

안드로이드 스마트폰 제어기반의 개인용 탑승로봇 구현

Development of a Personal Riding Robot Controlled by a Smartphone Based on Android OS

김연균, 김동현*

(Yeongyun Kim¹ and Dong Hun Kim¹)

¹Department of Electrical Engineering, Kyungnam University

Abstract: In this paper, a small, lightweight smartphone-controlled riding robot is developed. Also, in this study, a smartphone with a jog shuttle mode for consideration of user convenience is proposed to make a small, lightweight riding robot. As well, a compass sensor is used to compensate for the mechanical characteristics of motors mounted on the riding robot. The riding robot is controlled by the interface of a drag-based jog shuttle in the smartphone, instead of a mechanical controller. For a personal riding robot, if the smartphone is used as a controller instead of a handle or a pole, it reduces its size, weight, and cost to a great extent. Thus, the riding robot can be used in indoor spaces such as offices for moving or a train or bus station and an airport for scouting, or hospital for disabilities. Experimental results show that the riding robot is easily and conveniently controlled by the proposed smartphone interface based on Android.

Keywords: android, smartphone, compass sensor, riding robot, interface

I. 서론

사회가 발달하고, 많은 도시들이 산업화되어 가면서, 사람들은 이동할 때 좀 더 편리한 이동시스템을 찾고 있다. 집 근처의 가까운 거리, 혹은 빌딩 내 사무실에서 다른 건물 혹은 다른 사무실로의 이동, 혹은 도시 내에서의 인도 혹은 차도에서의 이동을 위해 간편하고 빠르게 이동할 수 있는 이동 시스템을 필요로 한다. 현재의 자전거는 커다란 크기의 불편함과 사람이 이동하지 않는 별도의 도로를 필요로 하므로 이러한 것을 모두 대체해주지는 못한다. 개인 소득이 높아지면 높아질수록 간편하며, 빠르게 이동할 수 있는 이동 시스템을 개인이 소유하며 사용하려고 하는 욕구는 점점 더 높아질 것이다. 가볍고 작은 크기의 이동시스템으로 사람이 탑승하여 작은 공간도 쉽게 움직이며, 계단이나 부정형 지형에서는 들고 다닐 수 있는(portable) 탑승형 이동수단이 있으면 이러한 요구를 만족시킬 수 있다. 따라서, 본 연구에서는 이러한 요구를 만족시키기 위하여 스마트폰제어에 의한 개인용 소형탑승로봇을 개발한다.

최근에 미국 애플사에서 개발된 아이폰은 통신, 영상처리, 로봇관련 연구에서 임베디드 플랫폼으로 사용되어오고 있다[1-5]. 다양한 센서와 통신 모듈로 구성된 아이폰은 여러 분야에서 사용될 수 있다. 별도의 제어기 없이 아이폰으로 로봇을 무선 제어하는 연구들이 최근에 많이 진행되고 있다. [6]의 연구에서는 이동로봇을 무선으로 제어하기 위

해 별도의 하드웨어 제작 없이 아이폰을 사용한다. 탑승형 이동 수단에서 제어를 위해 스마트폰이 사용된 경우는 아직 연구논문으로 검색되지 않는다. 하지만 로봇의 무선제어 기능이 발전하면 점차 모든 곳에 스마트폰 제어의 사용은 점차 늘어날 것이다. 왜냐하면 스마트폰 기기 자체는 많은 센서와 통신장치, 카메라로 구성된 소형의 완벽한 하드웨어이기 때문이다. 본 과제에서는 사용자가 소형이동로봇 위에 올라타서 스마트폰을 사용한 인터페이스로 소형탑승로봇을 제어하며 운용한다. 연구 대상의 이동로봇 플랫폼(platform)은 크기가 노트북 컴퓨터를 펼친 크기 정도이고, 높이는 30cm 이하로 성인이 혼자 들 수 있는 무게로 제한한다.

스마트폰을 사용해 이동로봇을 원격 제어하는 연구는 저자의 연구[7]를 비롯해 국내외에서 최근에 활발하게 이루어지고 있다[8-10]. 하지만 세그웨이 같은 탑승로봇에 스마트폰을 제어기로 직접 사용하는 연구는 아직 많이 있지 않았다. 이것은 사람과 기계와의 인터페이스 방식이 안정되어 있지 않은데서 비롯된다. 저자는 [10]에서 이동로봇을 원격 제어할 때 사용할 수 있는 세 가지 인터페이스 방식(터치형, 가속도센서 반응형, 조그셔틀형)을 제안하고, 사용자 제어에 의한 인터페이스 방식을 분석하였다. 본 연구에서는 이러한 인터페이스를 이용하는 제어시스템을 제작하고, 실제 탑승 가능한 로봇을 H/W로 구현하려고 한다.

이동로봇을 제작하고 구현하는데 있어 이동로봇이 원하는 위치와 방향으로 이동하는 것은 중요하다. 하지만 이동로봇은 이동로봇에 장착된 모터의 기계적 특성과 잘못된 바퀴의 정렬위치(wheel alignment)로 인하여 원하는 위치와 방향으로 이동하지 못한다. 그래서 대부분의 이동로봇 구현 연구에서는 센서의 도움을 받아 모터의 기계적 특성과 바퀴의 정렬위치를 보정한다. 기존의 이동로봇 연구에서는 이

* 책임저자(Corresponding Author)

Manuscript received April 5, 2013 / revised April 23, 2013 / accepted May 14, 2013

김연균: 경남대학교 첨단공학과(grantkyg@gmail.com)

김동현: 경남대학교 전기공학과(dhkim@kyungnam.ac.kr)

※ 본 연구는 2013학년도 경남대학교 학술연구 장려금 지원으로 이루어졌음.

동로봇에 엔코더를 이용하거나 엔코더와 다른센서를 융합하여서 보정하였다[11-15]. 하지만 기존의 이동로봇 플랫폼에 엔코더를 장착하는 것은 기구적으로 어렵고, 엔코더가 장착된 모터는 가격이 비싸다는 단점이 있다. 그래서 본 연구에서는 지자기센서를 이용하여 위의 문제점을 해결하려고 한다.

본 논문은 다음과 같이 구성되어 있다. II 장에서 개인용 탑승 시스템과 제안된 스마트 기반의 탑승로봇의 H/W에 대해 설명한다. III 장에서는 스마트폰과 탑승로봇과의 통신 방법, 인터페이스, 탑승로봇의 H/W 제어방법에 대해 제안한다. IV 장은 사용자가 탑승로봇에 타서 스마트폰으로 탑승로봇을 제어하는 실험 내용과 결과를 보여준다. 마지막으로 V 장에서는 본 논문의 결론과 향후 연구방향에 대해 제시한다.

II. 문제정의와 H/W설계

개인용 탑승 시스템은 미국에서 제작한 세그웨이[16]가 가장 대표적인 상품이다. 세그웨이는 그림 1과 같이 이륜형 역진자 로봇의 형태로, 사람이 폴(pole)을 기울여 제어한다. 휠 자체의 밸런싱을 제어하기 위하여 5개의 자이로 센서와 2개의 기울기 센서를 사용한다. 7개의 센서와 알고리즘의 도움으로 동적 안정도를 확보하여 세그웨이는 두 바퀴로 이동할 수 있다. 하지만 단점으로는 이러한 기능이 세그웨이의 단가를 높여 개인용으로 세그웨이를 구입하여 사용하기에는 많은 부담이 있다. 세그웨이의 방향전환을 위해서는 제어기로 기계적인 폴을 사용한다. 이러한 기계적인 폴은



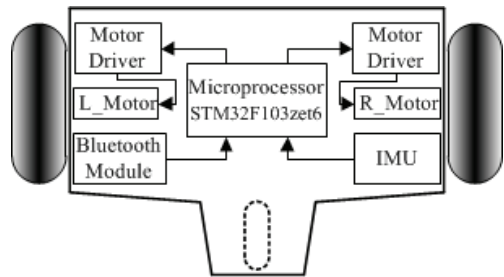
그림 1. 세그웨이.
Fig. 1. Segway.



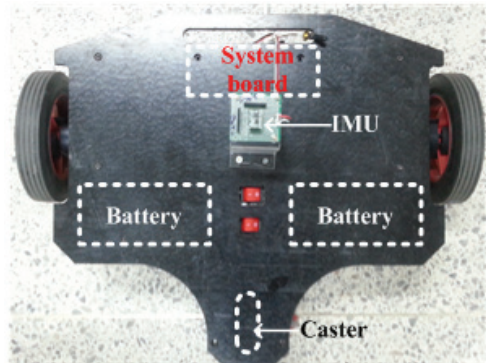
그림 2. 스마트폰 기반의 탑승로봇.
Fig. 2. Riding robot based on smartphone.

세그웨이의 플랫폼 크기와 무게를 증가시켜 혼자서 세그웨이를 계단 등에서 옮기기에는 무리가 있다. 따라서 본 연구에서는 이러한 문제점을 극복하기 위하여 기계적인 제어장치인 폴을 없애고 그림 2와 같이 스마트폰을 사용자가 들고 무선 제어할 수 있는 개인용 탑승 시스템을 개발한다. 스마트폰의 사용은 탑승로봇 플랫폼의 기계, 전기적인 현재 상태를 모니터링 할 수 있고, 사용자편의의 조그셔틀 인터페이스를 사용하여 탑승로봇을 쉽고 편리하게 제어할 수 있다. 그리고 탑승한 사람의 밸런싱을 유지하기 위하여 로봇에 자이로센서를 사용하여 로봇의 이동속도를 제어한다. 로봇이 이동 중에 자이로센서로부터 로봇 몸체 두 방향의 각 속도를 측정하여 로봇의 양 쪽 모터의 속도를 제어한다. 로봇 몸체의 각 속도를 측정함으로써 로봇은 탑승한 사람의 밸런싱 유지와 탑승자의 명령에 따라 로봇의 양 쪽 모터가 구동되고 있는지 알 수 있다. 그림 3은 제안된 탑승로봇의 H/W 구조와 외관을 보여준다.

표 1과 2는 탑승로봇의 크기와 무게 그리고 H/W 구성을 나타낸다. 탑승로봇의 MCU는 32비트 마이크로 컨트롤러인 STM32F103zet6를 장착하여 통신패킷 분석과 모터 제어등과 같이 탑승로봇을 전체적으로 제어하는 역할을 한다. 그리고



(a) The interior of H/W configuration.



(b) The exterior of H/W configuration.

그림 3. 제작된 탑승로봇.

Fig. 3. Implemented riding robot.

표 1. 제안된 탑승로봇의 크기 및 무게.

Table 1. Size and weight of the proposed riding robot.

구분	단위
가로	600 [mm]
세로	482 [mm]
높이	253 [mm]
무게	30 [kg]

표 2. 제안된 탑승로봇의 H/W 구성.

Table 2. H/W configuration of the proposed riding robot.

구분	제품명	사용 목적
MCU	STM32F103zet6	모터제어 및 상황판단
BLDC 모터	BG90-S024300	바퀴 구동
모터 드라이버	SBDM-25A	BLDC 모터 구동
블루투스 모듈	ESD110	안드로이드 스마트폰과 통신
IMU 모듈	EBIMU-9 DOF	방향 및 각속도 측정

스마트폰과 통신을 위하여 블루투스 무선통신 모듈이 장착되었다.

III. 개발환경 구축 및 통신 방법

본 연구에서는 탑승로봇을 제어하기 위하여 안드로이드 OS기반의 스마트폰을 사용한다. 탑승로봇을 제어하기위한 스마트폰의 어플리케이션(application)을 개발하기 위해서는 안드로이드 SDK (Software Development Kit), 소프트웨어 툴인 이클립스(eclipse) 그리고 JDK (Java Developer's Kit)가 필요하다[17]. 안드로이드 SDK, 이클립스, JDK를 기본으로 한 개발환경은 그림 4와 같다. 그림 4와 같이 탑승로봇 제어용 안드로이드 어플리케이션 개발은 윈도우(Windows) 혹은 리눅스(Linux)와 같은 운영체제가 설치되어 있는 컴퓨터에 JDK, SDK, 그리고 프로그래밍 통합개발환경인 이클립스가 설치되어있는 개발환경에서 개발한다. 탑승로봇 제어 안드로이드 어플리케이션은 이클립스에서 프로젝트를 생성한 후, SDK에 포함되어 있는 터치(touch), 블루투스(bluetooth), 그리고 그래픽(graphics)과 같은 탑승로봇 제어와 관련된 API (Application Programming Interface)들을 사용하여 코드를 작성하고 컴파일(compile)한다. 컴파일을 통하여 생성된 실행 파일은 데이터 케이블을 이용하여 스마트폰으로 전송한다. 실행파일인 탑승로봇 제어용 어플리케이션을 실행하여 스마트폰과 탑승로봇 간의 통신은 블루투스 무선통신을 사용하여 스마트폰에서 탑승로봇에 명령패킷을 주기적으로 전송한다. 안드로이드 SDK, 이클립스, JDK를 이용한 안드로이드 어플리케이션 개발환경 구축은 구글의 안드로이드 개발자 사이트[18]에 자세하게 설명되어 있다.

1. 터치 센서 기반의 조그셔틀 제어

탑승로봇에 서있는 사용자가 스마트폰으로 로봇을 제어할 때에는 스마트폰의 화면에 표시되어진 조그셔틀 스위치에 손가락을 놓고, 원하는 방향으로 손가락을 움직이면 스마트폰은 블루투스 무선통신을 통하여 로봇에 제어명령을 전달한다. 아이폰의 경우 블루투스 통신을 사용하기 위해서는 불법으로 소프트웨어 개조를 해야 하지만 안드로이드 기반 스마트폰에서는 이러한 개조 없이 블루투스 무선통신을 사용할 수 있다. 로봇은 스마트폰으로부터 수신받은 명령 패킷을 분석하여 로봇의 양쪽 모터를 구동시킨다.

[19]에서 사용자는 로봇을 제어 할 때 손 터치 인터페이스가 가장 적합하게 사용될 수 있다고 연구되었다. 따라서 이를 바탕으로 한손으로 스마트폰을 쥐고 원하는 방향으로 조그셔틀을 이동하는 인터페이스를 제안 한다. 그림 5는 터치 센서 기반의 조그 셔틀을 보여준다. 사람의 손가락에 의

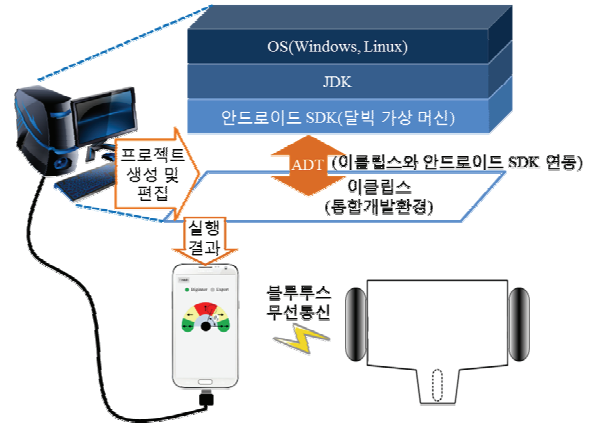


그림 4. 탑승로봇 제어용 안드로이드 어플리케이션 개발환경.
Fig. 4. Android application development environment for controlling a riding robot.

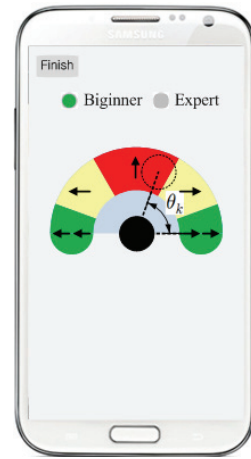


그림 5. 터치 센서를 이용한 인터페이스.
Fig. 5. Interface based on a touch sensor.

해 터치된 위치는 검정색으로 채워진 원에서 빈 점선 원으로 이동하게 된다. 스마트폰은 일정한 시간 간격으로 이동된 터치 위치와 원점사이의 각도 θ_k 를 측정하여 다섯 영역으로 나눈다. 다섯 영역은 각각의 영역에 대한 명령이 정해져 있다. 스마트폰은 터치된 영역의 명령을 포함한 통신 패킷을 탑승로봇에 블루투스 무선통신을 통하여 0.1초 간격으로 송신한다.

스마트폰에서 제공하는 선택된 인터페이스를 통해 원하는 위치로 이동한 후 스마트폰에서 손가락을 떼면 바퀴가 움직이지 않아 다른 사람들이 탑승하여 구동하는 것을 막을 수 있다. 또한, 스마트폰을 제어기로 사용함으로써 얻어지는 플랫폼 공간의 축소화와 경량화의 장점 이외에도 이동 중에 기록된 GPS 좌표와 마지막 시점의 위치가 스마트폰에 기록되어 사용자에게 정보를 제공해준다.

본 연구는 스마트폰에서 탑승로봇에 일정한 시간 간격으로 명령을 송신하여 탑승로봇을 제어한다. 탑승로봇을 제어하는 통신패킷은 그림 6과 같이 구성된다. 여기서 사각형한 개는 1바이트를 의미한다. 첫 번째 바이트는 “#” 또는 “\$” 또는 “!” 문자로 동작모드를 의미한다. “#”은 숙련자모

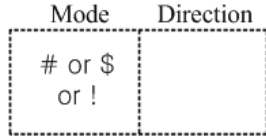


그림 6. 통신패킷.
Fig. 6. Communication packet.

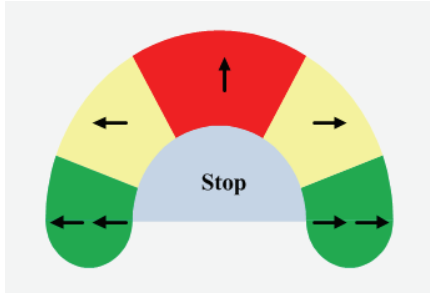


그림 7. 명령영역.
Fig. 7. Range of command.

드이고, “\$”는 초보자모드이고, 그리고 “!”는 정지모드이다. 초보자모드와 숙련자모드는 탑승로봇의 이동속도가 다르다. 나머지 1 바이트는 터치 각도인 θ_k 를 이용하여 다섯 영역으로 나누어진 명령을 의미한다.

그림 7은 θ_k 에 의해 나누어진 명령영역을 나타낸다. 통신패킷의 명령부분은 탑승자가 스마트폰의 스크린위에 터치한 위치와 원점사이의 각도인 θ_k 를 측정하여 $0(\rightarrow)$ 에서 4까지 1 바이트의 정수로 나타낸다. 첫 번째 영역의 명령인 $0(\rightarrow)$ 은 $0^\circ \sim 20^\circ$ 사이의 범위이고, 탑승로봇을 우측으로 급하게 회전하며 전진시키는 명령이다. 탑승로봇이 $0(\rightarrow)$ 명령을 처음 수신하였을 때 정지상태이면 탑승로봇은 제자리에서 우측방향으로 회전한다. 두 번째 영역의 명령인 $1(\rightarrow)$ 은 $21^\circ \sim 60^\circ$ 사이의 범위이고, 탑승로봇을 우측으로 전진시키는 명령이다. 세 번째 영역의 명령인 $2(\uparrow)$ 은 $61^\circ \sim 120^\circ$ 사이의 범위이고, 탑승로봇을 전진 시키는 명령이다. 네 번째 영역의 명령인 $3(\leftarrow)$ 은 $121^\circ \sim 160^\circ$ 사이의 범위이고, 탑승로봇을 좌측으로 전진시키는 명령이다. 다섯 번째 영역의 명령인 $4(\leftarrow\leftarrow)$ 은 $161^\circ \sim 180^\circ$ 사이의 범위이고, 탑승로봇을 좌측으로 급하게 회전하며 전진시키는 명령이다. 탑승로봇이 $4(\leftarrow\leftarrow)$ 명령을 처음 수신하였을 때 정지상태이면 탑승로봇은 제자리에서 좌측방향으로 회전한다.

사용자가 스마트폰 화면에서 다섯 가지의 방향제어 영역이 아닌 정지(Stop)영역을 손가락으로 터치하면 스마트폰은 탑승로봇에 강제정지명령인 “!”를 송신한다. 강제정지명령을 수신한 탑승로봇은 모터에 전압이 입력되어, 즉 토크가 인가되어 정지하게 된다. 그리고 사용자가 탑승로봇에 탑승하여 스마트폰 조작시 스마트폰을 떨어뜨리는 경우도 있을 수 있다. 즉 사용자가 스마트폰의 화면에서 손가락이 떼어진 경우에는 스마트폰이 탑승로봇에 정지명령인 “!”를 송신한다. 정지명령을 수신한 탑승로봇은 모터에 전압을 인가하지 않아 자동적으로 정지하게 된다.

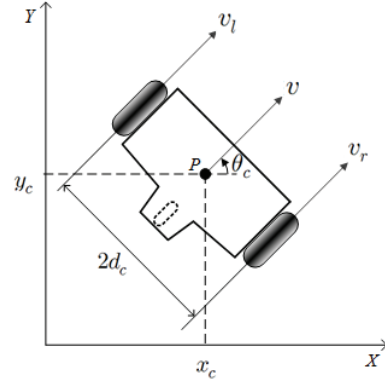


그림 8. 차등 구동 로봇 모델.
Fig. 8. Differential drive robot model.

2. 탑승로봇 제어

본 연구에서 구현된 탑승로봇은 그림 8과 같이 차등 구동 로봇 모델로 구성되어져 있다. 차등 구동 로봇모델은 세 개의 벡터로 표현 된다. 여기서 세 개의 벡터는 현재 위치와 기준 프레임에 대한 방위각이다. 점에서 로봇의 바퀴가 지면에 대해 미끄러짐이 없다고 가정할 때 수식은 다음과 같다[20].

$$\Delta x_c \sin \theta_c - \Delta y_c \cos \theta_c = 0 \tag{1}$$

여기서 P 는 양쪽 바퀴사이의 중심이고, θ_c 는 탑승로봇의 회전각을 의미 한다. 이 자연적 제한은 적분 불가능하다. 즉 로봇 모델은 논홀로노믹 제한이 있다. 다시 말해 로봇은 곧 바로 한번에 x 또는 y 방향으로 이동할 수 없다. 로봇의 선속도 v 와 각속도 ω 는 다음과 같다[20].

$$\Delta x_c \cos \theta_c + \Delta y_c \sin \theta_c = v, \Delta \theta_c = \omega \tag{2}$$

따라서 정리하면 다음과 같다.

$$\begin{bmatrix} \Delta x_c \\ \Delta y_c \\ \Delta \theta_c \end{bmatrix} = \begin{bmatrix} \cos \theta_c & 0 \\ \sin \theta_c & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{3}$$

v 와 ω 는 다음과 같이 구해진다.

$$v = \frac{v_r + v_l}{2}, \omega = \frac{v_r - v_l}{2d_c} \tag{4}$$

여기서 v_l 과 v_r 은 각각 왼쪽과 오른쪽 모터의 PWM (Pulse Width Model)값이고 $2d_c$ 는 탑승로봇의 양쪽 바퀴 사이의 수평거리이다.

스마트폰에 의해 제어되는 탑승로봇은 스마트폰으로부터 수신한 통신패킷에 포함된 명령에 의해 탑승로봇에 장착된 양쪽 모터를 제어한다. 탑승로봇의 양쪽 모터를 센서의 도움없이 명령에 따라 고정 값으로 제어하면, 탑승로봇은 양쪽모터의 기계적인 특성 차이와 장착의 부정확성에 의해 탑승자가 원하는 방향으로 이동하지 못한다. 본 연구에서는 양쪽 모터의 속도제어를 위하여 탑승로봇에 IMU (Inertial Measurement Unit)를 장착하였다. IMU는 탑승로봇의 방향각의 변화값인 $\Delta \theta_c$ 와 몸체의 기울어진 각도 변화값인 $\Delta \theta_v$ 를

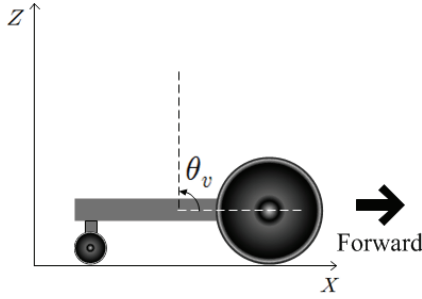


그림 9. 탑승로봇의 측면도.

Fig. 9. The side of a riding robot.

측정한다. 그림 9는 탑승로봇의 측면도를 보여준다. 탑승로봇은 IMU로부터 측정되어진 $\Delta\theta_c$ 와 $\Delta\theta_v$ 를 비교하여 양쪽 모터의 속도인 v_l 과 v_r 을 정하여 탑승자가 원하는 방향으로 이동한다. v_l 과 v_r 은 다음과 같이 주어진다.

$$\begin{aligned} v_l &= s_m + k_p(\Delta\theta_c^d - \Delta\theta_c) \\ v_r &= s_m + k_p(\Delta\theta_c^d + \Delta\theta_c) \\ s_m &= v_f - k_v(\Delta\theta_v^d + \Delta\theta_v) \end{aligned} \quad (5)$$

여기서 s_m 은 로봇몸체의 앞으로 기울어진 정도에 따른 상수의 속도값이고 $k_p(\Delta\theta_c^d \pm \Delta\theta_c)$ 는 탑승로봇의 방향각에 대한 비례제어기이다. k_p 는 로봇의 방향에 대한 비례상수이다. $\Delta\theta_c^d$ 는 각각의 명령에 따른 원하는 방향 각도변화 값이다.

3.1절에서 설명되어진 탑승자의 손가락 터치에 의한 각각의 방향명령에 따른 $\Delta\theta_c^d$ 는 표 3과 같다. $\Delta\theta_c$ 는 탑승로봇이 향하고 있는 각도 변화 값이다. 그리고 v_f 는 로봇의 고정된 속도이고, k_v 는 로봇 몸체의 앞으로 기울어진 각도에 대한 비례상수이다. $\Delta\theta_v^d$ 는 로봇몸체의 수평을 기준으로 원하는 각도 변화이고, 0으로 설정하였다. $\Delta\theta_v$ 는 로봇몸체의 수평을 기준으로 변화한 각도이다.

탑승로봇은 양쪽 모터를 이용하여 이동방향을 제어하는 것 뿐만아니라 로봇 몸체의 수평균형을 유지함으로써 탑승자에게 안정성을 보장하여야 한다. 본 연구에서는 탑승자의 안정성을 보장하기 위하여 자이로센서를 장착하여 XY평면을 기준으로 로봇 몸체의 각속도를 일정한 시간간격으로 측정한다. 로봇 몸체의 수평을 기준으로 측정된 각속도는 $\Delta\theta_v$ 이다. 탑승로봇의 뒤쪽에 장착된 캐스터 휠은 탑승로봇이 뒤쪽으로 기울어지는 것을 방지한다. 탑승로봇은 앞쪽으로 기울어진 각도 변화값인 $\Delta\theta_v$ 를 30[msec] 간격으로 측정하여 이동속도를 제어한다. 탑승로봇이 앞쪽으로 기울어 질

표 3. 명령에 따른 $\Delta\theta_c^d$.
Table 3. $\Delta\theta_c^d$ of the command.

동작	우회전	우전진	전진	좌전진	좌회전
명령	0	1	2	3	4
기호	→→	→	↑	←	←←
$\Delta\theta_c^d$	-15°	-5°	0°	5°	15°

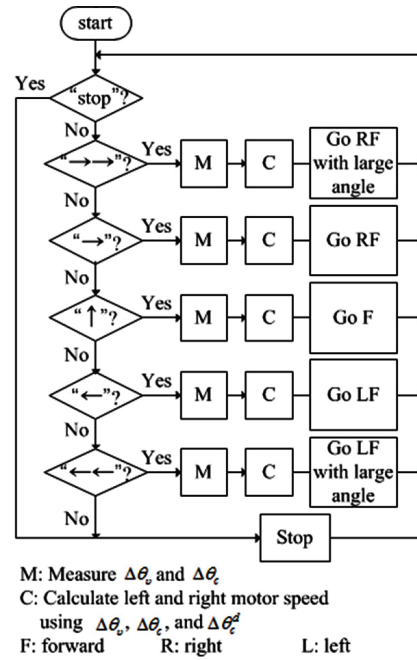


그림 10. 탑승로봇의 알고리즘.

Fig. 10. Algorithm of the riding robot.

경우 $\Delta\theta_v$ 는 작아진다. $\Delta\theta_v$ 는 음수가 되어 수식(5)에 의해 모터의 속도는 증가하게 되어서 탑승로봇이 수평을 유지할 수 있게 해준다.

3. 알고리즘

본 연구에서 제작된 탑승로봇은 그림 10과 같은 알고리즘으로 구동된다. 탑승자가 III 장에서 설명되어진 것과 같이 터치기반의 조그서를 인터페이스를 이용하여 원하는 방향으로 탑승로봇에 명령을 블루투스 무선통신을 통하여 송신한다. 탑승로봇은 탑승자의 스마트폰에서 송신된 명령패킷을 파싱하여 그림 10의 알고리즘과 같은 과정에 따라 동작한다. 탑승자가 정지명령이 아닌 “→”, “←”, “↑”, “↔”, 그리고 “↔↔” 명령을 송신하면, 탑승로봇은 탑승로봇에 장착된 IMU를 통하여 $\Delta\theta_v$ 와 $\Delta\theta_c$ 를 측정한다. 측정된 $\Delta\theta_v$ 는 탑승로봇의 이동속도를 정하고, $\Delta\theta_c$ 는 표 3과 같이 정해진 원하는 각속도와 비교하여 왼쪽과 오른쪽 모터의 속도차이를 주어서 원하는 방향으로 이동하게 된다. 그리고, 탑승로봇이 탑승자의 명령을 수신하여 좌·우측의 모터를 구동할 때, 탑승자의 균형유지를 위하여 좌·우측의 모터속도는 여덟 단계로 나누어서 증가하고, 정지할 때에도 여덟 단계로 나누어서 정지한다.

IV. 실험 결과

탑승로봇은 IMU로부터 측정된 $\Delta\theta_c$ 를 이용함으로써 탑승로봇의 기계적 특성으로 인한 부정확한 이동을 보정한다. 그림 11은 탑승로봇이 전진할 때, IMU로부터 측정된 $\Delta\theta_c$ 를 이용하지 않은 경우 그림 11(a)와 이용한 경우 그림 10(b)의 이동궤적을 보여준다. 탑승로봇은 III 장에서 설명된 스마트폰의 조그서를 인터페이스를 이용한 탑승자의 명령에 의해 이동하였다. 그림 11은 지자기센서로부터 측정된



(a) Go forward without a compass sensor.



(b) Go forward with a compass sensor.

그림 11. 탑승로봇의 직진 이동궤적.

Fig. 11. Trajectory of going forward.



(a) Trajectory of a beginner mode.



(b) Trajectory of an expert mode.

그림 12. 탑승로봇의 S자형 이동궤적.

Fig. 12. "S" trajectory of riding robot.

$\Delta\theta_c$ 를 이용하였을 때, 탑승로봇의 기계적인 특성으로 인한 부정확한 이동 궤적을 보정하는 것을 보여준다.

그림 12는 그림 5의 화면에서 비숙련자(beginner), 숙련자(expert) 모드로 각각 실험한 탑승로봇의 이동궤적을 보여준다. 그림 12에서는 S자형 궤적 실험을 위해 5 [m] 거리에 장애물 두 개를 설치하였고, 동일 서터 시간을 가지고 촬영

되었다. 그림 11과 12의 실험을 통하여 본 논문에서 제안된 스마트폰을 이용한 탑승로봇이 편리하고 효과적으로 제어되고 있음을 알 수 있다.

V. 결론

본 논문에서는 스마트폰을 사용하여 제어 가능한 소형, 경량화된 탑승로봇을 구현한다. 개인용 탑승로봇을 소형화하기 위하여 사용자편의의 조그 셔틀 인터페이스를 통해 기계적인 제어기 없이 스마트폰으로 탑승로봇을 제어한다. 본 연구에서는 제어기로 스마트폰을 사용하기 때문에 기계적 제어장치인 핸들이나 폴의 하드웨어를 없앨 수 있다. 이것은 탑승로봇의 크기, 무게, 비용을 상당히 줄여준다. 따라서 제안된 탑승로봇은 사무실과 같은 건물안 실내, 지하철역, 평평한 바닥을 가지는 실외 공간에서 사용될 수 있다. 실험의 결과로서 제안된 안드로이드 스마트폰 기반의 인터페이스는 쉽고 편리하게 탑승로봇의 제어를 가능하게 해주었다. 향후 연구계획은 스마트폰에서 지원하는 구글맵과 GPS를 사용하여 실외에서 사용자가 개입하여 제어가능한 반자율 주행모드를 개발하는 것이다.

REFERENCES

- [1] N. Kubota and W. Sato, "Robot design support system based on interactive evolutionary computation using boltzmann selection," *2010 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1-8, 2010.
- [2] B. R. Geltz, J. A. Berlier, and J. M. McCollum, "Using the iPhone and iPod Touch for remote sensor control and data acquisition," *Proc. of the 2010 IEEE Southeast Conference*, pp. 9-12, 2010.
- [3] L. YeMin and Z. Li, "Research and application of the EAN-13 barcode recognition on iphone," *2010 International Conference on Future Information Technology and Management Engineering (FITME)*, vol. 2, pp. 92-95, 2010.
- [4] S. Chemlal, S. Colberg, M. Satin-Smith, E. Gyuricsko, T. Hubbard, M. W. Scerbo, and F. D. McKenzie, "Blood glucose individualized prediction for type 2 diabetes using iPhone application," *37th Annual Northeast Bioengineering Conf. (NEBEC)*, pp. 1-2, 2011.
- [5] H. K. Y. Chan, Z. Huiru, W. Haiying, R. Gawley, Y. Mingjing, and R. Sterritt, "Feasibility study on iPhone accelerometer for gait detection," *5th International Conf. on Pervasive Computing Technologies for Healthcare (Pervas-iveHealth)*, pp. 184-187, 2011.
- [6] F. Chandra, N. Kok Yew, and K. Boon How, "Development of a convenient wireless control of an autonomous vehicle using apple iOS SDK," *TENCON 2011 - 2011 IEEE Region 10 Conference*, pp. 1025-1029, 2011.
- [7] H. Jung, Y. Kim, and D. H. Kim, "Communication quality analysis for remote control of a differential drive

- robot based on iphone interface,” *Communication in Computer and Information Science (Indexed by SCOPUS)*, vol. 206, pp. 278-285, 2011.
- [8] H. Y. Jung, C. S. Lee, Y. H. Seo, and T. K. Yang, “Touch-based moving trajectory generation and data acquisition of a mobile robot using a smart phone,” *Journal of Korea lasbiburte of Information & Telecom (in Korean)*, vol. 10, no. 3, pp. 98-102, 2011.
- [9] J. J. Park, G. S. Choi, C. H. Chun, I. K. Park, and J. J. Kang, “Remote navigation and monitoring system for mobile robot using smart phone,” *Journal of the institute of Internet, Broadcasting and Communication (in Korean)*, vol. 11, no. 6, pp. 207-214, 2011.
- [10] Y. H. Jeon and H. Ahn, “Smart-phone based interface for mobile robot control,” *Conference of the Institute of Electronics Engineers of Korea (in Korean)*, vol. 33, no. 1, pp. 171-181, 2010.
- [11] J. Borenstein and Y. Koren, “Motion control analysis of a mobile robot,” *ASME Journal of Dynamic Systems Measurement and Control*, vol. 109, pp. 73-79, 1987.
- [12] J. Borenstein and L. Feng, “Gyrodometry: A new method for combining data from gyros and odometry in mobile robots,” *Proc. of the International Conference on Robotics and Automation*, vol. 1, pp. 423-428, 1996.
- [13] S. Lee and J. B. Song, “Robust mobile robot localization using optical flow sensors and encoders,” *Proc. of International Conference on Robotics and Automation*, vol. 1, pp. 1039-1044, 2004.
- [14] J. Y. Choi and S. G. Kim, “Study on the localization improvement of the dead reckoning using the INS calibrated by the fusion sensor network information,” *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 8, pp. 744-749, 2012.
- [15] S. H. Bae and B. K. Kim, “An efficient outdoor localization method using multi-sensor fusion for car-like robots,” *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 10, pp. 995-1005, 2011.
- [16] http://en.wikipedia.org/wiki/Segway_PT
- [17] Y. S. Chang, K. Kim, and N. H. Sung, *Step by Step Android Application Development (in Korean)*, INFINITYBOOKS, 2011.
- [18] <http://developer.android.com/training/index.html>
- [19] S. Chemlal, S. Colberg, M. Satin-Smith, E. Gyuricsko, T. Hubbard, M. W. Scerbo, F. D. R. McKenzie, B. J. Pierre, and O. Pierre-Yves, “A comparison of three interfaces using handheld devices to intuitively drive and show objects to a social robot: the impact of underlying metaphors,” *The 18th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 1066-1072, 2009.

- [20] D. H. Kim, “Self-organization of unicycle swarm robots based on a modified particle swarm framework,” *International Journal of Control, Automation and Systems*, vol. 8, no. 3, pp. 622-629, Jun. 2010.



김 연 균

2009년 경남대학교 전기전자 공학부 졸업. 2010년~2011년 동 대학원 첨단 공학과 석사 졸업. 2012년~현재 동 대학원 첨단공학과 박사 과정. 관심분야는 지능제어, 스웩로보틱스, 신호처리.



김 동 현

2001년 한양대학교 전기공학과(공학박사). 2001년~2003년 미국 듀크 대학교 연구원. 2003년~2004년 미국 보스턴 대학교 연구원. 2004년~2005년 일본 동경대학교 박사 후 과정. 2012년~2013년 영국 에식스대학교 방문교수. 2005년~현재 경남대학교 전기공학과 부교수. 관심분야는 스웩 시스템, 이동로봇 경로계획, 지능 로봇 제어, 탑승로봇, 국방로봇, 비선형 적응제어.