무인 자율 주행 차량 시스템 설계 및 제어에 관한 연구

A Study on the Design and Control Method for Unmanned Ground Vehicle System

문 희 창*, 박 명 옥, 김 정 하
(Hee-Chang Moon, Myung-Wook Park, and Jung-Ha Kim)

Abstract: The research presented covers the design and control method of unmanned ground vehicle (UGV). An electric vehicle is used and is driven by DC motor. The power system on the UGV has been adjusted and actuators have been installed for steering and brake automation. A toggle switch is implemented to easily switch between manual and autonomous states. The UGV state is monitored by a velocity sensor, as well as steering and brake position sensors. An emergency stop device was designed and installed to quickly and safely stop the UGV. Different control methods, including the PID controller, were studied for improved steering responsiveness, and results were confirmed through experimentation. Satisfactory performance was achieved and several possible areas of future research have arisen.

Keywords: UGV (Unmanned Ground Vehicle), vehicle control, longitudinal control, lateral control

I. 서론

1. 연구배경

우리나라에 자동차가 소개된 지 100여 년이 흘렀다. 1970년대 이후 고도성장의 중추적인 역할을 해온 자동차는 이제 단순한 이동수단에서 고도로 집적된 첨단 기술을 접목 시키면서 그 영역을 넓혀가고 있다. 더욱이 텔레메틱스와 지능형 교통시스템(ITS) 등과 같이 자동차와 첨단 기반 시설들이 조화를 이루면서 사람이 차량을 조작하지 않고도 움직이고 목적지까지 이동할 수 있는 기술들이 개발 되고 있다[1]. 이러한 무인 자동차 기술은 편리한 시스템 개발이라는 측면뿐만 아니라 현재 사회적인 문제로 대두되고 있는 교통사고 문제와 공장 자동화, 의료 기기의 첨단화로 삶의 질적 향상 등의 효과를 가져 올 수 있으며, 유일한 분단 국가로 남아 있는 우리나라의 커다란 관심 분야인 국방 부문에 까지 큰 영향을 미치게 된다.

무인 자동차에 대한 개발과 적용은 현재 상용차와 군사 분야에서 활발히 진행되고 있다. 상용차의 경우 대표적인 예로 적응 순항 제어시스템 (ACC: Adaptive Cruise Control)을 들 수가 있다. 이 시스템은 차선유지 시스템, 충돌경고 시스템 등과 더불어 운전자의 운전을 도와주기 위해 차량에 적용된 전자제어기술 중 하나로 운전자에 의하여 조향(steering control) 및 차량의 가속(throttle control)과 감속(brake control)이 차량에 부착된 센서를 통해 인식된 선행 차량과의 거리 및 상대속도 등에 의해 판단된 결과를 통해 자동적으로 조절되는 시스템이다[2,3].

군사 분야의 경우 두 차례의 큰 전쟁을 치른 미국에서 활발한 무인 자동차 연구가 진행되고 있다. 그 예로 미국방성은 무인 자동차 경주 대회인 DARPA Grand Challenge 및 Urban Challenge 등을 개최하여 무인 자동차 개발에 힘쓰고

있으며 미국 국방성은 2015년까지 미군 수송차량의 1/3을 무인화 하려는 계획을 가지고 있다.

우리나라에서도 국방 과학 분야를 중심으로 무인기술에 대한 연구를 실시하고 있다. 2012년까지 전방 일반 전초 (GOP: General Out-Post)의 경계력 강화 조치의 일환으로 GOP 과학화 경계 시스템을 단계별 시험, 보급할 예정이며, ‘견마 로봇’을 개발해 군수 및 민간 분야에 활용할 뜻을 밝혔다. 또한, 국방과학연구소에서는 2020년까지 야지용 무인 자율 주행 차량을 총 10단계의 자율레벨을 구분하여 연구 목표를 설정하고 여러 방위 산업체와 공동 연구 개발을 하고 있다. 그리고 무인 지상무기와 함께 무인화 프로젝트가 한창 진행 되고 있는 항공기 분야에서도 미국, 이스라엘, 영국, 프랑스에 이어 무인기술을 이용한 스마트 무인항공기 (UAV: Unmanned Aerial Vehicle)기술개발사업을 추진하고 있으며 2012년까지 완료를 목표로 하고 있다[4].

이렇듯 무인 기술은 로봇 산업뿐만 아니라 군사 및 실생활에서도 많이 보급되고 있으며 앞으로도 기술적, 경제적인 파급 효과가 클 것으로 기대를 모으고 있다.

2. 연구 목적

무인 자율 주행 차량은 적응 순항 제어시스템(ACC)과 같이 차량 시스템 안에서 독립적인 시스템으로 작동되는 것이 아니라 서로 다른 목적을 가지고 있는 여러 시스템들 예를 들어 차량, 장애물, 비전, 항법, 통합 시스템이 주행 상황에 따라 적절한 우선권을 가지고 차량을 제어하게 된다.

무인 자율 주행 차량에서의 차량 시스템은 무인 자율 주행 차량에 있어 가장 기본이 되는 플랫폼으로 각종 센서와 장치들이 탑재가 되며 이들로부터 오는 신호를 바탕으로 조향, 가속, 브레이크의 장치를 By-wire로 구성된 Smart actuator로 각각 작동시켜 차량을 움직이게 된다. 장애물 시스템은 초음파센서나 레이저스캐너와 같은 장애물 감지 센서를 이용하여 무인 자율 주행 차량의 전방 또는 측방의 장애물을 감지하여 차량의 주행경로 확보 및 장애물회피 목적으로 사용하게 된다. 비전 시스템은 카메라를 이용하여 차량의 전방 및 측면의 물체를 인식하고 물체까지의 거리를 측정하며 포장

* 책임저자(Corresponding Author)

논문접수: 2008. 3. 19., 수정: 2008. 7. 2., 채택확정: 2010. 2. 22.

문희창: Univ. of Florida(moonhc2@gmail.com)

박명옥: 국민대학교 자동차공학전문대학원(solyson@naver.com)

김정하: 국민대학교 기계자동차공학부(jhkim@kookmin.ac.kr)

도로(on road) 또는 비포장도로(off road)에서 차량의 원활한 주행을 위해 주행 노면 인식 등 다양한 목적으로 사용된다. 항법시스템은 GPS (Global Positioning System)를 가지고 차량의 현재위치를 측정하고 차량 주행에 필요한 경로점(waypoint)설정 및 전역경로계획(global path planning)을 하는 시스템이다[4]. 마지막으로 통합 시스템은 무인 자율 주행 차량을 구성하는 각각의 시스템에서 발생하는 정보를 통합하여 차량의 주행 상태와 장애물 정보를 비교하여 차량의 최적경로 생성 및 차량의 주행 안전성을 고려하여 차량 시스템에 제어 명령을 전달하여 차량의 움직임을 제어하게 된다.

본 논문은 무인 자율 주행 차량 시스템의 설계와 제어에 관한 논문으로 무인 자율 주행 차량을 개발하는 데 있어 무인주행에 알맞은 시스템 구축과 제어 알고리즘을 개발하는 데 목적을 두고 있다. 나아가 전기차량이 아닌 엔진차량을 바탕으로 무인 시스템을 설계해 나갈 수 있는 가능성을 검증하고자 한다.

II. 시스템 구성

무인 자율 주행 차량은 크게 6개의 시스템으로 나눌 수 있다. 그림 1은 무인 자율 주행 차량의 시스템구성도이며 구성하는 시스템은 다음과 같다. 차량을 제어하는 차량 시스템, 장애물 감지하는 장애물 시스템, 장애물과 지형정보 획득을 목적으로 하는 비전 시스템, 위치 정보와 경로 정보를 설정하는 항법 시스템, 각 시스템을 총괄하는 통합 시스템, 마지막으로 원격 조종하거나 모니터링 할 수 있는 원격 시스템으로 구성되어 있다. 각 시스템은 여러 센서에서 발생하는 데이터를 획득하고 처리하여 통합 시스템에 전달할 데이터를 생성하고 UDP버스를 통해 통합 시스템에 전달한다. 통합 시스템은 각 시스템에서 전송된 데이터에 우선권(priority)을 부여하고 데이터의 중요도에 따라 차량의 제어명령을 다르게 계산하여 최종적으로 차량의 움직임 제어를 위한 명령을 생성한다. 이 명령은 다시 차량 시스템으로 전송되어 조향 및 가속페달을 제어하여 차량의 이동, 회피, 정지 등 차량을 제어하게 된다.

1. 차량 시스템

차량 시스템은 무인 자율 주행 차량에서 가장 기본이 되는 시스템이며, 자율주행을 위해 통합 시스템의 제어 명령에 따라 조향 및 주행이 가능해야 한다. 또한 자율모드와 매뉴얼

그림 1. 무인 자율 주행 차량의 시스템 구성도.

Fig. 1. Configuration of unmanned ground vehicle system.

그림 2. 차량 시스템구성도.

Fig. 2. Configuration of vehicle control system.

모드의 전환이 간단하고 용이해야 하며, 차량의 상태를 모니터링 할 수 있는 시스템을 개발해야 한다. 그리고 무엇보다 중요시 해야 할 사항은 자율모드 동작 중 시스템 오류로 인한 사고를 방지하는 것이다. 이를 위해 안전장치를 고려해야 한다. 그림 2는 차량 시스템의 전체적인 구성을 보여주고 있다.

1.1 차량

무인 자율 주행 차량을 개발하기 위해 본 연구에서 사용된 차량은 2.2KW, 3HP의 48V DC 모터가 장착된 전기차량 (EV: Electronic Vehicle)으로, 세부 제원 및 사양은 표 1과 같다. 차량 시스템과 각 시스템의 전원공급을 위해서 최대출력 2KW의 발전기를 장착하였다.

그림 3. 무인 자율 주행용 전기차량.

Fig. 3. Electronic unmanned ground vehicle.

표 1. 전기차량의 사양.

Table 1. Specification of EV.

Length	2300 (mm)
Width	1195 (mm)
Height	1240 (mm)
Wheel Base	1665 (mm)
Height	1240 (mm)
Wheel Tread	Front: 925 (mm) Rear: 985 (mm)
Weight	250 (Kg)
Turning Radius	2.7(m)
Steering	Rack & Pinion Gear
Brake	Mechanical Brake Cable System To Drum Brake
Motor	DC2.2 KW/3 HP
Trans Axle	Double Reduction Helical Gear

1.2 종방향 시스템

무인 자율 주행 차량의 종방향 제어 요소는 가속, 제동, 변속으로 구성된다. 각 요소는 주행 환경에 따라 선택운용 된다. 차량의 구동을 위한 요소는 연구에 사용한 전기차량 자체 구동모터이다. 차량의 속도를 측정하기 위하여 각도 분해능이 0.36° 인 엔코더를 좌, 우 뒷바퀴에 각각 설치하였다.

본 연구에서 사용된 차량의 브레이크는 기계식 드럼(drum) 브레이크로 브레이크페달과 와이어로 연결되어 있다. 브레이크페달의 위치제어를 위해 70:1의 감속기가 장착된 200W급 AC 서보모터를 사용하였다. 모터와 브레이크페달은 수동과 자동으로 작동이 가능하도록 유연한 벨트를 이용하여 브레이크 페달과 연결하였다. 또한 브레이크의 페달의 초기 위치 설정과 상태를 확인하기 위해 모터 축에 Rotary Potentiometer를 그림 5와 같이 장착하였다.

본 연구에서 사용한 전기차량의 변속제어는 구동모터의 제어기에 신호를 입력하여 모터의 회전방향을 결정한다. 구동모터의 회전방향은 차량에 장착된 변속스위치에 의해 선택된 방향이 48V 전압으로 차량 모터 컨트롤러로 입력되어 모터의 회전방향을 결정하게 된다. 자율 주행 모드에서 변속제어를 하기 위해서 DAQ의 Digital Output에서 5V의 신호를 SSR (Solid State Relay)에 입력하여 전진, 후진 및 중립을 위한

그림 4. 차량 구동용 DC 모터와 속도 측정용 엔코더.
Fig. 4. DC motor for vehicle traction and encoder for velocity measurement.

그림 5. 브레이크제어를 위한 서보 모터와 포텐쇼미터.
Fig. 5. Servo motor and potentiometer for brake control.

그림 6. 변속제어를 위한 SSR 제어 판넬.
Fig. 6. SSR control panel for transmission control.

그림 7. 조향제어를 위한 서보 모터와 포텐쇼미터.
Fig. 7. Servo motor and potentiometer for steering control.

그림 8. 비상 정지를 위한 원격제어기와 스위치.
Fig. 8. Remote controller and Switch for E-Stop.

48V의 전압을 모터드라이버로 인가하게 된다.

1.3 횡방향 시스템

횡방향 제어는 차량의 주행 안정성뿐만 아니라 경로 추종 성능에 큰 영향을 미친다. 이에 따라 본 연구에서는 조향력 확보를 위해 15:1의 감속기가 장착된 100W급 AC 서보 모터를 사용하였다. 이 모터를 타이밍 폴리와 벨트를 이용하여 조향 축과 연결하였고, 정확한 조향각 측정을 위해 그림 7과 같이 앞 바퀴 허브와 차체를 Linear Potentiometer로 연결하였다.

1.4 E(Emergency)-stop 시스템

무인 자율 주행 차량의 시스템 운용에 있어 불안정 또는 오류 등의 문제는 사고 발생의 원인이 되어 차량 및 구동기에 손상을 줄 수 있다. 자율 모드 주행시 각 시스템과의 통신 오류와 그 밖에 차량 시스템에서 구동기와 통신오류 등으로 인한 오작동을 할 경우 시스템은 긴급정지를 해야 한다. 이는 통합 시스템이나 차량 시스템에서 통신 오류를 검출하여 차량의 주행을 바로 멈출 수 있어야 하며, 원격 제어를 통해 차량을 강제로 멈추어야 한다. 본 연구에서는 RF 송·수신기를 이용하여 원격 비상 정지회로를 구성하였고 차량에도 그림 8과 같이 E-Stop 스위치를 장착하였다.

III. 차량 시스템 제어

무인 자율 주행 차량 제어시스템은 통합 시스템의 명령에 따라 차량의 주행시 필요한 조향, 가속, 제동 그리고 변속을 이행하게 된다. 차량 시스템의 제어요소는 종방향 제어와 횡방향 제어 그리고 E-Stop 제어로 나눌 수 있다. 본 연구에서는 이 세 가지의 제어요소를 가지고 연구를 진행하였으며, 주행 상황에 맞게 서로 유기적으로 작동할 수 있도록 하였다.

1. 종방향 제어

1.1 종방향 시스템 모델링

무인 자율 주행 차량의 구동부는 2.2KW, 3HP의 동력성을 가지는 48V DC모터와 모터드라이버 그리고 Transaxle로 이루어져 있다. 가속 페달 작동 유격에 비례하여 발생하는

그림 9. DC 모터의 수학적 모델.

Fig. 9. DC motor mathematical model.

전압(0~5V)을 모터드라이버의 입력신호로 사용하며 이에 비례하여 구동모터제어를 위한 PWM (Pulse Width Modulation) 신호를 발생하여 모터의 속도를 제어하게 된다.

모터를 구동시키는 것에서부터 동력 전달장치를 통해 바퀴를 회전시키기까지의 과정을 DC 모터에 전압을 인가하여 회전하게 되는 것으로 단순화 하게 되면 그림 9와 같다.

DC 모터의 운동 방정식은 다음과 같다.

$$J\ddot{\theta} + b\dot{\theta} = Ki \quad (1)$$

$$L\frac{di}{dt} + Ri = V - K\dot{\theta} \quad (2)$$

식 (1)과 (2)를 사용하여 라플라스 변환을 하고 전달함수를 구하면 식 (3)과 같다.

$$\frac{\omega(s)}{V(s)} = \frac{K}{JLs^2 + (JR + Lb)s + bR + K^2} \quad (3)$$

그러나 종방향 제어를 위해 위의 전달함수 식을 사용하는 것은 한계가 있다. 우선 이 모터의 회전관성모멘트나 내부저항(R) 및 인덕턴스(L) 등 매개변수의 정확한 값을 알 수가 없기 때문이다. 그리고 모터의 회전이 모터에 설치된 감속기와 기어 등의 기계적인 구조를 통해 마지막으로 뒷바퀴에 전달되기 때문에 그 중간 과정에 대한 모델링을 할 수 없다. 따라서 전달함수를 구하기 위해 본 연구에서는 과도응답 데이터를 통해 시스템의 전달함수를 구하였다[6].

과도응답 데이터를 이용하는 방법은 우선 시스템의 계단응답이 지수의 합에 의하여 주어진다고 가정하면, 다음과 같이 쓸 수 있다.

$$y(t) = y(\infty) + Ae^{-\alpha t} + Be^{-\beta t} + Ce^{-\gamma t} + \dots \quad (4)$$

최종값을 빼고 $-\alpha$ 가 가장 늦은 근이라고 가정하면 다음과 같이 된다.

$$\begin{aligned} y - y(\infty) &\cong Ae^{-\alpha t} \\ \ln[y - y(\infty)] &\cong \ln A - \alpha t \\ &\cong \ln A - \alpha t \end{aligned} \quad (5)$$

이것은 기울기가 $-\alpha$ 이고, y축과 만나는 점이 $\ln A$ 인 직선의 방정식이다. 만일 $\ln[y - y(\infty)]$ (혹은 A가 음이면 $\ln[y(\infty)-y]$)의 그림을 선으로 대체하면, A와 α 를 추측할 수 있다. 일단 이것이 추측되면 $y - [y(\infty) + Ae^{-\alpha t}]$ 를 그린다. 이는 근사적으로 $Be^{-\beta t}$ 이고, log 그림에서는 $\ln B - \beta t$ 인 곡선과 같다. 매번 가장 늦은 항을 제거하여 데이터의 정확성을 얻을 때까지 그 과정을 반복한다. 최종 모델의 계단응답을 그리고 나서 계산된 모델의 질을 평가하기 위해 그것을 데이터와 비교한다. 이

그림 10. 속도에 대한 전기차량의 과도응답데이터.

Fig. 10. Transient response data of EV about velocity.

방법을 현재 시스템에 적용하였으며, 그 과정과 결과는 다음과 같다.

그림 10과 표 2에서 무인 자율 주행 차량의 종방향에 대한 계단응답 데이터를 보여주고 있다. 이는 데이터의 최종값이 '1'이라는 것을 알 수 있으며, $y(\infty)$ 가 $y(t)$ 보다 크기 때문에 A가 음이라는 것을 알 수 있다. 그러므로 $\ln[y(\infty)-y]$ 를 구하여 A와 α 를 구할 수 있다.

$\ln[y(\infty)-y]$ 에 대한 그래프가 그림 11에 나타나 있으며, 추세선의 기울기와 절편으로부터 A와 α 를 구하면 다음과 같다.

$$A = -1.555965632, \quad \alpha = 1.220354594$$

위 결과로부터 구해진 A와 α 를 식 (4)에 대입하게 되면

표 2. 속도에 대한 전기차량의 과도응답데이터.

Table 2. Transient response data of EV about velocity.

t	y(t)	t	y(t)
0	0	1.65625	0.694973
0.140625	0	1.859375	0.79595
0.34375	0.032485	2.15625	0.900523
0.65625	0.169865	2.75	0.97084
0.953125	0.3565056	∞	1
1.265625	0.500797

그림 11. A, α를 찾기 위한 $\ln[y(\infty)-y]$ 의 그래프.Fig. 11. $\ln[y(\infty)-y]$ graph for finding A and α .

그림 12. B, β 를 찾기 위한 $\ln[y - (1 + Ae^{-at})]$ 의 그래프.

Fig. 12. $\ln[y - (1 + Ae^{-at})]$ graph for finding B and β .

그림 13. 과도응답 데이터의 지수함수 그래프.

Fig. 13. Exponential function graph of transient data.

$y(t)$ 에 대한 식은 다음과 같다.

$$y(t) = 1 - 1.55597e^{-1.22035t} + Be^{-\beta t} + Ce^{-\gamma t} + \dots \quad (6)$$

이 과정을 반복하여 B와 β 를 구하기 위해 계단응답 데이터에서 $1 + Ae^{-at}$ 을 빼고 그 결과에 대한 로그 그래프를 그리면 그림 12와 같다. 여기서 B와 β 를 추정하면 다음과 같게 된다.

$$B = 0.6412095766, \quad \beta = 6.83628828$$

위의 모든 결과를 종합하여 $y(t)$ 의 추측값을 계산하면 식 (7)이 된다.

$$y(t) = 1 - 1.55597e^{-1.22035t} + 0.64121e^{-6.8363t} \quad (7)$$

식 (7)에 표 2에서의 시간 t 를 대입하여 표현한 그래프는 그림 13에서 보여준다. 이 그림에서는 $t=0$ 근처에서 y 값은 음수를 갖는다. 이를 개선하기 위하여 B와 β 값을 변화시켜서 원래의 데이터에 근접하도록 관계식의 계수를 조절하였으며 이를 통하여 얻은 관계식은 식 (8)과 같으며 식 (8)에 대한 그래프는 그림 14에 나타내었다.

$$y(t) = 1 - 1.55597e^{-1.22035t} + 0.55621e^{-4.141t} \quad (8)$$

그림 14에서 실제 실험을 통해 차량에서 얻어진 응답과 수학적 모델로부터 구한 관계식을 통한 응답을 볼 수 있다. 두 응답이 시정수의 차이는 있지만 비슷한 양상을 보이고 있

그림 14. 과도응답과 최종 지수함수 데이터의 비교.

Fig. 14. Comparison between transient response and the last exponential function data.

다. 따라서 식 (8)을 라플라스 변환하면 식 (9)와 같다.

$$Y(s) = \frac{1}{s} - \frac{1.55597}{s + 1.22035} + \frac{0.55621}{s + 4.141} = \frac{-0.40313s + 5.05349}{s^2 + 5.36135s + 5.05349} \quad (9)$$

식 (9)를 이용한 구동을 위한 최종 전달함수는 다음 식 (10)과 같다. 이 식은 RHP (Right Half Plane)에 영점이 존재하지만 영점의 위치가 $s=20$ 보다 작기 때문에 시간응답에 대한 영향을 무시할 수 있다.

$$G(s) = \frac{-0.40313s + 5.05349}{s^2 + 5.36135s + 5.05349} \quad (10)$$

1.2 페루프 시스템과 제어기 설계

구동부의 속도를 제어하는데 있어 제어 목표에 맞는 응답을 얻기 위해 PID 제어기를 설계하였다.

구동부의 PID 제어를 위해 엔코더로부터 속도값을 피드백 받아 Closed Loop system을 구성하였으며, 속도에 따라 가속 및 감속이 적절히 이루어지도록 하였다. 구동부 Closed Loop system의 제어 흐름도는 그림 15와 같다.

그림 16은 컨트롤러를 설치하지 않은 Open-Loop system의 속도에 대한 계단응답을 보여주고 있다. 속도 제어를 할 수 없기 때문에 차량의 최대 속도인 20km/h까지의 응답을 측정하였다. Settling Time(t_s)이 대략 2.8~3초인 것을 확인할 수 있었다. 차량을 사람이 운전할 경우에 응답시간이 빨라도 대처할 수 있지만 자율 주행에서 속도입력에 대해 응답이 빠르면

그림 15. 종방향 제어 흐름도.

Fig. 15. Control flowchart for longitudinal control.

그림 16. 전기 차량의 속도응답(제어기가 없을 경우).

Fig. 16. Velocity response of EV (none controller).

표 3. 종방향 제어기의 PID 계수.

Table 3. PID Coefficient of longitudinal controller.

K_p	0.00001
K_i	0.0368
K_d	0

그림 17. 전기차량의 속도응답(제어기가 있을 경우).

Fig. 17. Velocity response of EV (with controller).

차량의 주행 안정성에 영향을 줄 수 있다고 판단되어 제어기의 Settling Time(t_s)을 6~7초로 사람보다 느린 응답을 제어 목표를 설정하였고 Steady State Error는 1%로 제어 목표를 설정하였다. Overshoot Response는 5% 미만으로 설정하였다.

위에서 설정한 제어 목표를 바탕으로 PID 제어기의 계수 값을 조절하여 적절한 수치를 얻게 되었으며, 그 결과를 그림 17에서 보여주고 있다. 각각의 계수 값은 표 3과 같고, 실험을 하면서 미분기의 영향은 거의 받지 않았기 때문에 계수 값을 '0'으로 결정하였다.

2. 횡방향 제어

2.1 횡방향 시스템 모델링

차량운동을 모델링 하는 것은 시뮬레이션하는데 있어서 필수적인 뿐만 아니라, 제어시스템을 구성 할 때에도 중요한 요소가 된다. 그림 18은 선형 2자유도 모델인 Bicycle Model을 보여주고 있다. 그리고 선형 2자유도 모델은 차량의 횡방향 운동과 요 운동에 대한 방정식으로 식 (11)과 (12)와 같이 표현된다[7,8].

$$mV\left(\frac{d\beta}{dt} + \gamma\right) = -2K_f\left(\beta + \frac{l_f}{V}\gamma - \delta\right) - 2K_r\left(\beta - \frac{l_r}{V}\gamma\right) \quad (11)$$

그림 18. 자전거 모델.

Fig. 18. Bicycle model.

$$I \frac{d\gamma}{dt} = -2K_f\left(\beta + \frac{l_f}{V}\gamma - \delta\right)l_f + 2K_r\left(\beta - \frac{l_r}{V}\gamma\right)l_r \quad (12)$$

차량이 정상원 선회를 할 경우 중심점의 옆 미끄럼각의 변화는 없고, 요 각속도도 일정하므로 식 (11)과 식 (12)에서 정상상태의 조건 $d\beta/dt = 0$, $d\gamma/dt = 0$ 를 대입하여 다음 식과 같이 기술 할 수 있다.

$$2(K_f + K_r)\beta + \left[mV + \frac{2}{V}(l_f K_f - l_r K_r)\right]r = 2K_f\delta \quad (13)$$

$$2(l_f K_f - l_r K_r)\beta + \frac{2(l_f^2 K_f + l_r^2 K_r)}{V}r = 2l_f K_f\delta \quad (14)$$

식 (13)과 식 (14)를 옆 미끄럼각 β 에 대하여 정리하면 다음과 같다.

$$\beta = \left(\frac{1 - \frac{m}{2l} \frac{l_f}{l_r} K_r V^2}{\frac{m}{2l^2} \frac{(l_f K_f - l_r K_r)}{K_f K_r} V^2} \right) \frac{l_r}{l} \delta \quad (15)$$

식 (15)에서 옆 미끄럼각 β 는 차량의 속도와 현재 차량의 속도에 관계된 식임을 알 수 있다.

2.2 조향 응답성 제어

차량이 주행 중 조향을 하였을 때 지면의 마찰계수가 작거나, 차량의 속도가 클 경우 대부분의 차량은 언더스티어 (understeer) 현상을 보이며 이는 제어기가 원하는 만큼의 조향 응답을 보여주지 않는다. 이는 차량이 회전을 하거나 바깥쪽으로 밀리는 현상 때문이다. 이를 해결하기 위해 제어기에서 원하는 조향값에 부족한 조향값을 더해주는 방법으로 제어를 하였다[9].

그림 19. 조향 응답성 제어 블록도.

Fig. 19. Block diagram of steer responsiveness control.

그림 19는 조향 응답성 제어의 구조를 보여주고 있으며, 위 제어를 통해 차량 시스템으로 입력되는 조향값은 식 (16)과 같다.

$$\delta_{des} = \delta_{arbiter} + \delta_{motor} \quad (16)$$

여기서 δ_{des} 는 최종적으로 조향제어를 위한 구동기에 입력되는 조향값이고, $\delta_{arbiter}$ 는 통합 시스템에서 계산한 조향값으로 차량 시스템으로 전달되는 명령값이다. δ_{motor} 는 차량 시스템에서 조향응답성을 높이기 위해 통합 시스템에서 계산된 값에 더해지는 값이 된다. 부족한 조향각의 결정을 통합 시스템에서 요구하는 조향값과 차량의 센서에서 계산되는 속도 그리고 실험차량의 운동특성에 의해 결정되며, 그 값은 차량이 정상원선회를 할 경우에 차량 중심에 발생하는 옆미끄럼각 $-\beta$ 와 같게 했다.

$$\delta_{motor} = -\beta \quad (17)$$

2.3 페루프 시스템 설계

횡방향 제어도 종방향 제어와 같이 페루프제어기를 설계하였다. 제어기를 통하여 계산된 조향각은 차량의 속도에 따라 최대 조향각이 변동이 되며 실제 차량의 최대 조향각 범위가 $-30^{\circ} \sim 30^{\circ}$ 일 때 제어기를 통해 계산된 최대 조향각 범위는 차량의 주행 안전성을 고려하여 $-25^{\circ} \sim 25^{\circ}$ 로 제한을 두었다.

3. E-stop 제어

차량이 자율 주행 모드로 주행할 경우 발생하는 위험 상황은 여러 가지가 있다. 차량 주위에 갑작스런 장애물로 인한 환경적 요인과 무인 자율 주행 차량의 각 시스템의 오류에

그림 20. 횡방향 제어의 제어흐름도.

Fig. 20. Control flowchart for lateral control.

그림 21. 비상정지의 순서도.

Fig. 21. Flowchart of E-Stop braking.

그림 22. 무인 자율 주행 차량의 배선도.

Fig. 22. Wiring diagram of UGV.

의한 시스템적 요인이 있다. 환경적 요인은 차량에 설치된 레이저 스캐너와 카메라를 이용해 장애물을 감지하여 상황에 맞게 적응할 수 있다. 하지만 시스템적인 요인은 스스로 적응할 수 없는 상황에 이르게 할 수 있기에 상황에 맞는 대응 방안이 필요하게 된다. 본 연구에서는 시스템적인 요인을 시스템간의 통신오류로 인한 요인과 차량 시스템의 오류로 인한 요인으로 제한을 두었다.

3.1 시스템간 통신오류에 대한 비상정지

현재 무인 자율 주행 차량에서 각 시스템간의 통신은 UDP (User Datagram Protocol)가 사용되고 있다. UDP의 정보 전송 방식은 서로 주고받는 형식이 아니라 한쪽에서 일방적으로 보내는 방식의 통신 프로토콜이다. 이런 통신 방식의 문제점은 통신 에러를 검출할 수 없기 때문에 시스템간 통신이 끊어지면 시스템 스스로 확인할 방법이 없다는 것이다.

이러한 통신의 문제점을 개선하기 위해서 시스템간 TCP (Transmission Control Protocol)/IP(Information Provider) 통신 방식을 추가 했다[10,11]. 이 방식은 서버와 클라이언트 사이에 통신 에러가 발생하면 시스템에서 통신에러 메시지가 발생하고 발생한 메시지를 확인함으로써 어느 시스템에서 통신 에러가 발생하였는지 알 수 있다. 본 연구에서는 무인자율주행시스템을 구성하는 시스템간의 통신연결은 TCP/IP로 구현하고 데이터는 고속통신방법인 UDP로 데이터 통신을 하였다. 만약 차량 시스템이 동작 중에 다른 시스템과 통신오류가 발생하면 차량 시스템은 E-Stop Mode로 전환되어 차량을 정지시킨 후 사용자명령을 기다리게 된다.

3.2 차량 시스템의 오류에 대한 비상정지

차량 시스템에서의 오류는 제어기와 구동기 사이의 통신 오류와 센서 자체의 오류 등으로 나눌 수 있다. 이들 오류가 발생할 경우 무인 자율 주행 차량을 제어할 수 없게 되어 큰 위험을 초래할 수 있다. 따라서 본 시스템에서는 이를 방지하기 위해 비상정지 기능을 구축하였다. 이 기능은 그림 22와 같이 RF 송·수신기를 가지고 구성된 E-Stop 리모트 컨트롤러와 차량에 장착된 비상정지 스위치를 통해 비상시에 리모트 컨트롤러나 스위치가 동작하였을 때 차량의 전원을 차단하도록 하였다.

IV. 실험 및 결과

1. 가속 및 감속 실험

차량의 가속 및 감속 실험은 설계한 제어기의 성능을 확인하고 실제 주행을 할 수 있는가에 대한 실험이다. 그림 23은 사용자가 임의의 속도를 차량 시스템으로 입력하였을 경우 차량의 속도와 감속을 위한 브레이크 페달의 position에 대한 그래프이다.

전체적으로 통합 시스템에서 전달되는 속도명령을 잘 추종하고 있는 것으로 확인 되었다. 무인 자율 주행 차량의 반응시간은 0.828sec로 이 시간은 통합 시스템에서 차량 시스템으로 데이터를 전송하고 이것을 차량 시스템이 차량을 구동시킨 후 데이터를 피드백 받는 시간 지연으로 보인다. 또한 감속제어는 속도가 '0'일 경우에는 full braking을 하도록 설정해 놓았으며, 속도를 입력하게 되면 브레이크 페달의 위치가 0°로 움직이게 된다. 브레이크의 반응시간은 0.203sec로 만족할 만한 결과를 얻었다.

2. 조향 응답성 실험

그림 24는 주행 속도 10km/h에서 차량의 조향각을 변화시킴에 따른 실제 차량의 조향각 변화를 보여주고 있다. 통합 시스템에서 차량 시스템으로 요구한 조향값과 실제 차량 시스템에서 모터로 명령하는 조향각의 차이가 많다는 것을 볼 수 있다. 이론적으로 옆 미끄럼각은 차량의 속도가 작을 경우 무시할 정도로 작다. 하지만 실험 결과는 10km/h에서 12°의 조향을 할 경우 실제로 입력되는 값은 두 배 이상이다. 이 원인은 차량의 파라미터를 구하는 과정에서 상당히 큰 오

그림 23. 가속 및 감속 테스트 결과.

Fig. 23. Acceleration and deceleration test results.

그림 24. 조향응답성 실험 결과.

Fig. 24. Result of steer response test.

차가 있었기 때문이다. 여러 시스템을 설계하면서 생기는 무게 변화 또는 무게 중심을 정확하게 획득하기가 어렵다. 그리고 선회강성계수 또한 중요한 변수로서 앞의 데이터가 정확하지 못하면 이 역시 상당한 오차가 발생한다. 이러한 오차의 누적으로 인해 목표했던 결과를 얻지 못했지만 차량에 대한 파라미터가 정확하고 이들을 추정할 수 있다면 보다 좋은 결과를 얻을 것이라 본다.

3. 시스템간 통신오류에 대한 비상정지 실험

본 연구에서 무인 자율 주행 차량이 험로를 주행하는 데 있어 통신에 대한 안정성 확보를 위하여 통신오류가 발견되었을 경우 차량 시스템에서 이것을 인식하고 비상정지를 하는 방안을 고려하였다.

통신오류가 발생했을 경우 그 신호를 '1'로 하였으며, 실험을 위해 인위적으로 오류신호를 발생시켰다.

그림 25와 26은 각각 주행속도 5km/h, 10km/h일 경우의 실험이다. 그림 25를 보게 되면 통신 오류가 검출된 후 속도가 '0'이 될 때까지 걸리는 시간은 약 1.656sec인 것을 확인할 수 있었다. 또한 제동거리가 약 180cm 였다. 주행속도 10km/h에서의 실험에서는 응답속도가 약 2.7sec이며, 제동거리는 약 465cm로 확인되었다.

4. 차량 시스템 오류에 대한 비상정지 실험

차량 시스템에서 오류가 발생하였을 경우에는 리모트 컨트롤러를 통해 차량을 정지시킬 수 없다. 이러한 경우를 대비해서 설계된 모듈은 사람이 인위적으로 차량을 멈추게 해야 한다.

실험은 차량 속도 5km/h, 10km/h에서 차량의 전원을 차단했을 때의 차량의 속도변화를 관찰하였다. 오류 검출 신호를

그림 25. 통신오류에 의한 비상정지 실험(5km/h).

Fig. 25. E-Stop test about communication error(5km/h).

그림 26. 통신 오류에 의한 비상정지 실험(10km/h).

Fig. 26. E-Stop test about communication error(10km/h).

그림 27. 전원 차단에 대한 비상정지 실험(5km/h).

Fig. 27. E-Stop test about cutoff power(5km/h).

그림 28. 전원 차단에 대한 비상정지 실험(10km/h).

Fig. 28. E-Stop test about cutoff power(10km/h).

‘1’로 정하였고, 앞 절에서의 실험과는 달리 브레이크 제어를 할 수 없으므로, 차량이 멈추는 시간이 더 증가한 것을 확인할 수 있다.

차량의 주행속도가 5km/h일 때의 실험 결과를 그림 27에서 보여주고 있다. 응답속도는 약 3.015sec였으며, 제동거리는 약 204cm로 확인되었다.

그림 28은 주행속도가 10km/h일 경우의 실험 결과이다. 응답속도는 약 4.94sec로 확인되었다. 또한 제동거리는 782cm인 것으로 확인되어 상당히 긴 제동거리를 보였다. 실험에서 확인하였듯이 속도가 빨라짐에 따라 정지 시간이 증가하는 것을 볼 수 있고, 브레이크가 작동을 하지 않기 때문에 차량이 파손될 경우도 발생할 수 있다. 따라서 추가적인 제동 시스템의 설계로 차량의 안정성을 확보해야 한다.

V. 결론

무인 자율 주행 차량을 구성하기 위해 구동부 제어 시스템과 브레이크 제어시스템을 설계하였으며, 종방향 및 횡방향 제어를 위해 기본적 모델링을 통해 시스템을 해석하였다. 시스템 모델링은 여러 차례 실험에 의한 데이터를 바탕으로 해석이 이루어졌으며, 효율적인 주행을 위해 PID 제어를 설계하고 실제 시스템에 적용하였다. 또한 차량의 안정성 주행에 목표를 두어 차량에 문제가 발생되면 비상정지를 할 수 있게 설계를 하였다.

본 연구를 통해 무인 자율 주행 차량의 기본적 설계와 개발을 수행하였으며 부분별 평가를 실행하였다. 실험에 대한 결론은 다음과 같다.

차량의 종방향 및 횡방향 제어를 위해 AC 서보 모터를 사

용하였으며, 포텐쇼미터를 이용하여 초기화를 하고 위치제어를 통하여 조향 및 제동을 하였다. 위치 제어를 통해 만족할 만한 결과를 얻었지만 긴급상황 또는 속도제어를 필요로 할 경우에는 만족할 만한 성능을 얻을 수 없으므로 위치제어와 속도제어를 동시에 해야 한다. PID 제어를 통해 종방향 제어를 하였고, 실험을 통해 이를 검증하여 만족할 만한 결과를 얻었다. 하지만 정상상태 부분에서 외란에 대한 영향이 있어 이를 제거할 수 있는 알고리즘이 필요하다. 조향 응답성 실험은 데이터의 부정확성으로 인해 좋지 않은 결과를 보였다. 차량에 대한 정확한 정보가 필요하다. 시스템 간 통신오류가 발생했을 경우 비상정지 실험은 실제 주행일 경우 주행속도 10km/h에서 약 2.7sec의 응답속도를 보였으며, 제동거리가 4.65m인 것으로 어느 정도의 안정성을 확보하였다. 또한 제동모터가 작동하지 않을 경우 인위적으로 차량을 정지시켜야 한다. 비상정지 모듈을 이용하면 단순히 차량의 전 원만을 차단하게 되기 때문에 제동거리가 속도에 따라 또는 도로 환경에 따라 증가할 수 있어 차량 안전에 문제가 생기게 된다. 이를 보완하기 위해서는 기존에 사용하던 제동 모터 이외의 구동기를 하나 더 설치해야 한다. 이 구동기는 차량 시스템의 제어대상이 아니며, 별도로 제어를 구성해야 차량 시스템이 프로그램 자체에 오류가 발생해도 이에 대응하여 반응할 수 있다.

본 연구를 통해 무인 자율 주행 차량의 주행성능을 확인할 수 있었지만 여러 환경에서의 다양한 실험을 통해 더욱 자세한 시스템 해석이 필요로 하고, 차량 제어를 위해 좀더 강한 제어기의 필요성을 느꼈다. 또한 앞으로 실용차량을 대상으로 한 무인시스템 설계에 대한 가능성을 볼 수 있었다.

참고문헌

- [1] 이구도, 김상우, “지능형교통시스템(ITS)을 위한 첨단차량제어시스템(AVCS),” 제어 · 자동화 · 시스템공학회, 제 6권 제4호, 2000.
- [2] Y. S. Kim and K. S. Hong, “An IMM algorithm for tracking maneuvering vehicles in an adaptive cruise control environment,” *International Journal of Control, Automation, and Systems*, vol. 2, no. 3, pp. 310-318, Sep. 2004.
- [3] U. Hofman, A. Rieder, and E. D. Dickmanns, “Radar and vision data fusion for hybrid adaptive cruise control on highways,” *Machine Vision and Applications*, 14 pp. 42-49, 2003.
- [4] 유혁, 이장호, 김재은, 안이기, “저가형 무인 항공기의 자동비행시스템 개발,” *Journal of Control, Automation, and Systems Engineering*, vol. 10, no. 2, Feb. 2004.
- [5] 문희창, 손영진, 김정하, “다중 GPS를 이용한 무인자동차의 주행 알고리즘 개발,” 제어 · 로봇 · 시스템학회 논문지, 제14권 제1호, Jan. 2008.
- [6] G. F. Franklin, J. David Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 3rd Ed., Addison Wesley, 1995.
- [7] 박보용, 허승진, *Vehicle Dynamics: an Introduction*, 문운당, 2001.
- [8] Tomas D. Gillespie, “Fundamentals of Vehicle Dynamics,” *Society of Automotive Engineers*, pp. 196-208, 1994.
- [9] R. Rajamani, “Vehicle Dynamics and Control,” Springer, 2007.

- [10] 우훈제, “Peer-to-Peer 방식을 이용한 무인 차량 시스템의 통신 아키텍처 개발,” 국민대학교 자동차공학전문대학원 석사학위 논문, 2006.

- [11] W Richard Stevens, *Unix Network Programming Vol. 1: The sockets networking API*, 3rd Ed., Addison-Wesley Professional, 2003.

문 희 창

2001년 선문대학교 기계설계학과 졸업.
2003년 국민대학교 자동차공학전문대학원 석사. 2009년 국민대학교 공학박사.
현재 Univ. of Florida, CIMAR Lab. 연구원. 관심분야는 무인자동차, 소형이동로봇, 카메라를 이용한 장애물 인식, 마이크로 로봇.

크로 로봇.

박 명 옥

2006년 국민대학교 기계자동차공학부 기계공학과 졸업. 2008년 국민대학교 자동차공학전문대학원 석사. 2008년~현재 동 대학원 박사과정 재학 중. 관심분야는 무인자동차, 무인자율차량 시스템의 제어.

김 정 하

1981년 성균관대학교 기계공학과 졸업.
1986년 Univ. of Cincinnati 공학석사. 1990년 Univ. Pennsylvania 공학박사. 1994년~현재 국민대학교 기계자동차공학부 교수. 관심분야는 차량전자제어시스템, 무인자동차.