

Color Line 탐색을 이용한 AGV의 주행제어에 관한 연구

박영만* 박경우** 안동순***

A Study on the Navigation Control of Automated Guided Vehicle using Color Line Search

Young-man Park* Kyung-woo Park** Dong-soon Ahn***

요 약

유연 생산 시스템(FMS)이나 자동화 창고(AWS) 등에 이용되는 AGV(Automated Guided Vehicle)에 관하여 많은 연구가 진행 중이다. 기존의 AGV는 마그넷 테이프나 전기와이어, RF나 Laser를 가이드라인으로 사용하고 있어 가이드라인 설치와 변경 시 시간과 비용이 많이 드는 단점이 있다. 본 논문에서는 단일 컬러 CCD카메라를 가지고 50mm 황색 컬러테이프를 가이드라인으로 사용하여 주행하는 AGV를 구현하였다. 컬러테이프를 사용하므로 라인설치 및 변경, 추가 시 작업 시간이 빠르고 비용이 적게 드는 장점이 있다. 구현한 AGV의 구조와 컬러 특성만을 추출하여 주행용 가이드라인을 탐색하는 영상처리 기법과 AGV 주행 결과를 제시하였다.

Abstract

There are active researches on automated guided vehicles(AGV) generally used in flexible manufacturing system(FMS) or automated warehouse systems(AWS). Because existing AGV uses magnetic tapes, electric wire, RF or laser as guidelines, its installation and modification require a lot of money and time. The present study implemented AGV that detects paths marked with 50mm yellow tape using a mono-color CCD camera. Because it uses color tape, it is easy and inexpensive to install and change lines. This study presented the structure of the developed AGV, the image processing technique for detecting guidelines by abstracting the characteristics of color, and the result of operating AGV.

* 목포기능대학 컴퓨터정보과 부교수

** 목포대학교 공과대학 컴퓨터공학과 부교수

*** 목포대학교 공과대학 컴퓨터공학과 교수

있다.

1. 영상 처리부

영상처리에 사용할 컴퓨터는 펜티엄IV 1.7GHz와 512MB의 메모리, 512MB의 flash HDD를 장착하고 운영체제는 Windows98을 사용하였다.

I. 서론

산업현장에서 보다 효율적인 생산시스템을 구축하기 위하여 AGV가 사용되고 있으며 반도체 공정, 자동화창고(AWS), 유연생산시스템(FMS) 등에 널리 이용되고 있다.

AGV는 작업의 정확성을 위해 주어진 가이드를 충실히 따라가는 동작이 중요하다. 현재 많이 사용되는 AGV의 가이드 시스템은 마그넷 테이프, 전기와이어, RF나 Laser Beacon, 카메라를 통한 영상정보를 이용하는 방법 등이 있다.[1][2][3] 이러한 기존의 방법들은 각기 단점을 갖고 있다. 마그넷 테이프나 전기와이어는 가이드 라인의 설치와 가이드라인의 변경 시 비용이 많이 들며, RF나 Laser Beacon는 고가의 장비 가격과 설치 환경에 제약을 받는다. 영상정보를 이용하는 방법은 과거에는 영상처리 시간이 많이 소요되어 실시간제어가 어렵고 일정한 조명을 필요로 하는 단점을 지녔으나 현재는 컴퓨터 속도의 향상에 따라 실시간 제어가 가능하게 되었다.

본 논문에서는 단일 CCD카메라를 가지고 특정한 색상을 갖는 폭 50mm의 황색 비닐테이프를 가이드라인으로 사용하고, 불균일한 조명환경을 갖는 복도에서 색상정보만을 탐색하여 주행하는 AGV를 설계 및 제작하고 주행 결과를 제시 하였다.

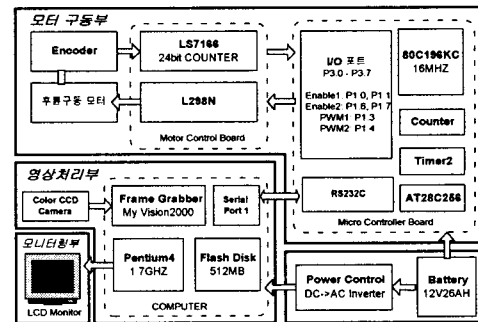


그림1. AGV 시스템의 전체 구성도
Fig. 1 Total structure diagram of AGV system

카메라는 Color CCD 단일카메라(SCC-835)와, frame grabber로 My Vision00이 사용되었다.

AGV 주행 경로로 사용되는 황색 라인테이프를 탐색하기 위하여 AGV 전면에 카메라를 50°로 경사시키고 700mm의 높이에 장착하였으며, 렌즈는 가변 줌렌즈(SVF-245, f:4.5mm ~ 10mm)에 외부의 불빛이나 형광등에 의한 반사광 영향을 줄이기 위해 편광필터(VF-30CPKS)를 장착하였다.

2. 모터 구동부

2.1 AGV Motor Control Board

AGV는 두개의 DC 모터로 후륜으로 구동되며 두개의 전륜은 보조바퀴로 사용된다. 80C196의 마이크로프로세서를 컨트롤러로 사용하며, 분해능 600~2400Puls/Rotation 성능을 갖는 Encoder와 24비트 고속 카운터, L298N 모터드라이버로 구성되어 있다.

DC 모터는 12V, 1.5A, 3000rpm으로 75:1 감속기를 사용하여 40rpm, 18.75kg.cm 토크를 출력하는 2개의 모터를 사용 하였다.

2.2 모터 구동회로 설계

마이크로컨트롤러로 80C196 보드를 사용하였으며

II. AGV 시스템의 구성

AGV 시스템은 영상 처리부, 모터 구동부, 모니터링부로 그림1과 같이 구성되었으며, 이외 AGV에 Power를 공급하기위하여 Battery로 구성된다. 영상처리에 사용되는 컴퓨터를 구동하기 위하여 DC 12V, 26AH Battery에서 DC/AC Inverter를 통하여 Power를 공급하고 CCD Camera와 DC Motor, Motor Control Board에 DC 12V를 Battery에서 직접 공급되는 구조로 되어

DC 모터 2개를 가·감속, 정·역회전 구동을 위한 모터 드라이버는 L298N을 각 모터 당 1개씩 병렬로 구성하여 사용하였다. L298N은 DC동작의 경우 2A 까지 사용할 수 있으므로 부하구동 시 3A 이상 까지 사용할 수 있도록 Enable, IN, OUT을 병렬로 구성하여 4A 까지 출력을 낼 수 있도록 하였다.[4]

Phase(IN1,IN2)에 logic 신호를 주어 정 회전, 역회전, 정지신호를 주고 Enable에 PWM 신호를 0~255 까지 인가하여 256단계의 속도를 제어하도록 구성하였다. Encoder를 사용하여 바퀴의 이동거리를 판단할 수 있도록 분해능 600P/R를 사용하며 이 Encoder에서 발생하는 펄스를 카운트하기위해 24bit 고속카운터인 LS7166 두개를 사용하였다. 또한 소프트웨어적으로 4 체배 하여 2400P/R으로 분해능을 높여 구동할 수 있다.

3. 모니터링부

AGV의 카메라에서 생성되는 영상을 디스플레이하고 주행제어를 위한 환경 설정 및 주행과 제어 동작 상태를 파악하기위해 15" LCD 모니터와 무선 키보드를 사용 하였다. 모니터에 그림2와 같이 출력되며 그 내용은 다음과 같다.

- (1) Color Line 색상 설정
- (2) 탐색 대상 물체의 최소, 최대 픽셀 크기 설정
- (3) AGV 주행라인 탐색과 주행 목표선 출력
- (4) 주행라인과의 각도 계산 및 좌우 바퀴 속도 출력
- (5) 컴퓨터의 Vision Program과 모터 구동 보드인 80C196 보드와의 시리얼 송·수신 상태
- (6) 영상처리 시간 출력

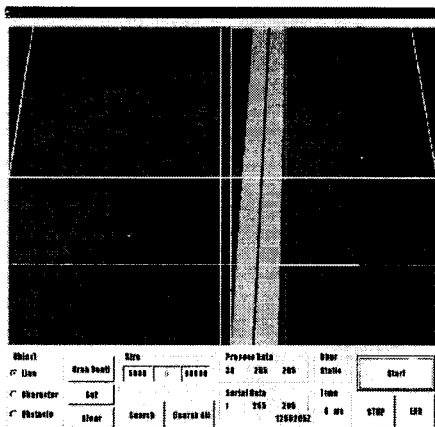


그림2. AGV 모니터 화면
Fig. 2 Screen of AGV Monitor

III. 라인탐색 영상처리

1. Camera Calibration

카메라 Calibration은 기계시각에서 카메라의 파라미터를 추정하는 기초적이고 필수적인 과정으로 시각시스템의 정확성에 영향을 미친다.

Calibration을 통하여 왜곡된 영상을 보정하는데 사용하는데 본 연구에서는 이를 응용하여 실제 공간의 물체 크기를 알아내는데 활용하였다.

단일 카메라로 영상을 해석하는 경우에는 3차원 공간의 물체에 대한 크기나 거리의 판별이 어렵게 된다. AGV 주행 선로 상에 장애물이 존재할 경우에 그 크기나 위치를 인식하여 회피 동작을 하려면 실제 장애물의 크기와 AGV 주행 범위에 얼마만큼 침범했는지를 필수적으로 계산해야 할 것이다.

카메라를 700mm의 높이에 전방향의 아래 50°로 경사시키고 40mm×40mm 마크 9개를 사용하여 실험하였다. 실제 치수는 그림3과 같고, 영상에서 중심좌표를 측정하여 표1에 나타냈다.

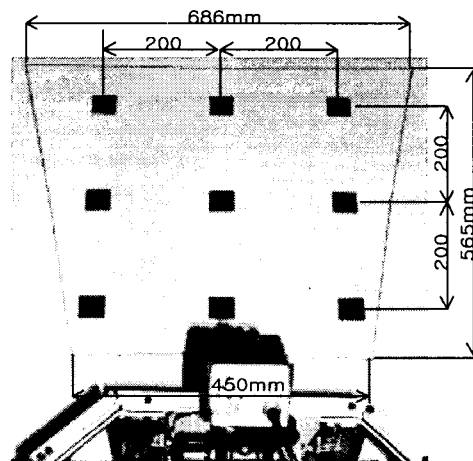


그림3. Calibration에 사용된 인식 마크 영상
Fig. 3 Recognition mark image which is used at Calibration

영상에 잡힌 주행로의 영역은 사다리꼴 모양의 686mm×565mm×450mm 크기이다. 표1의 데이터를 보면 1열의 Y 중심좌표가 2Pixel 정도, 3열의 Y 중심좌표가 3Pixel 정도 차이가 발생하여 영상의 번두리로 갈수록 등골게 약간 왜곡현상이 발생함을 알 수 있다.

표1. Calibration 마크의 영상 좌표
Table.1 Image Coordinate of Calibration mark.

구분	마크 중심좌표		비고
	X좌표	Y좌표	
1	119	53	1행 좌측
2	320	51	1행 중앙
3	521	53	1행 우측
4	88	193	2행 좌측
5	319	193	2행 중앙
6	550	193	2행 우측
7	51	381	3행 좌측
8	318	384	3행 중앙
9	585	381	3행 우측

본 논문에서는 이 같은 왜곡 현상에 대한 영상 보정에는 사용하지 않고 주행 경로 상에 장애물을 발견하였을 때 이 장애물의 크기와 주행 중심선과 떨어진 거리를 계산하는 정보로 활용한다. 표1의 데이터와 마크간의 실제 길이 비를 계산하여 3차원 공간의 물체 크기를 계산한다. 1행의 마크 1, 2, 3간의 길이는 200mm 이나 영상에서는 201Pixel이므로 영상의 좌표가 실제 길이와 유사함을 알 수 있다.

2. Color Line 탐색 실시간 영상처리

이미지 상에서 물체를 탐지하는 방법은 Edge, Coner, Pattern Matching 등 물체의 모양을 이용하는 방법과 물체의 고유한 성질인 Color를 이용하는 방법이 있다. Color는 물체의 모양이 왜곡, 변형되어도 영향을 받지 않는 특성이 있다. 본 연구에서는 모든 물체의 Color 특성만을 이용하여 영상처리 하였다.

Visual C++ 6.0을 사용하여 프로그래밍 하였으며 Color Line 탐색 영상처리 순서는 그림 4와 같다.

2.1 Color Model 선정

영상처리에 사용될 Color Model은 3가지 기본 색상으로 모든 색상을 표현하는 RGB와, 명암정보와 색상정보로 구성된 YUV가 일반적으로 많이 사용된다.

YUV는 U,V의 색상정보로 물체를 탐색하여 처리 데

이터 량이 감소하고 명암에 강한 장점이 있으나 모니터에 출력할 경우에는 다시 RGB로 변환해야하는 단점을 갖고 있어 RGB15 Color Model을 사용하였다.

RGB15는 16비트로 (0,R(5),G(5),B(5))의 데이터 구조를 갖는다.

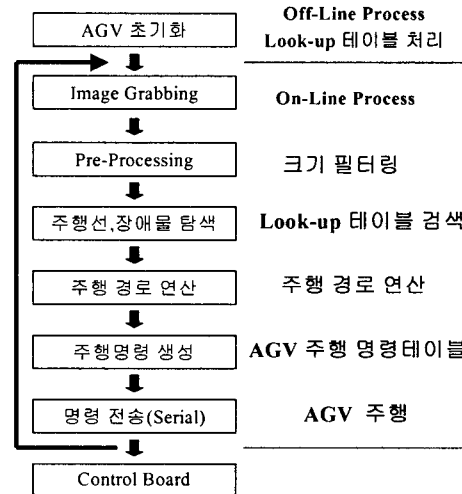


그림4. Color Line 탐색 영상처리 순서도
Fig.4 Flowchart of Image Processing for Color Line Search.

2.2 Off-Line Process

AGV 주행 전에 입력된 영상에서 Color Line과 장애물을 탐색하기 위하여 탐색할 물체의 색상을 모니터에서 영상을 보면서 마우스로 드래그하여 Color를 설정하고, 또한 동일한 Color를 갖는 영상 캡처에 해당하는 화소를 필터링 하는데 사용되는 탐색 대상 물체의 최소, 최대 픽셀 크기 데이터를 설정한다. 이는 프로그램 종료 시 데이터 파일로 저장되며 이 데이터 파일은 다음에 AGV 구동 시에는 재설정이 필요 없이 바로 영상처리에 사용된다.

2.3 주행선, 장애물 탐색

Color를 이용한 물체를 분리할 때 Threshold를 이용하는 방법과 LUT(Look-up Table)를 이용하는 방법이 있다. Threshold를 이용하는 방법은 1픽셀에 대하여 아래와 같은 범위 안의 Color값을 if문을 반복적으로 사용하여야 한다.

$$Rmin < R < Rmax, Gmin < G < Gmax, Bmin < B < Bmax,$$

본 논문에서 사용하는 LUT를 이용하는 방법은 탐색

시간을 줄이고 특정한 Color를 추출하기 위하여 Color 정보를 미리 저장해 놓는 Look-up Table이 필요하다.[5]

그 크기는 RGB15 Color 이므로 R: G: B = 5: 5: 5, 즉 $2^{15} = 32K$ 의 테이블을 사용한다. 그림5와 같이 입력된 영상에서 Color 설정 시 Color Data에 해당하는 LUT에 1로 Set되며 영상처리 시 각 픽셀과 LUT와 비교하여 원하는 Color를 추출하게 된다. 임의 픽셀의 RGB 데이터가 (R,G,B)=(12,8,7)일 경우 LUT의 0x3107번지를 비교하여 1로 Set 되어 있으면 설정된 Color와 같으므로 녹색으로 출력하고 같지 않으면 그 영상을 그대로 출력하게 된다.

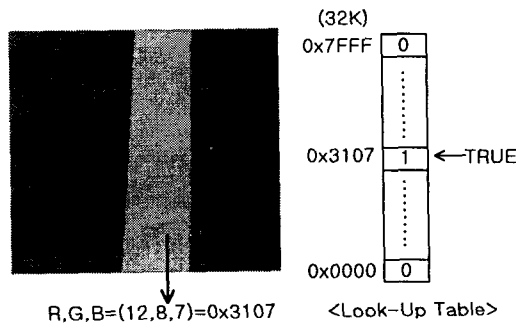


그림5. LUT에서 물체의 분리
Fig. 5 Isolation of the body in LUT

설정된 Color와 동일한 연속된 픽셀을 찾고 그 크기가 설정된 범위 안의 물체를 찾게 되면 영상 잡음에 해당하는 동일한 Color를 갖는 작은 점들은 선택되지 않아 크기에 의한 필터링이 이루어진다.

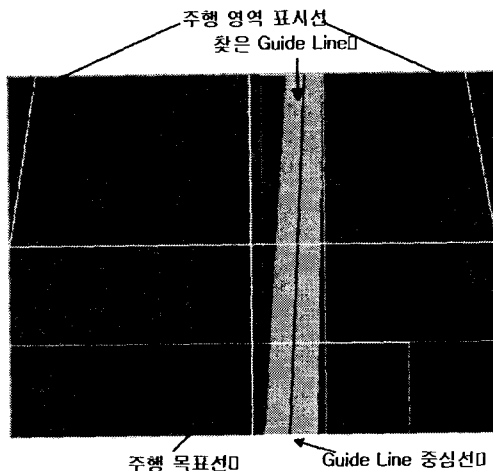


그림6. Color Line 탐색
Fig. 6 Color Line search

그림6은 AGV가 주행 중인 영상을 보인 것 이다. 영상의 중심 수직선을 주행 목표 선으로 백색으로 그리고 탐색한 Color Lin의 중심선을 세로로 8등분하여 곡선부에 대응 하도록 8구간의 중심선을 연결하여 흑색의 Guide Line 중심선을 긋는다. 올바른 주행을 위하여 주행 목표선과 Guide Line 중심선이 항상 일치하도록 틀어진 각도와 진행방향의 길이를 계산하여 두 구동바퀴에 회전속도를 결정하게 된다.

2.4 주행 명령 생성

640×480의 영상에서 AGV의 현재 위치를 320(x),479(y)로 하고 라인 추적 목표점을 탐색라인 7/8등분점으로 하여 주행라인에 비해 AGV가 틀어진 각도(angle)와 주행할 대각선 길이(Dist)를 다음과 같이 계산한다.

$$a \quad n \quad g \quad l \quad e \quad = \quad (\tan^{-1}(480-fy8,320-fx8) \cdot 180 / 3.14159) - 90;$$

$$Dist = \sqrt{(x1-x2)^2 + (y1-y2)^2};$$

각도와 주행할 대각선 길이를 합산하여 적절한 모터속도의 PWM 값이 220~250 범위가 되도록 계산한 후 표 2와 같은 명령을 8Byte로 생성하여 Serial Port1을 통하여 80C196 마이크로컨트롤러에 데이터를 전송한다.

표2는 Control 데이터를 나타내는 것으로 1은 계속 전진명령이고, L과 R은 주행 영역 안에서 장애물을 탐지 하였을 경우 회피 동작의 방향을 나타내며, 4의 경우는 강제적으로 정지 시킬 때 사용하는 명령이다.

표2. 주행 명령 테이블
Table. 2 Table of Navigation Command

순서	구분	전송 명령 데이터			크기	
1	Control	1	L	R	4	1byte
2	left	0~255			3byte	
3	right	0~255			3byte	
4	end data	Z			1byte	

IV. AGV 주행 제어

1. AGV 주행 제어 처리 순서도

Serial Port1을 통하여 80C196 마이크로컨트롤러에 수신된 데이터를 가지고 pwm1과 pwm2에 모터 속도 값을 전송하기 위하여 duty1과 duty2를 계산하고 모니터에서 통신상태를 확인하기 위해 수신데이터를 바로 송신한다. timer2_overflow 인터럽트가 발생하면 encoder 값을 읽어 들이고 카운터의 값에 의해 전진한 길이를 판단하게 된다.

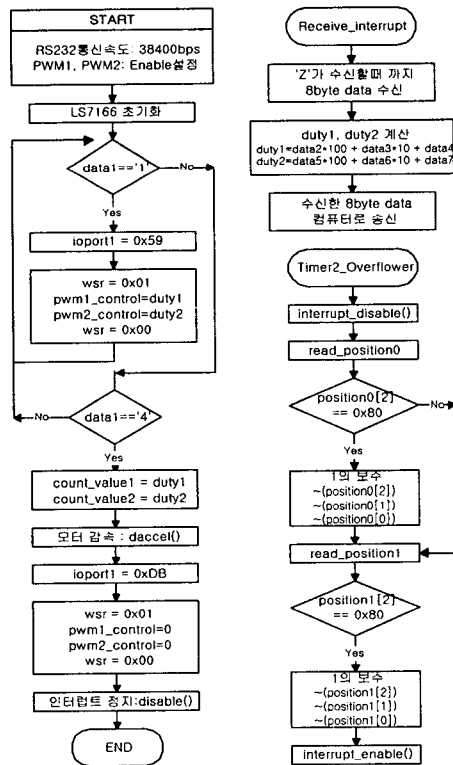


그림7. AGV 주행 제어 처리 순서도
Fig. 7 AGV Navigation Control flowchart

전진과 후진의 경우에 좌우 구동모터의 회전 방향이

반대가 되므로 LS7166 카운터 값도 어느 하나는 0xFFFFFFFF에서 0x000000으로 1씩 감소하고 또 다른 하나는 0x0000000에서 xFFFFFFF으로 1씩 증가하게 된다. 판단의 편리함을 위하여 1씩 감소하는 카운터는 값이 1씩 증가하도록 1의 보수를 취한다.

자세한 80C196 프로그램의 순서도는 그림7과 같다.

V. 주행 실험 및 결과

1. 실험 환경의 주행 모델

실험에 사용할 주행 모델은 그림8과 같은 규격으로 50mm 황색 테이프를 사용하여 직선부와 곡선부를 실내 복도 바닥에 부착하여 실험 환경의 가이드라인으로 사용하였다.

조명은 별도의 조명 장치를 사용하지 않고 다소 불균일한 학교 복도의 형광등을 켜 놓은 상태로 하였다.

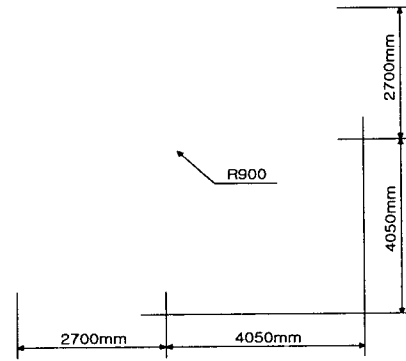


그림8. AGV 주행용 guide line
Fig. 8 Guide line for AGV Navigation

2. 주행 실험 및 결과

직선 주행에서는 라인이 영상에 나타나기만하면 잘 탐색하여 주행하였다. AGV가 후륜 구동으로 되어 있어서 곡선 주행에서는 회전 반경이 600mm 보다 적으면 라인이 영상에 나타나지 않으므로 라인을 이탈하는 현상이 발생하였다. 주요 결과는 다음과 같다.

- 1) 영상의 1Frame을 입력 받아 라인을 탐색하기 까지 6msec가 소요되어 미끄러짐이나 외력에 의한 라인 이탈시 즉시 주행 라인으로 복귀
- 2) 직선 라인의 경우 정속한 주행(PWM값 : 240)과 최대 335mm/Sec의 속도로 라인을 주행
- 3) 후륜 구동으로 곡선 부 주행 시 최소 600mm 이상의 회전 반경이 필요

VI. 결론

본 논문에서는 복도 환경에서 AGV 황색 주행라인을 설치하고 Color 특성만을 추출하여 라인을 탐색한 후 주행 목표선과 탐색한 라인의 중심선과의 각도와 거리를 계산하여 좌우 두 바퀴를 구동하여 가이드라인을 따라 주행하는 방법을 보였다.

본 연구에서 사용하는 방법은 라인을 탐색하는 영상처리 시간이 6msec 밖에 걸리지 않았으며, 바퀴의 미끄러짐이나 외력에 의한 오차가 발생한 위치에서 즉시 자세 교정이 이루어져 주행 라인으로 복귀하여 가이드라인을 따라 정속히 목적지 까지 계속 주행하였다.

AGV는 목적지에서 마크나 숫자 등 표식을 인식하여 정지하거나 필요한 어떤 동작을 할 필요가 있다. 추후 특정한 마크나 숫자인식이 필요하며 주행경로에 장애물이 있을 때 이를 회피하여 주행하는 방법에 관한 연구가 필요하다.

참고문헌

- [1] T.Tsumura, "AGV in Japan-recent trends of advanced research, development, and industrial applications", Proceedings of the 1994 IEEE International Conference on Intelligent Robots and Systems, vol.3, pp. 1477-1484, 1994
- [2] G.Beccari, "Vision-based Line Tracking and Navigation in Structured Environments", Proceedings of the IEEE International Symposium Computational Intelligence in Robotics and Automation CIRA'97, pp. 406-411, 1997
- [3] 김현태, "단일 카메라와 2차원 환경모델을 이용한 이동 로봇의 위치 추정과 복도 주행 제어", 연세대학교 박사학위논문, pp1-7, 1996.
- [4] <http://us.st.com/stonline/bin/fts.exe>
Dual Full Bridge Driver L298 Data Sheet, pp 6-7
- [5] Sun-Gi Hong, O-In Kwon, Ju-Jang Lee, "Real-Time Image Processing Using Color Detection in a Robotic Soccer Vision System", IEEE/ASME Transactions on Mechatronics, 1995.
- [1] T.Tsumura, "AGV in Japan-recent trends of advanced research, development, and industrial applications", Proceedings of the 1994 IEEE International Conference on Intelligent Robots and Systems, vol.3, pp.

저자 소개



박 영 만
1996년 2월-현재: 목포기능대학 컴퓨터정보과 부교수



박 경 우
1990년 3월-현재 : 목포대학교 공과대학 컴퓨터공학과 부교수



안 동 순
1990년 3월-현재 : 목포대학교 공과대학 컴퓨터공학과 교수