

〈論 文〉

이동로봇의 실시간 주행제어를 위한 제어시스템 설계 및 경로 추종제어 방법

고경철* · 조형석**

(1993년 1월 23일 접수)

Real-Time Control System for Mobile Robots and Path Tracking Control Algorithm

K. C. Koh and H. S. Cho

Key Words : Mobile Robot(이동로봇), Control System(제어시스템), Path Tracking Control(경로추종 제어), Landmark Recognition(랜드마크 인식), Neural Network(신경회로망)

Abstract

Real-time mobile robot controllers usually have been designed focused on control theory without paying attention to the importance of system integration. This paper demonstrates that autonomous mobile robots require a real-time controller with a wide range of capabilities in addition to control theory. An architectural frame work supporting these capabilities has been designed in actual hardware environments. Individual modules such as a path planner, a path tracking controller, position estimators, wheel controllers and other crucial elements have been successfully integrated into the control system using this frame work. The overall performance of the system was investigated via a series of tracking experiments with a prototype mobile robot named LCAR developed in the laboratory. The context of the research involves the architecture, its implementation and experimental results.

1. 서 론

최근 고정 궤도식 무인 운반차(AGV)의 결점을 보완하고 유연성을 높이기 위해 자율 주행(free-ranging) 방식을 채택한 이동 로봇(autonomous mobile robot)에 관한 연구가 활발하다. 이러한 자율 주행 방식은 목적 경로를 탑재 컴퓨터의 메모리에 기억시킴으로써 경로 변경이 용이하며, 충돌 회피 등 돌발적인 상황에 유연하게 대처할 수 있는

장점을 지니고 있다.

반면 이동로봇이 주어진 기준경로를 정확히 따라 주행하기 위해서는 로봇 스스로 매 제어주기마다 로봇의 현재위치 및 진행방향을 추정하는 자기 위치추정 문제와 이를 토대로 다음 제어주기에서의 조향각 및 속도를 결정하는 경로추종 제어문제 등에 어려움이 있다. 따라서 이를 위한 각종 위치센서 시스템 구성 및 위치추정 알고리즘, 그리고 경로로부터 벗어난 현재의 위치로부터 다시 기준경로에 복귀하도록 하는 경로추종 제어알고리즘등에 관한 연구가 자율 이동 로봇의 분야에서 해결해야할 주요 문제가 되고 있다.

*한국과학기술원 정밀공학과

**정회원, 한국과학기술원 정밀공학과

최근 10년간 이러한 이동로봇을 위한 주행제어 시스템에 관한 연구는 하드웨어적 시스템 구성(system integration)을 통한 실시간 제어 측면 보다는 주로 제어이론과 알고리즘 개발을 중심으로 진행되어 왔다.^(1~5) 그 중 Tsumura⁽²⁾는 기준경로를 일련의 기준 점들로 표현하고, 이를 매 제어주기마다 경로의 기준점과 로봇의 예측된 위치를 비교하여, 다음 주기에서의 조향각(steering angle)을 결정하는 경로 제어 방법을 제시한 바 있고, Kanayama⁽³⁾는 기준경로의 표현 방법으로, 기준점들의 배열 대신 경로를 몇 개의 직선 구간들(segments)로 나누어 설계하는 방법을 소개 하였다. 좀 더 부드러운 운동을 얻기 위해 Singh⁽⁴⁾ 등은 경로 궤적 오차를 보상함에 있어, 로봇의 현재 자세와 경로상의 기준자세 사이를 5차 함수 곡선으로 연결하는 quintic polynomials 방법을 경로 제어의 한 방법으로 제시하였다. 제어 시스템의 하드웨어 구조에 관한 연구로는 Crowley등⁽⁵⁾이 3개의 계층구조로 이루어진 이동 제어시스템(locomotion control system)을 제시하였는데, 이 방법에서는 가상의 이동체(virtual vehicle) 개념을 도입하여 직진과 선회운동을 독립적으로 제어함으로써 부드러운 운동 궤적을 얻을 수 있었다.

본 논문에서는 새로운 이동로봇의 제어구조로서

범용 개인용 컴퓨터를 주축으로 설계되는 이동로봇 제어시스템에 대해 기술하고자 한다. 이 제어시스템에서는 자율 이동로봇의 경로제어에 필수적인 경로계획기, 경로추종 제어 알고리즘, 자기 위치 추정기 그리고 휠 제어기 등을 모듈별로 연결/구성 하였으며, 본 연구실에서 연구용으로 개발한 휠 구동 지능형 이동 로봇(LCAR-90)에 실제로 적용하였다. 본 논문은 다음과 같이 구성된다. 제 2 장에서는 이동로봇 LCAR-90의 시스템 묘사와 함께 제어 시스템의 하드웨어적 설계배경에 대해 기술하고, 제 3 장에서는 제어시스템의 필수적인 요소중 하나인 이동로봇의 자기 위치 추정방법과 제어 시스템의 핵심부인 경로추종 제어에 관한 새로운 알고리즘을 적용한 결과에 대해 논의한다. 최종적으로 제 4 장에서는 본 연구를 통해 얻은 결론을 기술 하였다.

2. 전체 시스템

2.1 로봇의 구조 및 사양

이동로봇 LCAR는 크게 몸체와 휠 구동부 그리고 제어 시스템으로 구성되어 있으며, Fig. 1과 같은 외관을 하고 있다. 휠 구동부는 몸체 하단부에 설치되어 몸체의 중량을 기준으로 하여 좌우 2개의

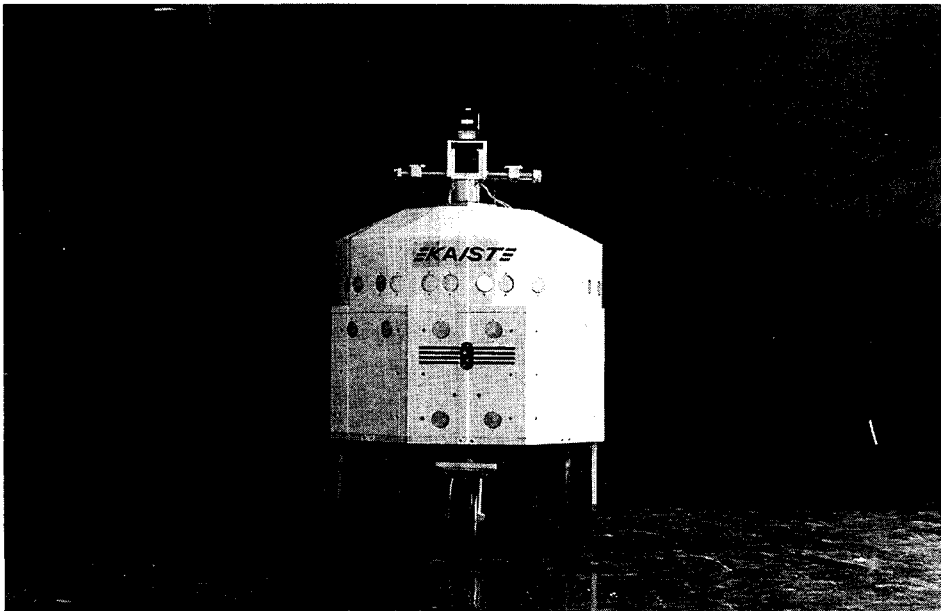


Fig. 1 Overview of the mobile robot(LCAR)

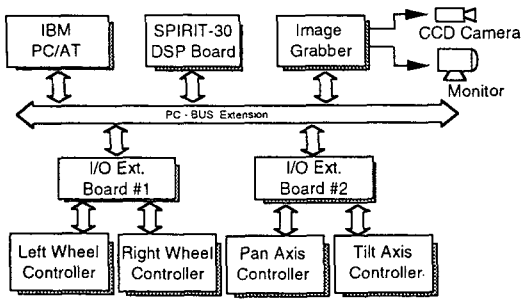


Fig. 2 H/W block diagram of the mobile robot controller

구동 휠 (driving wheels)과 전후 2개의 캐스터 휠 (caster wheels)로 이루어진다. 로봇은 양 구동휠의 속도차에 의해 직진과 선회운동을 하며, 캐스터 휠은 몸체의 진행 방향에 따라 수동적으로 회전하여 몸체의 균형을 유지하는 역할을 한다. 이동로봇을 위한 제어시스템은 Fig. 2에 보이는 바와 같이 주 제어 컴퓨터, 휠 제어부 그리고 센서부 등으로 구성되며 각 하드웨어적 구성요소의 기능은 다음과 같다.

2.2 주 제어 컴퓨터

주 제어 컴퓨터는 전체 시스템을 총괄 제어하는 기능을 한다. 즉 운전중 자동적인 경로계획, 각종 센서의 정보처리, 경로 추종제어 알고리즘을 수행하고, 각 하부 시스템의 동작 상태를 감시한다. 또한 각 하부 시스템간의 데이터를 취합/분배하는 통신 센터의 역할을 한다. 현재 산업용 로봇제어기에서 주 제어부는 16비트 레벨 CPU를 기본으로 하는 단일 기판(single board)형 컴퓨터구조로 설계되고 있으나, 이 구조는 시스템 확장에 제한이 있고, 프로그램의 변경이 자유롭지 못한 단점이 있다. 본 연구에서는 시스템의 하드웨어적 확장성과 외부 기기와의 접속 등을 고려하고, 또한 유연한 시스템 프로그래밍에 설계 초점을 맞추어, 개인용 컴퓨터 (IBM-PC/386)를 이동로봇의 주 제어부로 하는 하드웨어 구조를 제안한다. 따라서 이에 접속되는 휠 제어부와 센서부 등을 포함한 모든 주변장치들은 컴퓨터와 접속될 수 (compatible) 있도록 설계하였으며, PC 확장 버스(bus)를 통해 연결된다. 제어 알고리즘이나 센서 처리 알고리즘 등의 실시간 수행을 위해 실수 연산 전용 DSP보드 (CPU:TMS320C30, 16MHz, 33MFLOPS)를 채택

하여 주 제어 컴퓨터의 연산능력을 강화하였으며, 모든 실시간 관련 소프트웨어 모듈들은 DSP용 C언어로 프로그래밍하여 DSP보드에 다운로드(down loading)하여 수행되도록 하였다.

2.3 휠 구동 제어부

좌우의 구동휠은 벨트/풀리 방식의 감속기를 통해 각각의 배터리식 직류 서보 모터 (용량 : 40 W)에 의해 구동되며, 양 구동 휠축에는 회전당 2000 펄스의 광학식 증가형 엔코더(optical incremental encoder)를 각각 부착하여, 휠의 회전량을 감지할 수 있도록 설계하였다. 구동휠을 제어하기 위한 휠 제어부는 휠 제어 보드, 통신 인터페이스 보드 그리고 모터 드라이브 앰프 등으로 구성된다. 휠 제어 보드는 8032 마이크로 콘트롤러를 이용하여 설계하였고, 각 구동 휠의 위치 및 속도 제어를 담당한다. 주 제어 컴퓨터는 매 제어 사이클 (20 msec)마다 각 구동 휠이 회전해야 할 기준회전각을 각 구동 모터에 부착되어 있는 엔코더의 펄스단위 값으로 변환하여, 통신 인터페이스 보드의 병렬 입출력 포트(PIO)를 통해 각 구동휠의 휠 제어보드로 전달한다. 휠 제어 보드에서는 주 제어 컴퓨터로부터 받은 펄스 데이터를 기준 위치 명령치로 사용하고 엔코더 귀환에 의해 디지털 위치 제어를 수행한다. 제어방으로 선형 비례제어기를 이용한 폐회로 귀환제어와 매개변수 추정에 의한 피드포워드 제어를 채택한다. 전자는 시스템의 안정도를 높이는 기능을 하고 후자는 위치 오차를 줄이는 기능을 한다. 이 휠 제어 보드의 출력은 아날로그 전압 신호로 변환되어 다시 모터 드라이브 앰프로 전달되며, 드라이브 앰프내의 아날로그 속도제어부의 PWM 증폭부를 통해 모터가 구동된다. 속도 제어부에서는 입력신호를 기준속도 신호로 하여 모터의 속도를 제어한다. 여기서 속도 귀환 제어를 위해 구동 모터축에 부착되어 있는 타코미터의 유기전압을 이용한다. 속도 제어 루프는 속도게인(K_v)의 조절을 통해 제어 플랜트의 감쇄효과를 키워주는 기능한다.

2.4 센서부

이동로봇은 로봇 암(arm) 매니퓰레이터(manipulator)와는 달리 보다 넓은 범위의 공간에서 작업을 하게 되므로 될 수 있는 한 주위 환경에 대한 많은 정보를 필요로 한다. 본 연구에서 사용되는

이동로봇의 센서부는 센서의 감지범위와 각각의 특성을 고려하여 절대위치 추정을 위한 시각센서, 근접 장애물 탐지를 위한 초음파 센서 그리고 상대적 위치 추정을 위한 휠 엔코더 등으로 구성된다.

(1) 시각 센서부

이동로봇의 시각 센서로서 비전(vision) 시스템과 CCD카메라를 이용한다. 이 센서 시스템은 넓은 범위의 주위 환경을 인식하고, 미리 지정된 지점에 위치한 랜드마크(landmark)의 영상처리를 통해 주행중인 이동로봇의 절대 위치정보를 얻을 수 있다. 시각 센서부는 랜드마크 인식용 CCD카메라 1대, 3차원적인 주위환경인식을 목적으로 하는 스테레오(stereo) 비전용 CCD카메라 2대 등 총 3대의 카메라와 이동로봇의 상단부에 위치한 카메라 구동부(mount) 그리고 관련된 제어장치 및 영상처리부 등으로 이루어진다. 카메라 구동부는 카메라의 시각 영역을 넓히기 위한 카메라 구동장치로서 2대의 직류 서어보 모터에 의해 제어되고 패닝(panning)과 틸팅(tilting)의 2개의 자유도를 갖는다. CCD카메라는 768(H)×493(H)의 분해능과 16mm의 초점거리를 갖는 모델(SONY XC-6)로 선정하였고, 영상처리부에서는 512×512의 분해능과 256 gray-level을 갖는 PC 인터페이스용 이미지그래버(모델: ITEX-100)가 사용되었다.

(2) 초음파 센서부

초음파센서는 근접 장애물에 대한 거리정보를 제

공하는데 효과적이다. 이동로봇 LCAR는 단면이 6각형인 몸체를 갖고 있으며, Fig. 3과 같이 그 몸체면에 총 26개의 초음파 센서(ultra sonic sensor)를 둘레방향으로 일정한 간격(상단 18개, 하단 8개)으로 배치하였다. 이 센서들은 주행중에 진행방향을 중심으로 초음파 감지 영역 내에 존재하는 장애물들을 감지하는데 사용한다. 각각의 초음파 센서들은 초음파센서 처리 보드(CPU:8032)에 의해 제어되며, 이 보드는 초음파를 발생시키는 송신부(transmitter), 반향(echo) 신호를 받기 위한 수신부(receiver), 게인(gain) 제어부 그리고 센서들을 스캐닝하는 스위칭 모듈로 구성된다. 초음파 센서 처리 보드를 통해 얻은 정보들은 통신 인터페이스부를 통해 주 제어 컴퓨터로 보내지고, 주 제어 컴퓨터는 이를 이용하여 충돌 회피 동작을 수행한다.

3. 이동로봇 제어 시스템 설계

이동로봇의 제어 시스템의 소프트웨어적 구성은 다음과 같은 점을 고려하여 이루어진다.

- 현재 어디에 어떤 자세로 위치하고 있는가?
- 어디로 가야하는가?
- 목적지에 어떻게 가야하고 장애물과의 충돌을 피할 수 있는가?

이러한 질문들은 이동로봇 제어기의 구조적 설계를 위한 중요한 동기이다. 첫번째 질문은 이동로봇이 가능한 많은 센서 정보들을 종합 처리(sensor fusion)하여 로봇의 현재 주행 상태(위치, 방향, 속도 등)에 관한 정확한 정보를 얻어야 함을 의미한다. 이러한 정보의 중요성 때문에 이동로봇의 제어 시스템을 구성하는데 있어 자기 위치 추정부의 구성과 함께 그 처리 알고리즘이 중요한 모듈이 된다. 그리고 두번째와 세번째 질문들은 각각 경로 계획기(path planner)와 경로 추종 제어기(path tracking controller)의 필요성을 내포한다.

3.1 위치 추정시스템

이동로봇의 제어 시스템을 구성함에 있어서, 이동로봇의 정확한 위치 결정은 로봇이 원하는 경로를 따라 정확히 주행하는데 필수적이다. LCAR의 위치 센서들은 상대 센서로서, 구동 휠축에 설치되어 휠의 회전량을 측정하는 광학식 엔코더가 있고, 절대 센서로서는 특정 위치에 설치된 랜드마크

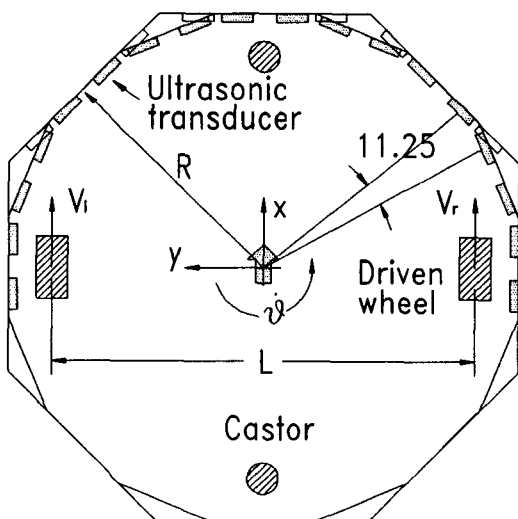


Fig. 3 Arrangement of ultrasonic sensors

를 인식하는 시각센서 등이 있다. 로봇이 2차원 평면상에서 주행한다고 가정하여, 이 평면상에서 고정된 좌표계(또는 절대좌표계)를 정의한다. 절대좌표계에 대해 이동로봇의 위치는 로봇의 회전중심점(C_R)의 (x, y) 좌표값으로 표현되고, 방향은 로봇의 진행방향의 축과 절대 좌표계의 X 축 사이의 각(heading angle)인 θ 로 표현한다. 이를 벡터로 표현한 것을 자세(posture)라 하며 이를 P 로 정의한다.

$$P = (x, y, \theta)^T \quad (1)$$

(1) Dead reckoning에 의한 위치추정

휠에 부착된 엔코더 또는 타코미터의 정보를 이용하여 이동로봇의 자세(posture)를 실시간으로 추정하는 dead reckoning 방법을 위치추정방법으로 이용한다. 로봇의 움직임을 곡선으로 표현할 때, 매 제어 주기마다 곡선의 곡률(x)과 진행거리의 증분(dS)을 아래의 식과 같이 좌우 구동휠의 회전량($d\theta_R, d\theta_L$)을 사용하여 계산할 수 있다.

$$dS = r_w(d\theta_R + d\theta_L)/2 \quad (2a)$$

$$x = \frac{2}{d_w} \frac{(d\theta_R - d\theta_L)}{(d\theta_R + d\theta_L)} \quad (2b)$$

여기서 r_w 는 휠의 반경이고, d_w 는 휠간의 거리이다. 일단 곡률(x)과 진행거리의 증분(dS)이 주어지면, 절대좌표계(world coordinate)에서의 로봇의 위치 및 방향의 변화량을 계산할 수 있다. 즉 샘플링 구간에서 로봇은 r 의 회전 반경을 갖고 $d\theta$ 의 회전 각도만큼 원호 운동을 한다고 가정한다. 반경(r)과 회전 각도($d\theta$)는 아래의 관계식으로 부터 구해질 수 있다.

$$d\theta = xdS \quad (3a)$$

$$r = 1/x \quad (3b)$$

따라서 원호 계산식과 좌표 변환을 이용하여, 절대좌표계에서의 자세변화 ($dx, dy, d\theta$)는 아래의 최종식으로 부터 구해진다.

$$dx = dS[\sin(\theta + d\theta) - \sin(\theta)]/d\theta \quad (4a)$$

$$dy = -dS[\cos(\theta + d\theta) - \cos(\theta)]/d\theta \quad (4b)$$

$$d\theta = xdS \quad (4c)$$

여기서 $d\theta \neq 0$ 일때, 즉 로봇이 원호운동을 한다고 가정한다. 또한 작은 간격의 샘플링 구간을 갖거나 회전각도 $d\theta$ 가 0에 가까울 경우 직선 운동으로 가정하여 구할 수 있다.

$$dx = dS \cos(\theta) \quad (5a)$$

$$dy = dS \sin(\theta) \quad (5b)$$

$$d\theta = 0 \quad (5c)$$

이러한 상대센서에 의한 위치 추정 방법의 정확도는 경로의 곡률이 클수록 그리고 로봇의 주행 속도가 빠를 수록 떨어지게 되고 근사적인 적분방법을 사용하므로 반복 주기를 짧게 할 수록 높아진다. 이 방법은 비교적 장치적으로 간단하고 빠른 샘플링 주기로 로봇의 위치 정보를 얻을 수 있는 장점을 갖는 반면, 로봇몸체의 실제 운동 정보가 아닌 휠의 운동정보에 의존하므로 휠의 변형, 바닥면의 굴곡, 바닥면과 휠간의 미끄러짐이 있는 경우 큰 오차가 발생한다. 또한 거리에 따른 오차의 누적특성 때문에 장거리 주행에는 적합하지 못하다. 반면 적당한 구간마다 랜드마크를 설치하고, 카메라와 같은 시각센서를 사용하여 로봇의 절대위치를 측정하는 방법은 오차의 누적특성이 없고 환경의 불확실성에 대처할 수 있으나, 장치비가 비싸고 감지범위가 제한되는 단점을 갖는다. 따라서 이 두 가지 방법은 상호 보완적이어서 랜드마크범위외의 지역에서는 dead reckoning 방법, 마크범위내의 지역에서는 시각센서 방법을 이용하는 복합센서방식이 유용하다.

(2) 시각센서를 이용한 절대 위치 추정

이는 특정한 형상의 랜드마크(landmark)들을 경로 부근의 지정된 위치에 설치하고, 카메라를 통해 마크의 영상을 얻어, 영상특징의 변화로부터 로봇의 절대 위치를 추정하는 방법이다. 여기서 문제는 카메라와 마크간에 위치적인 관계를 찾는 것이다. 지금까지 어떠한 마크의 형태가 로봇의 정확한 절대위치를 얻기에 적합한 것인가에 관한 많은 연구가 있어 왔다.⁽⁶⁻⁸⁾ 이러한 연구중에는 Kabuka와 Arenas⁽⁶⁾ 등의 원형의 마크를 이용한 방법이 있는데, 이는 마크에 대한 로봇의 상대적 위치와 방향의 변화는 마크의 영상 평면상의 타원의 속성과 관계가 있다는 원리를 바탕으로 한다. 이 연구에서는 추정 알고리즘의 실시간 처리를 위해 하드웨어적으로 연산처리하는 방법을 제시하였다. Tsmura 등⁽⁷⁾은 사각형태의 랜드마크를 사용하고 줌(zoom) 렌즈를 사용하여 영상의 민감도(sensitivity) 문제를 해결하려는 시도를 하였고, 마크상의 영상 특징점들을 다수 선택하여, 최소 자승법(least square method)을 적용함으로써 위치추정 오차를 최소화

하는 방법을 보여준 바 있다. 그러나 이러한 방법들은 아직까지 일반적인 제어 샘플링시간이 msec 수준인 것에 비해 많은 계산량을 요구하며, 실시간 처리 측면에서 문제점을 지닌다. 최근 J. H. Kim 등은⁽⁸⁾ 줄 무늬 패턴의 랜드마크를 이용하고, 라인 스캐닝(line scanning) 카메라를 통해 마크의 영상 데이터를 얻는 간단한 위치추정 방법을 개발하였는데, 이는 영상데이터를 얻는데 걸리는 시간을 최소화하고, 수학적 위치 관계 모델을 확정형(explicit form)으로 구하였다. 통상 로봇의 위치 정보는 이동 로봇의 주행중 필요로 하므로, 정확한 위치정보를 얻을 수 있는 방법을 있어도 이를 실시간으로 처리하지 못한다면 실현성이 떨어진다. 최근의 연구에서⁽⁹⁾ 이러한 문제에 초점을 맞추어, 좀더 실시간으로 처리할 수 있고 복잡한 수학적 모델을 필요로 하지 않는 절대 위치 추정 방법을 제시한 바 있다. 보통 우리 인간이 어떤 사물을 보고 그 사물에 대한 자신의 상대적인 위치와 방향을 가늠할 때, 기억속에 있는 그 사물의 원래 형상과 크기에 대한 정보와 눈에 비친 그 사물의 영상정보를 비교한다. 이 때 복잡한 사물의 경우, 모서리간의 거리, 윤곽선간의 각도등과 같은 특징치등을 추출하여, 이미 인간의 두뇌속 학습된 감각에 의해 상대적인 위치를 추정한다고 보여진다. 이러한 직관적인 개념을 바탕으로한 절대 위치추정 방법을 제시한다. 즉 사각형의 마크를 랜드마크로 이용하고, 이 마크의 영상과 로봇의 위치의 상관관계를 신경망모델을 통하여 구현한다. 이 때 사용되는 영상 정보로는 마크의 모양의 변화나 크기의 변화를 대변하는 특징점들을 이용한다.

· 마크의 설계 및 영상특징

Fig. 4에 보인바와 같이, 흰색바탕에 폭 40 mm의 검은색의 띠로 이루어진 사각형의 마크가 랜드마크로서 사용된다. 그리고 마크 좌표계의 원점은 마크의 검은색 띠의 내면의 사각형이 이루는 4개의 꼭지점 A, B, C, D의 중심점 O_m 로 한다. 이 때 이 동로봇은 평면에서 움직이고, 카메라는 평면위에 Pan운동만 한다고 가정한다. 마크의 내면 사각형의 4개의 꼭지점들을 이진(binary) 영상처리를 통해 구한다. 이 4점들은 모두 같은 평면상에 존재하며, 영상좌표계에서의 이 특징점들의 좌표들은 각각 (x_{i1}, y_{i1}) , (x_{i2}, y_{i2}) , (x_{i3}, y_{i3}) , (x_{i4}, y_{i4}) 라 할 때, 이들의 무게중심 P_s 의 좌표 (x_{i5}, y_{i5}) 를 다음과 같이 계산한다.

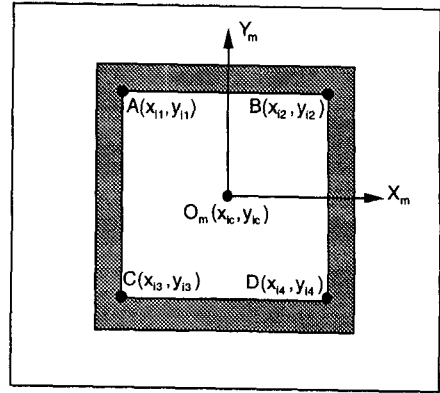


Fig. 4 Image features of the landmark used for the image processing

$$x_{i5} = (x_{i1} + x_{i2} + x_{i3} + x_{i4}) / 4 \tag{6a}$$

$$y_{i5} = (y_{i1} + y_{i2} + y_{i3} + y_{i4}) / 4 \tag{6b}$$

이 5개의 점들로 이루어진 영상특징벡터 S 를 다음과 같이 정의 한다.

$$S = [(x_{i1}, y_{p11}), \dots, (x_{i5}, y_{i5})] \tag{7}$$

· 신경망모델의 도입

마크형상변화와 로봇의 위치관계사이의 비선형적 관계를 Fig. 5와 같이 다층형 피이드포워드 신경망(multi-layered feedforward neural network) 모델로 구현한다. 신경망의 연결 강도는 마크 주변에서

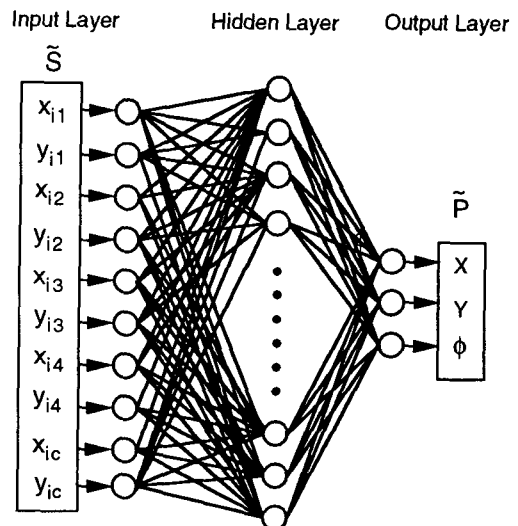


Fig. 5 Neural net model for absolute position estimation

미리 지정된 여러 지점들로부터 구한 학습 패턴에 대해, 반복 학습을 통해 결정된다. 다층형 신경망 모델은 잘 알려진 바와 같이 해석하기 어렵고, 불확실성이 내재된 비선형 함수관계를 구현하는 데 효과적이고, 주어진 대상에 대한 수학적 모델링이 필요없는 장점을 지닌다. 이러한 다층형 순방향 신경망을 마크에 대한 로봇의 위치를 결정하는 계산적 모델(computational model)로 사용한다.

$$\bar{P} = NN(S) \quad (8)$$

여기서 신경망의 입력 S 는 각 마크마다 정의한 영상특징벡터이며, 출력은 마크에 상대적인 로봇의 위치벡터 $P = [X, Y, \phi]^T$ 이고 $NN()$ 은 신경망에 의해 수행되는 함수를 의미한다. 이 신경망의 학습 방법으로 오차 역 전파 방법(error back propagation method)⁽¹⁰⁾을 사용한다. 이 학습방법은 학습 오차의 제곱의 합(sum of square error)으로 에너지함수(energy function) E 를 정의하고, 이를 최소화 하기 위한 점진적 감쇄법칙(steepest gradient decent rule)에 근거한 법칙을 사용한다.

$$E = \sum_{k=1}^n (\bar{P}_t^k - \bar{P}^k) (\bar{P}_t^k - \bar{P}^k)^T \quad (9)$$

여기서 \bar{P}_t^k 는 k 번째 학습데이터의 목표위치벡터(target position vector)이고 \bar{P}^k 는 k 번째 학습입력에 대한 신경망 계산결과인 실제위치벡터(actual position vector)이다. 이러한 학습을 통해 신경망 모델에서 충분히 입력 패턴과 출력 패턴의 사이의 관계가 형성되면, 신경망모델은 위치추정 알고리즘에서 이동로봇의 위치와 마크영상의 특징점사이의 관계 모델로서의 기능을 하게 된다. 제안된 방법에서는 마크에 대한 기하학적 해석이나, 정확한 카메라 모델링없이 단지 선택된 학습샘플만 사용하므로, 위치·영상 관계모델을 구하는데 수반되는 문제점

을 해결한다. 학습데이터들은 다음과 같은 방법으로 구한다. 마크전방의 우측 공간을 X_m 축 방향으로 40 cm 간격으로 8분할하고, Z_m 축을 기준으로 12°씩 회전 4분할하여 32개의 교차점을 구한다. 각 지점에 이동로봇을 위치시킨후 그 위치를 목표위치(X_t, Y_t)로 하고, 다시 각 위치에서 카메라의 방향(ϕ_t)을 1~5° 간격으로 변화시킨다. 따라서 총 $8 \times 4 \times 5 = 160$ 개의 학습데이터를 얻을 수 있었다. 그리고 각각의 위치 및 방향에서 실제로 얻은 마크영상의 특징벡터(\bar{S}^k)를 구하여 학습패턴(\bar{T}^k)을 구성하였다.

$$\bar{T}_t^k = (\bar{S}_t^k, \bar{P}_t^k), \text{ 여기서 } k=1, \dots, 160. \quad (10)$$

Fig. 6은 이 방법을 적용하여 절대 위치 추정 실험을 한 결과이다. 이 실험에서는 마크에 대해 수직인 직선 경로를 마크 중심부로부터 30 cm 간격으로 6개 선정하여 시험경로로 하였고, 각 경로상에서 10 cm간격으로 로봇을 정지시켜 절대위치를 추정하였다. 그림은 로봇이 마크로부터 멀리 떨어진 지점에 위치할 수록 위치 추정오차가 커짐을 보여 준다. 이는 거리가 멀수록 마크의 영상 크기가 작아져서 센서의 민감도가 떨어지기 때문이다. 제안된 방법은 향후 신경망계산기의 출현을 기대해 볼 때 병렬처리에 의한 알고리즘의 실시간 처리에 그 특징이 있다고 본다.

3.2 경로 계획

이동 로봇을 위한 경로 계획은 전체 주위 환경에 대해 수행되는 전체 경로 계획(global path planning)과 각 경로를 구간화 하여 수학적으로 표현하는 국부 경로 계획(local path planning)으로 나뉘어진다. 전체 경로 계획은 출발점과 최종 목적지가 결정되면, 주위 환경(예 : 복도, 문, 시설물 등)에 관한 사전 정보를 미리 이용하여 수행된다. 반면 국부 경로 계획은 실제 이동로봇의 동적 특성과 기구학적 한계를 고려하여 이동 로봇이 추종 가능하도록 각 경로의 구간을 기하학적인 수학적식으로 표현하는 것이다. 이때 이동 로봇의 부드러운 가감속이나 유연한 회전은 휠의 미끄러짐과 운동중 과도한 충격을 피하는 데 중요하다. 이전의 연구에서 경로계획시 부드러운 경로를 만들기 위해 곡률 연속 곡선(예 : clothoid, spiral curve 등)을 이용한 경로설계 방법들이^(11,12) 발표된 바 있다. 그러나 이러한 곡률연속 곡선을 사용한 경로의 계획방법은 곡선식의 확정형의 해가 존재하지 않거나, 복잡한 계산을 필요로하여, 실시간 국부 경로 계획이 곤란

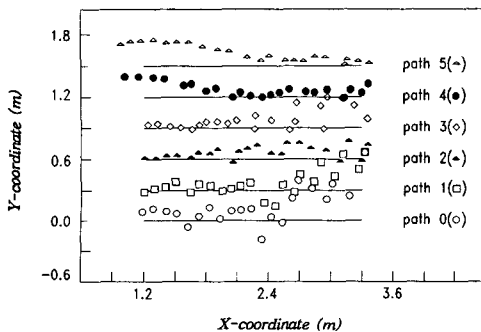


Fig. 6 Position estimation results for the stationary robot

한 단점이 있다. 따라서 본 연구에서는 경로를 단순히 직선경로의 구간(segment)의 조합으로 설계하고, 인접한 세그먼트간에 어떤 제한 조건(위치, 방향, 곡률 등의 연속조건)이 필요하지 않는 간단한 국부 경로계획 방법을 사용한다. 그리고 경로구간의 사이를 부드럽게 연결하는 역할을 경로제어기가 담당하도록 하는 방법을 채택한다. 따라서 국부 경로계획시 충돌회피 동작과 같은 실시간 경로 변경을 할 때, 경로설계에 대한 계산 부담을 덜어줌으로써 보다 효율적인 제어 시스템을 구성할 수 있다.

3.3 경로 추종 제어

이동로봇이 주행시 경로를 벗어나지 않도록 선속도명령(v_d) 및 회전속도 명령(ω_d)을 매 제어 사이클마다 결정하는 것을 경로제어라 한다. 본 연구에서 채택한 경로제어 제어법⁽¹³⁾을 간단히 소개한다.

(1) 최소시간 제어기 (bang-bang control)

먼저 제안된 경로제어 방법의 기본이 되는 최소시간 bang-bang제어에 대해 알아 본다. 이를 위해 직선 운동만을 하는 가상의 이동체를 고려하고, 그 속도(v_c) 및 가속도(a_c)의 최대값이 아래와 같이 제한된다고 가정한다.

$$|v| \leq v_{\max} \quad (11a)$$

$$|a| \leq a_{\max} \quad (11b)$$

여기서 v_{\max} 는 최대 허용 속도이고 a_{\max} 는 최대 허용 가속도이다. 그리고 초기에 이 이동체의 경로상에서 속도 v_r 로 움직이는 가상의 목표점이 주어질 때, 이동로봇이 식(11a,b)의 제한조건을 만족하면서 최소 시간내에 현재의 위치 및 속도(x_c, v_c)로부터 목표점의 위치 및 속도(x_r, v_r)에 도달하기 위한 로봇의 가속도 a_c 를 구하는 문제를 고려한다. 먼저 이동체의 목표점의 위치 및 속도(x_r, v_r)에 대해, 현재 로봇의 위치 x_c 에서 식(11b)의 최대 가속도로 감속했을 때 목표점에 도달할 수 있는 속도를 v_s' 라 할 때 아래의 식으로 계산되어진다.

$$v_s' = v_r + [2a_{\max}|x_r - x_c|]^{1/2} \operatorname{sgn}(x_r - x_c) \quad (12)$$

그리고 최대 허용 속도조건 식(11a)에 의해

$$v_s = \mu(v_s', v_{\max}) \quad (13)$$

V_s' 의 크기를 제한한다. 여기서 $\operatorname{sgn}(\cdot)$ 은 부호함수이고, $\mu(a, b)$ 는 a 의 절대값이 b 로 제한되는 clamping 함수이다. 이렇게 v_s 가 구해지면, 샘플

링 주기를 T_s 라 할 때, 가속도(a_c)는 다음과 같이 결정할 수 있다.

$$a_c = \mu[(v_s - v_c)/T_s, a_{\max}] \quad (14)$$

이러한 1차원 운동에 대한 운동체의 제어 방법은 식(11b)의 제한조건을 만족하며 최소시간 내에 목표점에 도달할 수 있다. 이때 식(11b)의 제한조건이 그 이동체가 얼마나 부드러운 가감속 운동을 하는 가를 결정하게 된다. 이를 직진과 회전의 2자유도를 갖는 이동로봇의 경로 제어기에 확대 적용한다.

(2) 선 속도 제어 알고리즘

이동로봇의 선 속도 제어는 앞절의 최소 시간 제어를 사용하여 구현된다. 기준 경로상에서의 목표점의 자세를 P_r 로 정의하고, 이동로봇의 현재의 자세를 P_c 로 정의할 때, 기준 경로의 자세(P_r)를 기준으로 한 로봇의 자세오차(P_{er})를 다음 식으로 구할 수 있다.

$$P_{er} = (x_{er}, y_{er}, \theta_{er})^T = T_{er}(P_r - P_c) \quad (15)$$

$$\text{여기서 } T_{er} = \begin{bmatrix} \cos(\theta_r) & \cos(\theta_r) & 0 \\ -\sin(\theta_r) & \cos(\theta_r) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

이다. 그러면 x_{er} 은 목표점과 로봇간의 진행 방향에 대한 거리가 된다. 그리고 목표점의 기준속도가 v_r 로 주어질 때, 목표점까지의 거리(x_{er})에 대한 최소시간 제어방법을 그대로 적용하여 유도한 선속도 제어 알고리즘을 정리하면 다음과 같다.

$$v_s' = v_r + [2a_{\max}|x_{er}|]^{1/2} \operatorname{sgn}(x_{er}) \quad (16a)$$

$$v_s = \mu(v_s', a_{\max}) \quad (16b)$$

$$a_c = \mu[(v_s - \omega_c)/T_s, a_{\max}] \quad (16c)$$

$$v_d = v_c + a_c T_s \quad (16d)$$

이때 만약 기준 경로가 n 개의 직선 구간으로 설계되고 각 구간마다 서로 다른 기준 속도($v_i, i=1, 2, \dots, n$)를 설정하더라도 식(16a)에서 식(16b)의 선속도 제어알고리즘에 의해, 실제 주행시 이동로봇은 식(11a)와 (11b)의 제한조건들을 만족하며 부드러운 가감속 운동을 얻을 수 있다.

(3) 회전 속도 제어 알고리즘

이번에는 이동로봇이 주어진 경로에서 벗어나, 경로에 대해 θ_{er} 의 방향오차를 가지고 경로의 수직 방향으로 y_{er} 만큼 떨어진 거리에 위치할 때, 주어진 경로에 복귀하기 위한 이동로봇의 회전속도 ω_d 를 구하는 문제를 생각한다. 이 때 단지 회전속도

의 제어만으로 방향 오차(θ_{er})와 거리 오차(y_{er})를 모두 줄여야 하므로 로봇의 현 지점에서 취해야 할 기준방향은 이 두 오차들을 모두 고려해 결정해야 하는 것이 앞서의 선 속도 제어와 다른 점이다. 이 문제를 해결 위해, 보통 국부 경로 계획에서 사용하는 곡률 연속 곡선을 이용하는 방법을 제시한다. 먼저 기준경로상에 경로진행 방향을 x 축으로 하는 X - Y 좌표계를 설정한다. 그리고 이 좌표계의 원점은 경로상의 임의의 점에 놓고 경로를 기준하여 좌우에서 접하는 x 에 대한 3차 커브식(cubic polynomials, $y_L(x)$)을 아래의 조건들을 만족하도록 구한다.

$$(i) y_L(x)|_{x=0}=0 \quad (17a)$$

$$(ii) dy_L(x)|_{x=0}=0 \quad (17b)$$

$$(iii) \frac{d^2 Y_L(x)}{dx^2} \Big|_{x=0} = 0 \quad (17c)$$

$$(iv) \frac{d^3 Y_L(x)}{dx^3} \Big|_{x=0} = c_x \quad (17d)$$

$$(v) |x(x)| \leq x_{max} \quad (17e)$$

여기서 $x(x)$ 는 곡률 함수로 X - Y 좌표상의 커브의 곡률은 아래의 관계식을 갖는다.

$$x(x) = \frac{d^2 y / dx^2}{[1 + (dy/dx)^2]^{3/2}} \quad (18)$$

여기서 (i)~(iii)은 곡선이 곡률의 연속성을 만족하면서 주어진 경로에 접하는 조건이고, (vi)~(v)는 곡선의 c_x 를 정하기 위한 조건으로 로봇의 최대 허용 곡률을 제한한다. 따라서 위의 조건들을 만족하는 식은

$$y_L(x) = c_x x^3 \quad (19)$$

으로 표현된다. 이 곡선은 현재의 로봇지점에서 부드럽게 주어진 직선 경로에 수렴할 수 있는 중간경로가 되며, 이를 랜딩커브라 정의한다. 이동로봇이 경로의 수직 방향으로 임의의 거리(y_{er})에 위치할 때, 로봇의 중심을 통과하는 랜딩커브를 구할 수 있으며, 이 점에서의 랜딩커브의 접선과 경로의 접선사이의 각이 로봇이 회전해야 할 목표 회전각이 된다. 이를 랜딩각도(θ_L)라 정의하며 아래의 식으로 표현된다.

$$\theta_L = \tan^{-1}(S_L) \quad (20)$$

$$S_L = c_x \left(\frac{y_{er}}{c_x} \right)^{2/3} \text{sgn}(y_{er}) \quad (21)$$

이 랜딩각도는 이동로봇이 선속도 v_c 의 속도로 움

직이고 있을 때 시간에 따라 변하게 된다. 따라서 랜딩각도의 변화율을 식(21)를 시간 t 에 대해 미분하면,

$$\omega_L = d\theta_L / dt = \frac{dS_L / dt}{(1 + S_L^2)} \quad (22)$$

$$dS_L / dt = 2 \left(|y_{er} / c_x \right)^{-1/3} y_{er} \text{sgn}(y_{er}) \quad (23)$$

여기서 y_{er} 는 이동로봇이 기준경로에 수직방향으로 접근하는 속도로서 아래식으로부터 구할 수 있다.

$$y_{er} = v_c \sin(\theta_{er}) - \omega_L x_{er} \quad (24)$$

지금까지 구한 랜딩각 및 그 변화율로부터 이동로봇이 취해야할 기준 방향각도(θ_{rs})와 그 기준 각속도(ω_{rs})를 아래와 같이 구할 수 있다.

$$\theta_{rs} = \theta_L + \theta_r \quad (25a)$$

$$\omega_{rs} = \omega_L + \omega_r \quad (25b)$$

따라서 기준 방향각(θ_{rs})과 기준 각속도(ω_{rs})가 구해지면, 최소시간 제어법을 적용할 수 있다. 이때 최대 허용 각속도(ω_{max})와 최대 허용 각가속도(α_{max})의 조건은 로봇의 동적 한계를 고려하여 결정하며, 최종적인 회전속도 제어알고리즘의 형태는 다음과 같다.

$$\omega_s' = \omega_{rs} + [2\alpha_{max}|\theta_{rs} - \theta_c|]^{1/2} \text{sgn}(\theta_{rs} - \theta_c) \quad (26a)$$

$$\omega_s = \mu(\omega_s', \omega_{max}) \quad (26b)$$

$$\alpha_c = \mu[(\omega_s - \omega_c) / T_s, \alpha_{max}] \quad (26c)$$

$$\omega_d = \omega_c + \alpha_c T_s \quad (26d)$$

여기서 ω_d 는 다음 제어샘플 순간에서의 이동로봇이 취해야 할 회전속도명령으로 휠서어보 제어기의 기준입력이 된다.

3.4 경로제어 실험

경로제어방법을 이용하여 이동로봇의 제어시스템을 Fig. 7과 같이 구성하였다.

(i) 이동로봇의 위치 및 속도 추정

일단 시각센서 영역밖의 경우 휠센서정보를 가지고 dead reckoning을 통해 이동로봇의 현재의 자세(x_c, y_c, θ_c) 및 속도(v_c, ω_c)를 계산하고, 시각영역내에서는 마크의 영상정보를 이용하여, 로봇의 자세를 추정한다. 이때 속도는 휠엔코더의 정보로부터 구한다. 이때 dead reckoning에서는 식(4)와 식(5)를 이용한다.

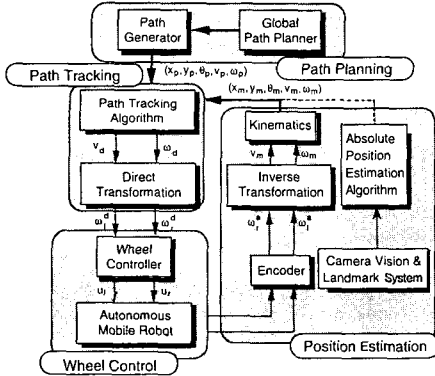


Fig. 7 S/W block diagram of mobile robot control system

(ii) 경로오차의 계산

식(15)을 이용하여 기준경로에 기준한 상대오차(P_{er})를 계산한다.

(iii) 경로 추종제어

앞절에서 설명한 바와 같이, 일단 상대 위치 오차(P_{er}) 및 기준 경로에서의 속도(v_r, ω_r)가 주어지면, 식(16a~d)과 식(26a~d)의 경로 추종제어 알고리즘을 수행하여 다음 제어샘플순간에서 이동로봇이 취해야 할 속도(v_d, ω_d)를 계산한다.

(iv) 로봇 기준 좌표계의 변환

앞의 단계 (iii)에서 구한 경로제어기의 출력 즉 v_d 및 ω_d 를 양 휠 구동형(two-wheel drive) 방식의 이동로봇에 대한 역 기구학적 해석을 통해 각 구동휠에 대한 기준속도로 변환한다. 이 변환식을 구하면 다음과 같다.

$$\omega_w^d = [v_d + d_w \omega_d / 2] / r_w \quad (27a)$$

$$\omega_l^d = [v_d - d_w \omega_d / 2] / r_w \quad (27b)$$

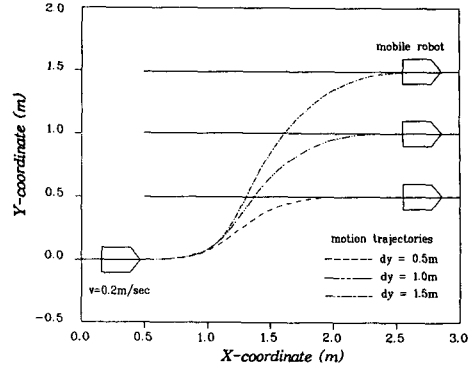
(v) 펄스폭치의 계산

각 구동모터에 대한 기준속도는 샘플링 주기 T 를 고려하여 다음의 계산식을 통해 휠의 펄스수(p^d, p^l)로 변환된다.

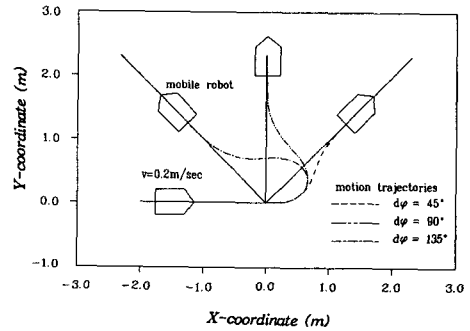
$$p^d = 2000 \omega_w^d T / 2\pi \quad (28)$$

여기서 $i=R, L$ 이다. 이 값들은 하부의 휠 제어 보드로 전달되어, 위치 제어루프의 기준입력이 된다. 이러한 (i)에서 (v)의 5단계는 매 제어 샘플마다 반복된다.

Fig. 8(a)~(b)는 DR위치 추정방법을 이용한 경로 추종제어 실험결과를 보여준다. 이 때 경로 추



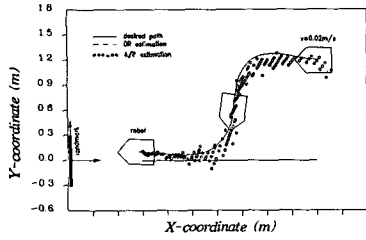
(a) Lane change motions



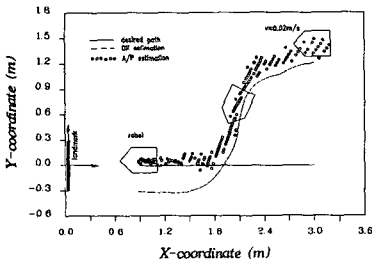
(b) Instant turning motions

Fig. 8 Experimental results of path tracking control

종 제어기에서는 앞절에서 제시한 최소시간 bang-bang 제어기와 랜딩 커브를 기초로 한 경로 제어법이 사용되었다. Fig. 8(a)는 이동로봇이 직선 경로를 따라 주행하다가 횡방향으로 경로를 급격히 변경했을 때, 그 경로변경폭의 변화($dy=0.5m, 1.0m, 1.5m$)에 따른 로봇의 운동체적들을 도시한 결과이다. 그림에서 알 수 있듯이 경로상에 위치의 불연속이 존재하여도 제안된 경로제어기에 의해 로봇은 주어진 동적 구속조건을 만족하며, 두 불연속적인 경로사이를 부드럽게 연결하는 운동체적을 보여준다. Fig. 8(b)는 경로의 각도가 순간적으로 바뀌었을 때, 그 각도 변경의 크기($d\phi=45^\circ, 90^\circ, 135^\circ$)에 따른 경로제어의 결과이다. 이 실험결과는 경로가 방향의 불연속을 갖더라도, 로봇은 주행중에 그 불연속 구간을 부드럽게 운동 궤적으로 연결하며 경로를 추종할 수 있음을 보여준다. 마지막으로 Fig. 9(a)~(b)는 시각센서를 이용한 절대위치 위치추정방법과 이 절에서 제안된 경로제어법을 사용하여 경로제어를 수행한 결과들이다. 이 실험에



(a) Case of normal condition (no slip & no bumper)



(b) Case of crossing over a bumper ($h_p=15$ mm)

Fig. 9 Experimental results of path tracking control based on absolute position(A/P) estimation

서는 로봇이 마크를 기준으로 $X=3.0$ m, $Y=1.2$ m인 지점에서 출발하여 마크를 감지하며 절대위치를 추정하고 랜딩커브에 기초한 경로제어법을 사용하여 마크의 X_m 축을 기준경로하여 $X=0.9$ m, $Y=0$ m인 지점에 도달하도록 하였다. Fig. 9(a)는 평평한 바닥과 미끄럼이 없는 잘 정돈된 환경 즉 정상조건에서의 주행 실험결과이고 Fig. 9(a)의 경우 바닥의 굴곡이나 미끄러짐과 같은 불확실한 환경에서의 주행 실험 결과이다. Fig. 9(a)와 같이 정상조건에서는 마크의 영상정보를 이용하는 위치추정방법이 최종지점에서의 y_m 축 방향으로 10 cm내외의 위치오차를 보였다. 그리고 절대 추정방식과 DR방법을 비교해보면, 짧은 거리의 주행이나 미끄러짐이 없는 평평한 바닥조건에서는 DR방법이 더 안정되고 정확한 위치추정결과를 보여준다. Fig. 9(b)와 같이 바닥의 굴곡이나 미끄러짐이 발생하는 환경에서는 DR방법은 오차가 누적되어 비교적 짧은 거리의 주행(주행거리 3.0 m 이하)에서도 최종지점기준으로 30 cm 정도의 오차를 보여준다. 반면에 절대추정법은 그 위치오차의 크기가 정상상태의 경우와 크게 변함없는 10 cm 내에 있음을 확인할 수 있다.

4. 결론 및 향후 과제

본 논문을 통해 proto-type 이동로봇 LCAR에 대한 제어시스템의 하드웨어적 구조와 각 소프트웨어 모듈에 대한 설계배경을 기술하였다. 설계된 LCAR-90 로봇의 제어시스템의 특징으로는 PC를 중심으로 센서부 및 휠제어부등 각 하부 시스템들이 연결구성되고, 휠제어부및 초음파 센서부 그리고 카메라 구동부의 제어부를 각각의 마이크로 프로세서를 사용하여 독립적 모듈로 설계하므로써, 계층적 구조가 되도록 한 것이다. 또한 이동로봇의 제어시스템을 구성함에 있어 필수적인 요소들인 위치추정시스템, 경로계획기 그리고 경로추종 제어부를 실시간 처리에 초점을 두어 개발하였다. 여기서 적용된 각각의 알고리즘에 대한 설명을 하였고, 성능과 효과를 일련의 실험을 통해 알아 보았다. 또한 실험을 통해 설계된 제어시스템 안정성과 제안된 제어알고리즘의 실시간 수행 가능성을 확인할 수 있었다. 경로제어 실험결과들은 제안된 경로제어 방법이 경로의 불연속적인 변화에 대해서도 안정되고 부드러운 운동제어를 보장함을 보여준다. 또한 랜드마크 인식을 통한 자기위치 추정결과에 신경망을 기초로한 위치추정 알고리즘의 효용성과 적용 가능성을 보여준다. 추후 연구 과제는 지금까지 개발된 로봇 제어시스템의 기능에 실시간 경로계획과 충돌회피 알고리즘을 포함시켜 좀더 복잡하고 불확실한 환경내에서도 자율적으로 운행할 수 있는 지능적인 이동로봇 제어시스템을 개발하는 것이다.

참고문헌

- (1) Amidi, O., 1990, "Integrated Mobile Robot Control," Master Thesis, Dept. of Electrical and Computer Eng., CMU.
- (2) Tsumura, T., 1981, "An Experimental System for AGV, Following the Route Stored in Memory," Proc. of 11th Int. Symp. on Industrial robots.
- (3) Kanayama, Y. and Yuta, S. Y., 1988, "Vehicle Path Specification by a Sequence of Straight Lines," IEEE J. of Robotics and Automation, Vol. 4, No. 3, pp. 265~276.

- (4) Singh, S. and Shin, D. H., 1989, "Position Based Path Tracking for Wheeled Mobile Robots," Proc. of IEEE IROS 89, pp. 386~391.
- (5) Crowley, J., 1989, "Asynchronous Control of Orientation and Displacement in Robotic Vehicle," Proc. of IEEE conf. on Robotics and Automation, pp. 1277~1282.
- (6) Kabuka, M. R. and Arenas, A. E., 1987, "Position Verification of a Mobile Robot Using Standard Pattern," IEEE J. of Robotics and Automation, RA-3, No.6, pp. 505~516.
- (7) Tsumura, T., 1982, "A System for Measuring Current Position and/or Heading of Vehicle," Bull. JSME, Vol. 25, No. 203, 821~826.
- (8) Kim, J. H. and Cho H. S., 1992, "Real-time Determination of a Mobile Robot's Position by Linear Scanning of a Landmark," Robotica, Vol. 10, pp. 309~319.
- (9) Kim, J. S., Koh, K. C. and Cho, H. S., 1992, "A Neural Net-Based Position Estimation Method for Mobile Robots in Indoor Environments," Proc. Int'l Symp. on Meas. and Con. in Robotics, ISMCR 92, pp. 899~906.
- (10) Rumelhart, D. E., McClelland, J. L. and the Topresearch group, 1986, "Parallel distributed processing," Bradford Books, Vol. 1&2 MIT Press.
- (11) Kanayama, Y. and Hartman, B. I., 1989, "Smooth Local Path Planning for Autonomous Vehicles," IEEE Int. Conf. on R&A, pp. 1265~1270.
- (12) Kanayama, Y. Kimura, Y, Miyazaki and Noguchi T., 1990, "A Stable Tracking Control Method for an Autonomous Mobile Robot," Proc. IEEE Conf. on R&A, pp. 384~389.
- (13) Koh, K. C. and Cho, H. S., 1991, "A Steering Control Method for Wheel-driven Mobile Robot," KACC, pp. 787~792.