇이 직진 운동을 할 경우는

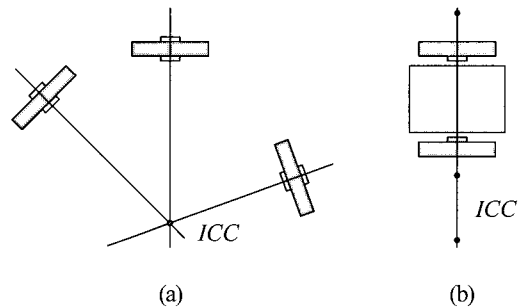


그림 3. 순간회전중심.
Fig. 3. Instantaneous center of curvature.

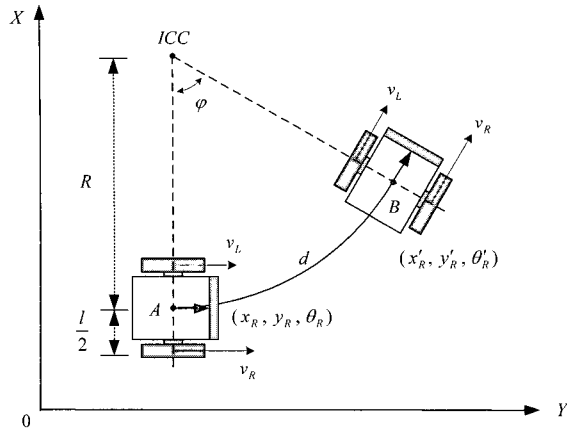


그림 4. 모바일 로봇의 ICC.
Fig. 4. ICC of mobile robot.

$R = \infty (v_R = v_L)$ 이고, $v_R \neq v_L$ 이면 로봇은 원궤적을 그리면서 주행하게 된다.

그림 4에서 모바일 로봇이 A 지점에서 B 지점으로 이동할 때, A 지점(시간= t)에서의 로봇의 위치는 (x_R, y_R, θ_R) 이고, B 지점(시간= $t + \delta t$)에서의 로봇의 위치는 (x'_R, y'_R, θ'_R) 이다. ICC 의 좌표는 (8)과 같이 정의된다.

$$ICC = [x_R - R \sin(\theta_R), y_R + R \cos(\theta_R)] \quad (8)$$

시간이 t 에서 δt 만큼 경과했을 때, 모바일 로봇의 위치 (x'_R, y'_R, θ'_R) 는 다음식에 주어진다[8].

$$\begin{bmatrix} x'_R \\ y'_R \\ \theta'_R \end{bmatrix} = \begin{bmatrix} \cos(\omega \delta t) & -\sin(\omega \delta t) & 0 \\ \sin(\omega \delta t) & \cos(\omega \delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_R - ICC_x \\ y_R - ICC_y \\ \theta_R \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega \delta t \end{bmatrix} \quad (9)$$

(9)는 모바일 로봇의 위치를 ICC 의 위치와 각속도, ω 에 대한 식으로 표시한 것이다. A 지점에서 B 지점까지 모바일 로봇의 총 이동거리, d 와 회전각도, ϕ 를 구해보면 다음과 같은 식을 구할 수 있다[8].

$$d = \int_t^{t+\delta t} v_1 dt = \int_t^{t+\delta t} \frac{v_L + v_R}{2} dt \quad (10)$$

$$\phi = \frac{d}{R} = \frac{\int_t^{t+\delta t} (v_L + v_R) dt}{l(v_L + v_R)} (v_R - v_L) \quad (11)$$

위의 (10), (11)을 바탕으로 모바일 로봇의 회전운동시 회전반경과 이동거리, 회전각도, 그리고 로봇의 선형속도 및 회전속도값을 얻을 수 있다.

III. 단일곡률궤적

1. 곡률

곡률(curvature)이란 그림 5에서처럼 일정속력으로 곡선위의 한 점(P)에서 다른 점(Q)으로 곡선을 따라 이동할 때, 얼마나 빠른 속도로 곡선을 회전하는 것의 척도이다. 곡률을 k 로

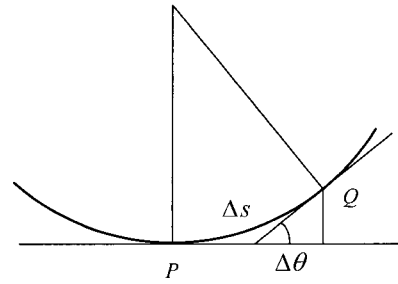


그림 5. 곡선의 곡률.
Fig. 5. Curvature.

표시했을 때, 다음과 같이 나타낼 수 있다[10].

$$k = \lim_{\Delta s \rightarrow 0} \left| \frac{\Delta \theta}{\Delta s} \right| = \left| \frac{d\theta}{ds} \right| \quad (12)$$

곡률반경은 일정한 곡률을 가진 원의 반경으로 회전반경이라고도 한다. 곡률반경을 ρ 는 (13)과 같이 정의된다.

$$k > 0 \text{ 이면 } \rho = 1/k \quad (13)$$

(12)을 살펴보면, 곡선을 따라서 총 이동한 거리인 Δs 는 곡선의 호의 길이이므로 곡선의 반지름에 비례한다. 곡률 k 는 Δs 에 반비례 하므로 전체적으로 곡률 k 는 곡선의 반지름에 반비례하게 된다. $k=0$ 이면, 곡률반경은 무한이다. $k=0$ 이면 곡선은 구부러지지 않으므로 직선이 된다. 직선은 무한반경을 가진 원으로 생각할 수 있다.

모바일 로봇이 곡률궤적을 따라서 주행할 때, 회전반경(혹은 곡률)은 주행오차에 크게 영향을 미친다. 모바일 로봇은 일반적으로 직선주행시 ($k=0$) 보다는 곡선주행시 ($k \neq 0$) 에 많은 오차를 발생시킨다. 이론적인 곡선주행궤적은 (7)-(9)을 이용해서 계산할 수 있는데, 순수 구름조건(pure rolling)과 미끄러짐이 없는 조건(non-slipping)을 만족한다고 가정하므로 이상적인 주행조건이 주어지지 않는 실제 주행과는 차이를 보이게 된다. 로봇이 곡선을 주행한다고 가정하면 모바일 로봇에는 구심력과 원심력이 동시에 작용한다. 이때 로봇의 바퀴와 지면과의 마찰력이 로봇에서 ICC 쪽으로 작용하는 구심력이 되어 곡선운동을 유지시켜 주는데, 이상적인 조건하에서는 이 때 지면과의 미끄러짐이 없으므로 주행오차가 0이 되지만, 실제적 환경에서는 미끄러짐으로 인해 주행오차가 발생한다. (13)을 통해서 구심력은 회전반경과 속도에 의해 영향을 받는다는 것을 알 수 있다[6].

$$F = k \frac{mv^2}{r} \quad (14)$$

그림 6은 모바일 로봇이 A 에서 출발하여 곡률궤적을 따라서 곡선주행을 하는 경우를 보여준다. 이상적인 조건하에서 로봇의 예상위치는 B1 이 되나, 실제로는 B2 위치에 위치하여 주행오차가 발생하는 것을 보여준다. 회전반경 및 주행속도가 주행오차에 미치는 영향에 대해서는 많은 연구가 선행되었다[6].

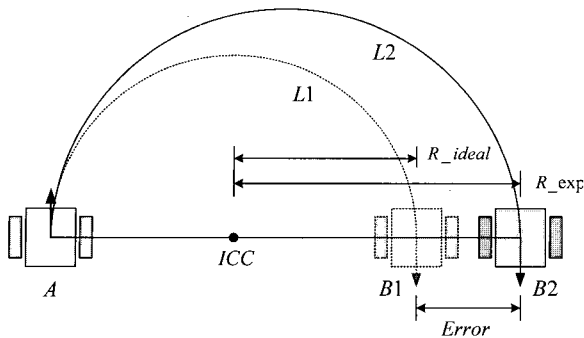


그림 6. 모바일 로봇의 곡률주행시 주행오차.
Fig. 6. Driving error of mobile robot on curvature path.

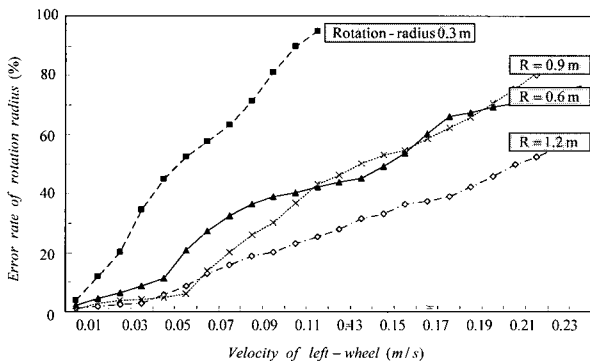


그림 7. 회전반경크기에 따른 주행오차.
Fig. 7. Driving errors in terms of speed and curve radius.

그림 7은 회전반경과 주행속도에 따라서 주행오차가 어떻게 변하는지를 보여주는 그래프이다. 오른쪽 바퀴 속도를 일정하게 하고, 좌측 바퀴 속도를 조정하여 로봇의 속도변화를 주었다. 좌측 바퀴 속도가 증가할수록 주행오차가 증가하고, 또한 회전반경이 작을수록 같은 속도에서도 주행오차가 큰 것을 알 수 있다. 그림 7에 나타나는 연구결과는 모바일 로봇이 곡선주행할 때, 회전반경이 작을수록 그리고 주행속도가 빠를수록 주행오차가 크다는 것을 보여준다[6].

2. 단일곡률궤적

그림 8(a),(b)은 단일곡률궤적과 더블곡률궤적을 나타내고 있다. 더블곡률은 단일곡률이 변곡점(point of inflection)을 중심으로 대칭형태로 이루어져 있다. 단일곡률궤적은 곡률반경이 같은 값을 가지지만, 더블곡률궤적은 변곡점에서 주행방향과 곡률반경이 변화한다. 그림 8(c)는 주행방향과 곡률반경이 계속해서 변하는 경우이다. 즉, 변곡점이 여러군데 존재한다는 의미이다. 모바일 로봇이 그림 8(c)와 같은 궤적을 따라서 주행한다고 가정한다면, 곡률반경과 주행방향이 변화할 때마다 (7)에 의해서 모바일 로봇의 좌우바퀴의 속도값이 계속 변해야 한다는 것을 알 수 있다.

또한 비슷한 거리를 주행한다고 할 때, 그림 8(a)의 단일곡률궤적이 회전반경의 크기가 가장 큰 것을 알 수 있다. 상대적으로 나머지 궤적은 회전반경의 크기가 변할 뿐만 아니라, 크기도 작다는 것을 알 수 있다. 단일곡률궤적이 회전반경이 가장 크므로, 그림 7에서 보듯이, 단일곡률궤적이 가장 작은 주행오차를 가질 것으로 예측할 수 있다. 실제로 실험을 통

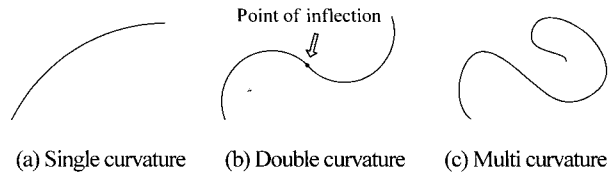


그림 8. 곡률.
Fig. 8. Type of curvature.

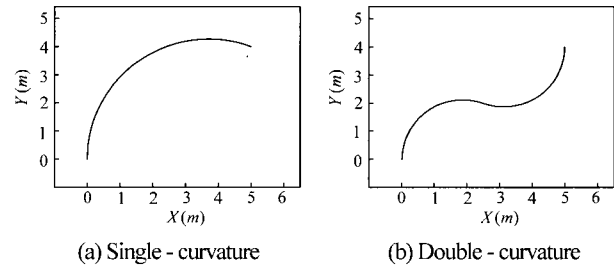


그림 9. 단일곡률궤적과 더블곡률궤적.
Fig. 9. Single-curvature and double-curvature trajectory.

표 1. 단일곡률궤적과 더블곡률궤적의 계산값.

Table 1. Computed single-curvature and double-curvature trajectory.

	Single Curvature Trajectory	Double Curvature Trajectory
Initial position	P=(0, 0, 90°)	P=(0, 0, 90°)
Final position	P=(5, 4, 109.6°)	P=(5, 4, 90°)
Path distance (m)	7.18 m	7.18 m
Rotation radius (m)	3.75 m	1.88 m
Rotation angle(deg)	109.65°	$\phi_1 = 109.65^\circ$ $\phi_2 = 109.65^\circ$
Position of ICC	(3.75, 0)	ICC1=(1.87, 0) ICC2=(3.13, 4)

해서 주행오차의 차이를 확인할 수 있는데, 실제 시스템에 적용하기 위해 시뮬레이션(simulation)을 통해서 회전각도, 주행거리, 궤적 등을 살펴보았다.

그림 9는 단일곡률궤적과 더블곡률궤적을 시뮬레이션을 통해 나타낸 것이다. (7)-(11)를 통해서 각각 궤적에 대한 회전반경, 주행거리, 회전각도, ICC의 위치등을 구할 수 있고, 그 값들을 표 1에 나타내었다.

시뮬레이션의 결과를 살펴보면 단일곡률궤적과 더블 곡률궤적은, 총 주행거리는 동일하며, 회전반경은 단일곡률궤적은 3.75m 인데 비해, 더블곡률궤적은 그의 절반인 1.875m 이다. 그리고 더블곡률궤적은 (2.5, 2)에 변곡점이 있다. 회전반경과 주행오차의 관계를 나타낸 연구 결과(그림 7)를 토대로, 단일곡률궤적의 주행오차가 더블곡률궤적보다 작아짐을 예상할 수 있다.

IV. 이동물체의 추적 및 포획 알고리즘

이 장에서는 그림 10에서와 같이 단일곡률궤적을 이용한 이동물체의 포획 경로 및 알고리즘을 제시한다. 일반적으로 이동물체의 포획에 있어서 이동물체와 모바일 로봇의 속도 및 방향에 따라 모바일 로봇의 주행경로는 매우 다양하게 나

타난다. 본 논문에서는 모바일 로봇의 한계(최대) 속도 및 한계(최대) 가속도를 이용한 최소시간 요구조건과 주행오차를 고려한 안정성 요구조건을 동시에 만족하는 주행경로를 선정시켰다.

1. 주행경로계획을 위한 제한 조건

이동물체를 효율적으로 포획하기 위해서 단일곡률계적 2개와 하나의 직선계적으로 경로를 구성하였으며, 이동물체는 일정한 속도, v_{obj} 로 직선운동을 한다고 가정하였다. 또한 모바일 로봇은 정지상태에서 출발하여 능동카메라를 이용해 이동물체의 속도와 예상 위치를 추정한 후, 물체의 뒤에서 안전하게 포획하도록 하였다.

모바일 로봇의 왼쪽 바퀴의 속도와 가속도를 각각 v_L, a_L 이라 하고, 오른쪽 바퀴의 속도와 가속도를 각각 v_R, a_R 이라 정의하면, 그 한계범위는 (15), (16)과 같이 표현할 수 있다.

$$|v_L|, |v_R| \leq v_{max} \tag{15}$$

$$|a_L|, |a_R| \leq a_{max} \tag{16}$$

여기서 v_{max} 는 모바일 로봇의 양 바퀴의 최대 허용 속도이고, a_{max} 은 모바일 로봇의 양 바퀴의 최대 허용가속도이다[11].

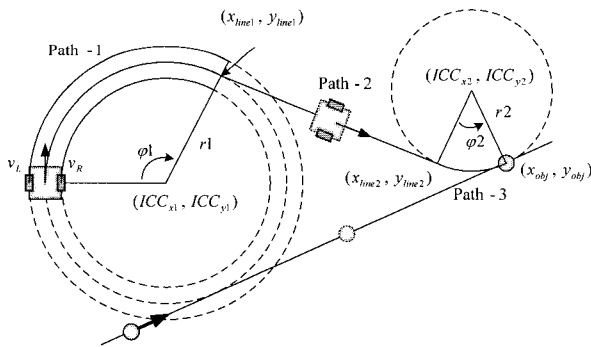


그림 10. 단일곡률계적을 이용한 포획경로.
Fig. 10. Capturing trajectory using single-curvature trajectory.

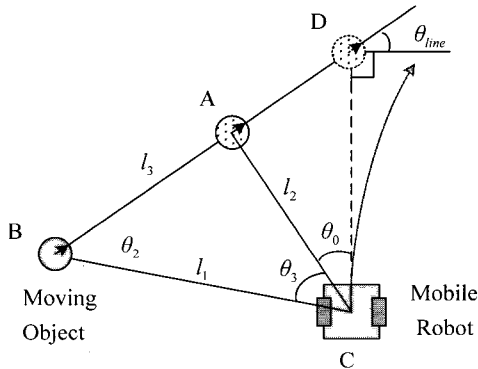


그림 11. 모바일 로봇의 이동물체 추적 알고리즘.
Fig. 11. Trajectory algorithm of mobile robot for capturing the moving object.

2. 모바일 로봇의 경로계획

모바일 로봇이 곡률계적을 따라서 이동물체를 추적할 때는 곡률반경에 따라서 로봇의 경로와 추적시간이 달라지므로 곡률반경을 고려한 경로계획을 세워야 한다. 이동물체는 일정한 속도로 직선운동을 한다고 가정하였으므로, 그림 11에서와 같이 모바일 로봇의 능동카메라를 이용해서 이동물체의 위치를 예측할 수 있다[11].

B 와 C 사이의 거리를 l_3 이라 하면, 다음과 같은 관계식을 나타낼 수 있다.

$$l_3 = l_1^2 + l_2^2 - 2l_1l_2 \cos(\theta_3) \tag{17}$$

$$\theta_2 = \cos^{-1} \left(\frac{l_1^2 + l_3^2 - l_2^2}{2l_1l_3} \right) \tag{18}$$

$$\theta_3 = \cos^{-1} \left(\frac{l_1^2 + l_2^2 - l_3^2}{2l_1l_2} \right) \tag{19}$$

(17)-(19)를 바탕으로 D 지점에서의 각도 θ_{line} 은 (21)과 같다.

$$\theta_2 + (\theta_3 + \theta_0) = \theta_{line} + 90^\circ \tag{20}$$

$$\theta_{line} = \theta_2 + \theta_3 + \theta_0 - (\pi/2) \tag{21}$$

곡률계적에서 모바일 로봇은 이동물체의 예상위치 D 와 θ_{line} 을 추정하고, θ_{line} 이 클수록 큰 곡률반경을 가지는 계적으로 움직이게 된다.

2.1 첫번째 회전 구간 Path-1의 경로계획

모바일 로봇은 그림 12에서와 같이 가장 먼저 단일곡률계적인 Path-1을 따라서 이동한다. 모바일 로봇이 정지 상태에서 이동하므로, Path-1은 가속구간으로 선정하였다.

이동물체를 추적하는데 있어서 물체의 초기 위치와 움직임의 방향이 로봇의 어느쪽에 위치하는냐에 따라서 모바일 로봇의 경로는 달라진다. 모바일 로봇에 장착된 능동카메라의 Pan각을 이용하여 이동물체의 위치가 로봇의 좌우 어느 방향에 있는지 인식하고, 로봇이 그 방향으로 회전하여 출발하도록 한다[11]. 주어진 경로에서 로봇은 시계방향(오른쪽)으로 움직이므로, 모바일 로봇의 왼쪽 바퀴는 반경이 $r1 + (l/2)$ 인 곡률계적을, 오른쪽 바퀴는 $r1 - (l/2)$ 인 곡률계적을 따라 움직인다. 즉, 왼쪽바퀴의 속도가 오른쪽 바퀴속도보다 빨라야 한다($v_L > v_R$).

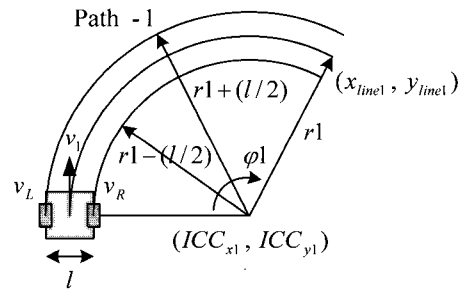


그림 12. Path-1에서의 로봇의 주행경로.
Fig. 12. Trajectory path of mobile robot on Path-1.

$$v_R = \int a_R(\tau) d\tau = a_R t \quad (22.1)$$

$$v_L = \int a_L(\tau) d\tau = a_L t \quad (22.2)$$

위의 속도와 가속도와의 관계식에서 볼 수 있듯이 속도와 가속도는 비례하므로, $a_L > a_R$ 이다. 추적시간을 최소화하기 위해 속도가 빠른 로봇의 왼쪽 바퀴를 기준으로 하여 최대가 속도, $a_L = a_{max}$ 로 로봇은 이동한다. (x_{line1}, y_{line1}) 에서 모바일 로봇은 직선궤적($v_L = v_R$)으로 전환하며, 이를 위해 로봇은 (23)을 만족하는 v_L, v_R 값을 가질 때까지 Path-1를 수행하게 된다.

$$v_L - v_R \leq 2a_{max} \quad (23)$$

또한, (23)-(26)를 통해 다음의 조건들을 만족하는 $r1, \phi1$ 을 설정한다.

$$v_L = (r1 + \frac{l}{2})\phi1' \quad (24)$$

$$v_R = (r1 - \frac{l}{2})\phi1' \quad (25)$$

$$r1 = \frac{l}{2} \left(\frac{v_L + v_R}{v_L - v_R} \right), \quad r1 \geq \frac{l}{2} \quad (26)$$

2.2 직선궤적 Path-2의 경로계획

모바일 로봇이 직선궤적인 Path-2을 이동할때, 이동물체의 매시간의 예상위치를 예측하고, 그 중 물체와 거리가 가장 가까워지는 이동물체의 위치를 목표점으로 설정하여 이동한다. 그림 13에서와 같이 이동물체의 예상위치, $\hat{x}_o(k+m)$ 와 모바일 로봇의 예상 위치, $\hat{x}_R(k+m)$ 의 차가 최소값이 되는 경로를 모바일 로봇의 Path-2에서의 경로로 설정한다[12].

$$(x_{line2}, y_{line2}) = \min_{m=1-R} \|\hat{x}_o(k+m) - \hat{x}_R(k+m)\| \quad (27)$$

2.3 두번째 회전구간 Path-3의 경로계획

모바일 로봇이 Path-3을 이동할 때는 회전 운동시의 오차를 줄이기 위해서 등속도 운동을 하며, (x_{obj}, y_{obj}) 에서 이동물체를 포획하기 위해 이동물체와 같은 속도를 가지도록 한다($v_L = v_R = v_{obj}$). 그림 11에서와 같이 이동물체와 모바일 로봇의 예상 위치를 파악하고 θ_{line} 의 크기를 고려하여 이동로봇과 모바일 로봇과의 거리가 최소가 되는 곡률로 회전하도록 한다.

단일곡률궤적을 이용한 이동물체를 포획하기 위한 전체 알고리즘은 먼저, 카메라를 이용하여 Path-1에서의 로봇의 이동방향을 결정하고, 최대 가속도로 이동한후, (23)를 만족하는 속도에 이르렀을 때 Path-2의 직선궤적으로 이동한다. 직선궤적에서는 로봇과 이동물체의 거리가 최소가 되는 위치를 예측하여 이동하며, 마지막 Path-3에서 로봇은 이동물체와 같은 속도를 유지하면서 이동물체를 포획하게 된다.

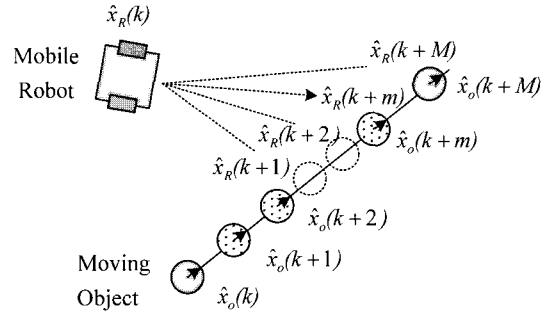


그림 13. Path-2 에서의 로봇의 주행경로.
Fig. 13. Trajectory path of mobile robot on Path-2.

V. 실험 및 실험결과

1. 실험구성

실험은 연구실 내부에서 수행하였다. 바닥은 일반적인 건물 내부의 콘크리트 바닥으로 바닥의 요철에 의한 영향은 무시하였다. 또한 실험시 다른 외부조건은 동일하다고 가정하였다.

• 이동물체: 이동물체는 특별한 기능이 필요하지 않으므로, 그림 14에 나타난 것처럼 간단하게 구성하였다. 이동물체는 2-자유도를 가지는 차륜구륜 형태의 로봇으로 제작하였고 모터는 Swiss minimotor사의 DC motor 2개를 사용하였다. 시스템은 80C196KC microprocessor로 구성하였고, 모터드라이브는 L298을 이용하여, 간단히 PWM(Pulse Width Modulation) 신호로 DC모터를 제어할 수 있도록 설계하였다. 배터리는 리튬이온 (Li-ion) 전지를 이용해서 로봇 구동시 독립적인 전원공급이 가능하도록 하였다. 이동물체의 위치 인식을 위해서, 이동물체의 윗부분에 빨간색 Color Patch를 그림 14(b)와 같이 부착하였다. Color Patch의 색상정보는 모바일 로봇에 장착되어 있는 CCD 카메라에 의해 획득되며, 모바일 로봇이 이 정보를 바탕으로 이동물체의 위치를 파악한다. 이동물체에 대한 하드웨어 제원을 표 2에 자세히 나타내었다.

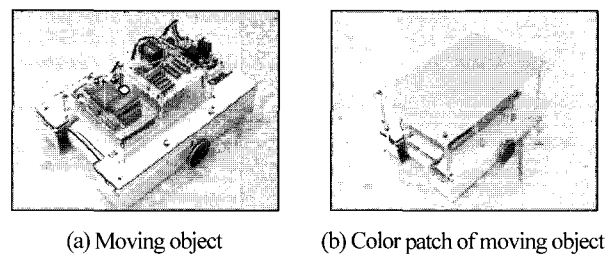


그림 14. Path-1에서의 로봇의 주행경로.
Fig. 14. Trajectory path of mobile robot on Path-1.

표 2. 이동물체의 하드웨어 제원.

Table 2. Hardware specifications of the moving object.

List	Specification
Size (mm)	246 * 191 * 113 (L*W*H)
Weight (kg)	0.95 kg
Distance between wheels (mm)	184 mm
Radius of wheel (mm)	19 mm

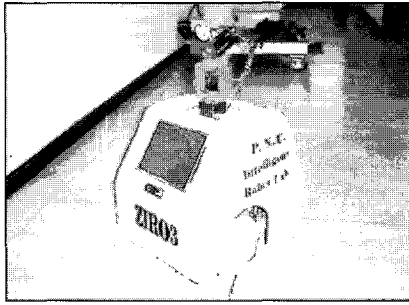


그림 15. 모바일 로봇의 실제 모습.
Fig. 15. Photograph of the mobile robot.

표 3. 모바일 로봇의 하드웨어 제원.
Table 3. Hardware specifications of the moving object.

List	Specification
Size (mm)	660 * 430 * 465 (L*W*H)
Weight (kg)	19.5 kg
Distance between wheels (mm)	400 mm
Radius of wheel (mm)	105 mm

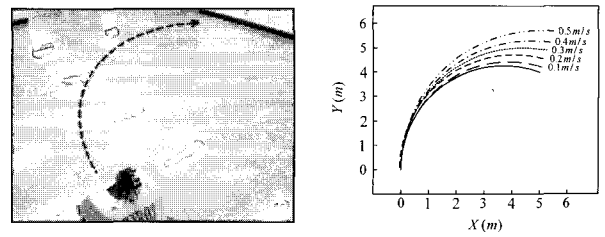
표 4. 각각의 궤적에 대한 모바일 로봇의 이론적인 속도값.
Table 4. Computed velocity of the mobile robot for each trajectory.

Lists		Single-curvature Trajectory (r=3.75m)		Double-curvature Trajectory (r=1.875*2m)	
v_1 (m/s)	Time(s)	v_L (m/s)	v_R (m/s)	v_L (m/s)	v_R (m/s)
0.1	72.8	0.105	0.095	0.089	0.111
0.2	37.9	0.211	0.189	0.179	0.221
0.3	26.9	0.316	0.284	0.268	0.332
0.4	21.9	0.421	0.379	0.357	0.443
0.5	19.3	0.527	0.473	0.447	0.553

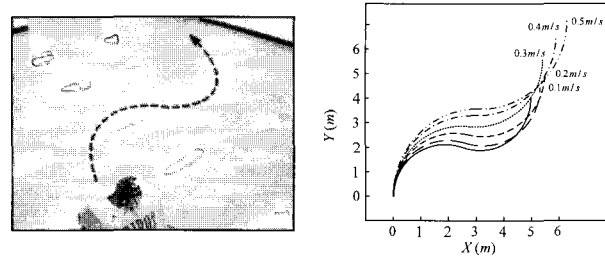
• 모바일 로봇: 실험에 사용된 모바일 로봇은 그림 15에 서와 같이 2자유도의 능동카메라와 센서가 부착되어 있어서 물체 및 주위환경인식이 가능하며, 자율 주행이 가능하도록 제작되었다. 시스템은 CAN 컨트롤러가 내장된 2개의 하위제 어기(87C196CA)와 Pentium III의 상위제어기로 구성되었고, 각 제어기는 네트워크로 연결되어 있다. 상위제어기에서는 센서정보와 로봇의 상태정보 및 환경정보로부터 로봇의 자세 및 속도명령을 결정하여 하위 제어기로 전송하고, 하위제 어기는 상위제어기가 내린 명령에 대하여 분산적으로 제어를 수행하는 구조이다. 모바일 로봇에 대한 하드웨어 제원은 표 3에 나타나 있다[11].

2. 단일곡률궤적 실험

첫번째 실험은 모바일 로봇이 단일곡률궤적을 따라서 곡 선주행할때와 더블곡률궤적을 따라서 곡선주행 할때의 주행 오차, 주행거리, 주행시간을 비교한 것이다. 모바일 로봇의 선형속도를 0.1m/s 에서 0.5m/s 까지 분류하여 그림 9의 궤적 을 따라서 움직이도록 하였다. 표 4는 시뮬레이션 결과값과 (1)-(7)을 바탕으로 양 바퀴 속도값과 주행시간 등을 정리해 놓은 것이다. 수식 및 시뮬레이션을 통해서 구한 이론적인 값들과 실제 실험으로 얻어진 값들을 비교해 궤적에 따른 주



(a) Single curvature



(b) Double curvature

그림 16. 모바일 로봇의 실제 주행.
Fig. 16. Real experiments of the mobile robot driving.

표 5. 단일곡률궤적 주행시 실제 회전반경, 주행오차, 주행 시간, 주행거리.
Table 5. Rotation radius, driving error, time, and distance along the single-curvature trajectory.

v_1 (m/s)	R_real(m)	Error(m)	Time(s)	Distance(m)
0.1	3.88	0.13	74.0	7.31
0.2	4.13	0.38	39.5	7.54
0.3	4.37	0.62	28.8	7.75
0.4	4.61	0.86	24.0	7.98
0.5	4.98	1.23	21.5	8.32

표 6. 더블곡률궤적 주행시 실제 회전반경, 주행오차, 주행 시간, 주행거리.
Table 6. Rotation radius, driving error, time, and distance along the double-curvature trajectory.

v_1 (m/s)	R1_real (m)	R2_real (m)	Error (m)	Time (s)	Distance (m)
0.1	2.02	2.06	0.51	77.2	7.67
0.2	2.25	2.25	1.02	41.6	8.08
0.3	2.47	2.38	1.44	32.1	8.12
0.4	2.85	2.89	2.43	28.8	8.80
0.5	3.12	3.15	3.01	26.2	9.33

행의 정확도를 살펴볼 수 있다. 모바일 로봇의 속도는 5단계 로 구분했으며, 가속도는 0.05m/s² 로 정하였다.

그림 16은 모바일 로봇의 실제 주행 실험 모습을 나타내고 있고, 표 5와 표 6에 단일곡률궤적 및 더블곡률궤적 주행 실험의 결과값을 정리하였다. 그림 17은 실험결과값을 바탕으 로 모바일 로봇의 속도에 대한 주행오차, 주행거리, 주행시간 을 그래프로 나타낸 것이다.

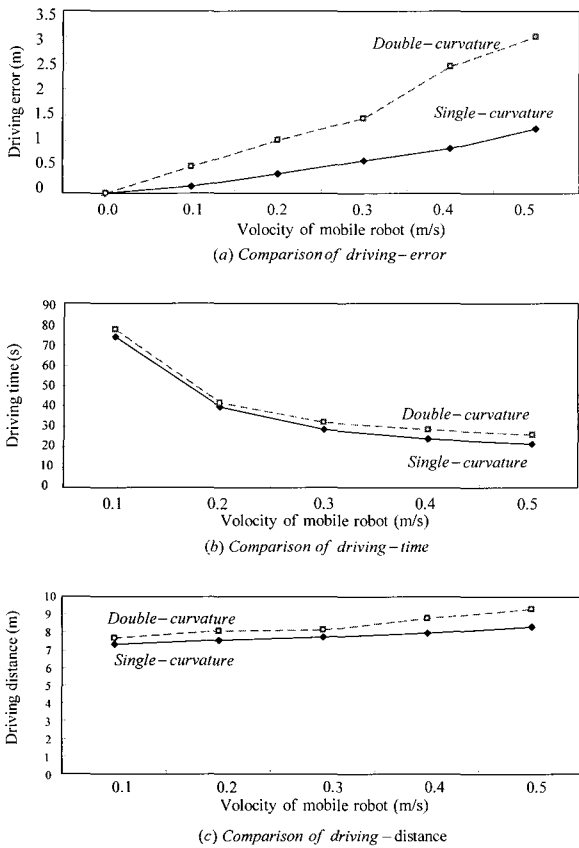


그림 17. 단일곡률궤적과 더블곡률궤적의 주행비교 그래프.
Fig. 17. Comparison of single curvature and double curvature driving characteristics.

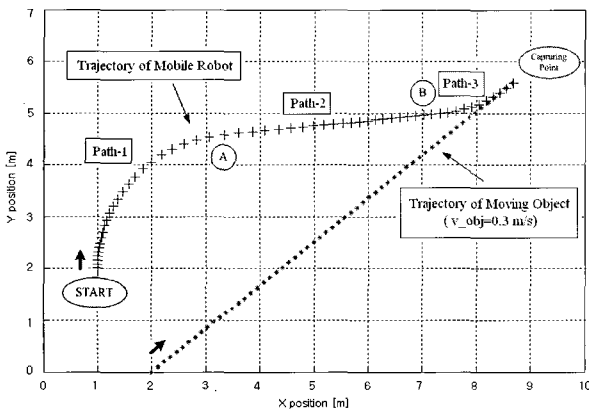


그림 18. 실제 추적 및 포획 궤적 경로.
Fig. 18. Real tracking and capturing trajectory.

3. 이동물체의 포획실험

두번째 실험은 그림 10에서 제시한 모바일 로봇의 이동물체 추적 및 포획 실험이다. 이동물체는 0.3m/s 로 등속도 운동을 하며, 물체의 초기 위치는 (2, 0, 40°)로 설정하였다. 모바일 로봇의 초기 위치는 (1, 2, 90°)이며, 최대가속도는 $a_{max} = 0.042m/s^2$, 최대속도는 $v_{max} = 0.588m/s$ 로 제한하였다. 그림 18은 포획 실험결과로써 실제 이동물체와 모바일 로봇의 이동경로를 위치 데이터를 바탕으로 0.5sec 간

격으로 표시한 것이다. 모바일 로봇은 가속구간인 Path-1을 따라 곡률반경이 2.6m 인 첫번째 곡률궤적을 주행하였다. 모바일 로봇의 오른쪽에 이동물체의 위치하였으므로 로봇은 오른쪽으로 회전하게 된다. 왼쪽 바퀴는 최대 가속도인 $0.042m/s^2$ 값을 가졌으며, 오른쪽 바퀴의 가속도는 $0.036m/s^2$ 이었다. A 지점에서 모바일 로봇은 (23)를 만족하게 되어 직선궤적인 Path-2로 주행궤적을 변경하였다. 직선궤적에서 로봇은 이동물체와의 거리를 줄이면서 B 지점까지 이동하였다. 보다 안정적인 주행을 위해 직선궤적에서 로봇은 속도를 이동물체와 같은 속도로 감속($0.546m/s \rightarrow 0.3m/s$)하였고, B 지점에서 Path-3의 곡률궤적을 따라서 이동하였다. 곡률반경은 2.4m 였으며, 이동물체와 같은속도($0.3m/s$), 같은 각도($40deg$)를 유지한 채 이동물체를 뒤에서 포획하였다.

포획까지 소요된 시간은 30.2sec 였다. 단일곡률궤적과 직선궤적으로 이루어진 포획경로를 따라서 모바일 로봇이 이동물체를 효율적으로 추적하고 포획하는 것을 실험을 통해 확인할 수 있었다.

VI. 결론

본 논문은 모바일 로봇의 경로계획에 있어서 단일곡률궤적을 제시하고, 더블곡률궤적과 비교하여 단일곡률궤적이 주행오차, 주행거리 및 주행시간에 있어서 효율적인 궤적임을 시뮬레이션과 실험을 통해서 알아보았다. 모바일 로봇은 DR (Dead reckoning) 센서를 사용하므로 지속적인 이동으로 인해서 오차가 누적되고, 이 오차를 보정하기 위해서는 복잡한 제어이론이 필요하다. 또한, 잦은 오차 보정은 주행상태를 불안하게 하는 요소가 될 수 있다. 경로계획에 있어서 주행오차를 줄일 수 있는 궤적을 선정함으로써 주행오차 발생의 가능성을 줄일 뿐만 아니라 주행시간을 줄이는 효과가 있음을 보였다. 이렇듯, 효율적인 주행경로선정만으로도 로봇의 주행오차를 줄일 수 있음을 알아보았다. 단일곡률궤적을 주행하면서 위치추정의 정도를 높이는 알고리즘이나 정밀제어기법을 사용한다면, 모바일 로봇의 주행정확도는 더욱 더 향상될 것이다.

참고문헌

- [1] K.-C. Henry Fok and M. R. Kabuka, "A flexible multiple mobile robots system," *IEEE Transaction on Robotics and Automation*, volume 8, no. 5, pp. 607-623, Oct. 1992.
- [2] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, John Wiley & Sons, 1989.
- [3] K. Hirai, M. Hirose, Y. Haikawa and T. Takenaka, "The Development of Honda Humanoid Robot," in *Proc. IEEE Int. Conf. on Robotics and Automations*, pp. 1321-1326, 1998.
- [4] 문종만, "2 자유도 구륜 이동 로봇의 기구학적 보정과 궤적추적에 관한 연구," 박사학위논문, 경희대학교, 2000.
- [5] U. Larsson, J. Forsberg, and A. Wernersson, "Mobile robot localization: integrating measurements from a time-of-flight laser," *IEEE Transaction on Industrial Electronics*, vol. 43, no. 3, pp. 422-431, June 1996.
- [6] Y.-J. N, *A Study on the reduction of Navigation error for Curvilinear navigation of Mobile Robot using Neural Network*, 석사학위논문, 전남대학교, 2002.

[7] K.-T. Song and W.-H. Tang, "Environment perception for a mobile robot using double ultrasonic sensors and a CCD camera," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 3, pp. 372-379, June 1996.

[8] G Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*, Cambridge university press, 2000.

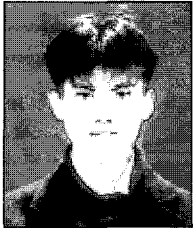
[9] W. E. Dixon, D. M. Dawson, E. Zergeroglu and A. Behal, *Nonlinear Control of Wheeled Mobile Robots*, Springer, 2003.

[10] S. I. Grossman and W. R. Derrick, *Advanced engineering*

mathmatics, HarperCollins Publishers, 2000.

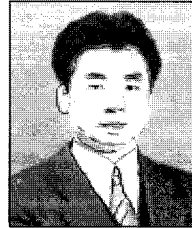
[11] 황철호, 이상헌, 조방현, 이장명, "이동물체 포획을 위한 최적 경로 계획," 제어·자동화·시스템공학 논문지 제 10 권 8호, pp. 696-702. 2004.8.

[12] 박진우, 박재한, 윤경식, 이장명, "이동로봇에 장착된 능동 카메라를 이용한 이동물체의 추적과 포획," 제어·자동화·시스템공학 논문지 제7권 제9호, pp. 741-748, 2001. 9.



최 병 석

2004년 부산대학교 전자전기통신공학부 졸업. 2004년~현재 부산대학교 대학원 전자공학과 석사과정 재학중. 관심분야는 로봇위치인식, 유비쿼터스 기반 지능형 로봇.



이 장 명

1957년 11월 22일생. 1980년 서울대학교 전자공학과 졸업(공학사). 1982년 서울대학교 대학원 전자공학과(공학석사). 1990년 University of Southern California 전자공학과(공학박사). 1992년~현재 부산대학교 전자전기통신공학부 교수. 관심분야는 로보틱스, 지능제어.

심분야는 로보틱스, 지능제어.