

다중 스마트폰을 이용한 이동로봇 대상의 레이싱게임 구현

Development of a Racing Game for Mobile Robots using Multiple Smartphones

정 하 민, 김 동 헌*

(Hahmin Jung¹ and Dong Hun Kim²)

¹Department of Advanced Engineering, Kyungnam University

²Department of Electrical Engineering, Kyungnam University

Abstract: This paper presents the development of a racing game by controlling multiple smartphones based on ad hoc communication. The proposed racing game by the smartphone-based control interface does not require any specific game console or remote controller. Thus, any multiple users who have smartphones are able to play the game in a monitor at the same time. In addition, the developed game is applicable to actual mobile robots in cases where the positions of all robots are measured, since its game unit is a mobile robot model with a differential drive. An experimental result shows that the racing game in a PC can be realized by the proposed communication interface through four iPhones based on acceleration sensors.

Keywords: smartphone, racing game, ad hoc communication, mobile robot, iPhone

I. 서론

일반적으로, 스마트폰을 포함하는 모바일 단말기는 얇고 가벼워서 휴대하기 용이하며 다양한 용도의 어플리케이션을 설치하여 사용자에게 유용하고 편리한 통신생활을 가능하게 한다. 최근의 스마트폰은 이동통신 단말기로서의 기능뿐만 아니라, 인터넷 검색, 메일 송수신, 동영상과 사진의 촬영 및 편집, 음악 감상 및 게임 등의 다양한 기능을 제공하고 있다. 또한 스마트폰은 휴대하기 용이할 뿐 아니라 카메라, GPS, 블루투스, 무선통신, 가속도센서 및 지자기 센서와 같은 다양한 하드웨어가 탑재되고 있으며, 어플리케이션을 설치해서 사용자의 필요에 따라 활용할 수 있는 장점이 있다. 최근에는 스마트폰의 하드웨어 및 소프트웨어의 성능 향상으로 인하여, 스마트폰을 게임, 산업용, 군사용, 의료용 등의 다양한 분야에서 사용하고자 하는 연구 및 개발도 지속적으로 이루어지고 있다[1-4].

휴대전화 시장에서 게임을 개발하는 과정은 휴대폰 제조사의 개발자에 의해 개발되었던 것과 비교해보면 현재 개발하는 방법은 많은 부분에서 달라졌다. 2009년 아이폰(iPhone)의 도입은 스마트폰이라는 시장을 만들게 하였고 많은 게임 개발자들이 생겨났다. 스마트폰의 앱 개발은 휴대폰 제조사에서 SDK (Software Development Kit)를 제공하기 때문에 쉽고 빠르게 앱을 개발하는 장점이 있다. 스마트폰은

기존의 휴대용 게임기를 대신할 정도로 빠른 연산처리와 다양한 장르의 게임을 제공한다[6]. 스마트폰은 사람과 소프트웨어가 만나는 중간 다리 역할을 하기 때문에 스마트폰에서의 게임들은 스마트폰에 장착된 다양한 센서들을 이용하여 편리하고 쉬운 조작성을 지원 받을 수 있다[7]. 이러한 게임들은 스마트폰의 화면을 보면서 일인용으로 플레이하기 때문에 쉽게 흥미를 잃을 수 있다. 이를 보완하기 위해 WiFi 통신을 이용하여 다수 또는 소수 유저간의 플레이를 관리하는 알고리즘들이 개발되었다[8,9]. 또한 스마트폰보다 넓은 화면에서 게임을 즐길 수 있게 하기 위하여 스마트폰을 무선 조종기로 사용하고 PC의 유닛을 운영 하는 게임도 점차 구현되고 있다[10].

게임뿐만 아니라 실제 하드웨어를 구동하기 위한 조종기로도 스마트폰은 사용된다[11,12]. 스마트폰을 무선제어기로 사용하면 별도로 다른 제어를 구매할 필요가 없다. 또한 아이폰과 같은 휴대전화기는 화면에 현재의 제어상황과 같은 많은 정보를 사용자에게 제공할 수 있으므로 무선제어기로도 스마트폰은 사용자에게 많은 장점을 제공해 줄 수 있다.

한편 하나의 스마트폰을 이용해서 이동로봇 또는 특정 컴퓨팅 장치를 제어하고자 하는 연구 및 개발은 지속적으로 이루어지고 있으나, 복수의 스마트폰을 이용해서 이동로봇 또는 컴퓨팅 장치와 무선 통신을 하여 원격으로 제어하고자 하는 연구 및 개발은 아직 미진하여 사용자에게 많은 혜택을 주고 있지는 않다. 즉, 종래에는 하나의 컴퓨팅 장치가 있는 경우, 복수의 스마트폰이 무선 통신을 통해서 컴퓨팅 장치에 연결된 상태에서 실시간으로 자신의 유닛만을 제어하거나, 컴퓨팅 장치에 설치된 특정 프로그램들을 각 스마트폰이 독립적으로 실행할 수 없었으며, 하나의 컴퓨팅 장치에서는 하나의 스마트폰만 연결된 상태에서 데이터의

* 책임저자(Corresponding Author)

Manuscript received April 12, 2013 / revised August 19, 2013 / accepted September 2, 2013

정하민: 경남대학교 첨단공학과(zerofull@naver.com)

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

※ 본 연구는 2013학년도 경남대학교 학술연구 장려금 지원으로 이루어졌으며, 이 논문은 2013 제 28회 ICROS 학술대회에 초안[18]이 발표되었음.

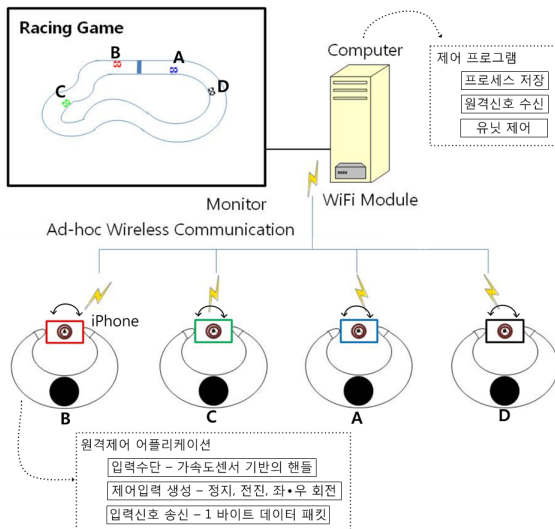


그림 1. 한 대의 PC와 다중 스마트폰과의 통신 인터페이스.
 Fig. 1. Communication interface between a computer and multiple smartphones.

전송 또는 제어가 이루어졌다.

저자는 스마트폰을 이용해 모니터에 있는 가상의 이동로봇을 제어하는 시뮬레이터를 연구하였다[13]. 이 연구를 통하여 애드혹 무선통신이 PC와 아이폰의 통신에 적절하게 이용될 수 있음을 보여주었다. 이 연구에서는 차량의 핸들을 모방하기 위해 아이폰의 가속도 센서를 이용하여 차등구동 로봇 모델을 아이폰으로 제어하였다. [14]에서 저자는 아이폰을 원격 제어기로 사용하기 위한 세 종류의 사용자 인터페이스를 제안하였다. 그리고 [15]에서는 여러 대의 스마트폰에서 최적의 통신속도와 데이터 패킷을 설계하는 연구를 하였다.

본 논문은 저자의 연구[15]를 확장한 것으로 이동 로봇 모델을 대상으로 실제로 사용가능한 레이싱 게임을 구체적으로 구현한 연구이다. 그림 1은 구현하려는 게임 시스템을 보여준다. 그림 1에서 네 명의 유저는 각각의 스마트폰을 핸들처럼 잡고 기울여서 하나의 모니터에 있는 각각의 모바일 로봇을 움직인다. 제안된 게임은 일대다(one to multi)의 애드혹 무선통신을 기반으로 스마트폰에 내장된 기울기 센서를 제어 인터페이스로 이용하는 것이며, 모든 사용자의 모바일(mobile) 로봇이 결승점으로 돌아오면 레이싱 게임은 종료된다.

본 논문은 다음과 같이 구성되어 있다. II 장에서는 아이폰 인터페이스와 전송패킷 그리고 모바일 로봇을 어떻게 게임에 적용했는지 설명한다. III 장에서는 게임속 가상 모바일 로봇을 제어하는 방법과 게임개발에 대한 내용을 다룬다. IV 장에서는 II, III 장에서 제안된 방법을 통하여 하나의 모니터 화면에서 네 명의 사용자가 자신의 모바일 로봇을 제어하는 레이싱 게임을 제안한다. 마지막으로, V 장에는 본 연구의 결론을 제시한다.

II. 제안된 게임 인터페이스와 통신패킷 설정

본 연구에서는 스마트폰으로 iOS를 기반으로 동작하는

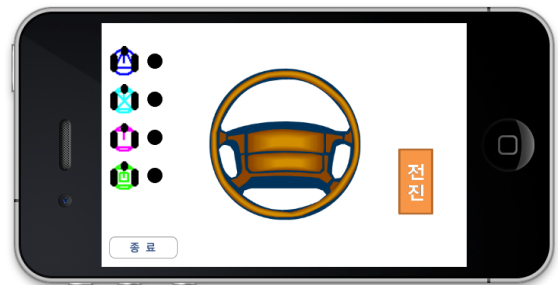


그림 2. 터치 센서와 가속도 센서를 이용한 인터페이스.
 Fig. 2. Interface using touch and acceleration sensor.

아이폰을 게임의 조종기로 사용한다. 본 장은 게임상의 모바일 로봇을 움직이는 아이폰의 인터페이스와 통신 패킷에 대해 설명한다.

1. 게임 인터페이스

본 논문에서 사용하는 아이폰 제어 인터페이스는 저자가 [14]에서 제안한 인터페이스 방식 중 기울기모드에 기반한 인터페이스이다. 그림 2는 터치 센서와 가속도 센서를 기반으로 제안된 인터페이스를 보여준다. 레이싱 게임을 위한 인터페이스는 핸들과 두 개의 버튼 그리고 네 개의 로봇 선택 버튼으로 구성된다. 네 개의 버튼을 이용해 사용자는 게임화면의 로봇 중에서 마음에 드는 로봇을 선택 할 수 있다. 그림 2에서 핸들모양의 그림은 아이폰의 기울어진 정도를 반영하여 회전하고, 종료 버튼은 앱(application)의 종료를 위해 사용된다.

아이폰에서 제어 화면은 우선적으로 무선 조종 인터페이스를 위한 이미지 작성이 필요하고, 그리고 나서 해당 이미지를 아이폰 전면부에 배치함으로써 화면이 구성된다. 아이폰에서 게임 인터페이스는 다음과 같은 방법을 통해 동작된다.

- a) 아이폰의 가속도 센서값을 통신 샘플링 시간보다 빠르게 설정하여 측정하고 이에 알맞게 핸들을 회전시킨다.
- b) 전진 버튼이 눌러졌는지 인터럽트를 확인하고 제어하는 로봇의 색을 체크한다.
- c) 버튼과 핸들의 기울기를 판단하여 이에 알맞은 패킷을 만든다.
- d) 만들어진 패킷은 논문 [15]의 내용처럼 샘플링 시간을 0.1초로 설정하여 전송한다.
- e) 수신된 데이터 패킷은 상위비트를 점검하여 아이폰 번호를 확인하고, 하위비트에 맞게 게임 속 모바일 로봇을 동작 시킨다.

2. 통신 패킷 설정

각각의 아이폰은 PC에 있는 자신의 모바일 로봇을 제어할 때 사용된다. 아이폰과 PC는 Wi-Fi 무선통신으로 연결된다. 무선 통신을 통한 로봇은 아이폰 데이터에 의해 신속하게 움직여야 한다. 하지만 PC에 접속하는 아이폰 수가 많아지면 많아질수록 한 대의 PC로 처리해야 할 무선통신 데이터와 계산시간은 점차 많아진다. 따라서, 연결된 아이폰 수에 맞는 최소한의 길이를 가지는 패킷을 만들어야한다.

본 연구는 그림 1과 같이 한대의 마스터(master)와 다중

표 1. 패킷 정의.

Table 1. Definition of packet.

상위4비트	하위4비트	
아이폰 번호	정지	1000
	전진	1100
	좌회전	0100
	우회전	0110

슬레이브(slave)와의 무선 통신을 위하여 애드혹 통신을 이용한다. 아이폰은 모니터 상에서 자신의 모바일 로봇을 이동시킬 수 있다. 로봇을 이동시키기 위해 아이폰에서 전송되는 명령데이터는 다음과 같이 정의 된다. PC에서 아이폰을 구별하기 위해 아이폰마다 자신의 ID가 필요하고 로봇의 움직임에 관련된 데이터가 필요하다. 로봇을 움직이기 위해 아이폰에서 전송하는 명령은 로봇의 전진, 좌측 이동, 정지명령이다. 이러한 아이폰의 ID와 모바일 로봇의 이동명령을 1바이트(byte)의 데이터 패킷으로 표현한다. 표 1은 아이폰에서 전송하는 데이터 패킷을 보여준다. 다수의 아이폰을 이용하기 위하여 상위 4비트는 아이폰의 ID로 설정되고, 하위 4비트는 로봇의 이동 명령으로 설정된다. 데이터 패킷은 상위 4비트를 확인하는 0xF0 상위 마스크와 하위 4비트를 확인하는 하위 마스크 0x0F로 구성된다.

III. 게임 개발

본 장에서는 게임에서 사용되는 모바일 로봇의 제어방법과 게임개발에 대한 내용을 다룬다.

1. 가상 모바일 로봇 제어

모바일 로봇을 이용한 레이싱 게임을 구현하기 위해 전역좌표계 X, Y를 이용하는 그림 3의 차등 구동 로봇을 사용한다. 그림 3에서 q 점은 구동부와 d 만큼 떨어진 곳에 위치한 캐스터이고 궤적제어를 이용하기 위해 사용된다. 로봇의 위치는 $P = [x, y, \phi]$ 로 표현하고 각각 x 축, y 축 좌표와 로봇의 회전각을 의미한다. 그림 3의 운동학 식은 다음과 같다[16].

$$\begin{aligned} \dot{x} &= v \cos \phi \\ \dot{y} &= v \sin \phi \\ \dot{\phi} &= \omega \end{aligned} \quad (1)$$

여기서 v 와 ω 는 로봇의 속도와 각속도를 의미하고 로봇의 제어 입력으로 사용된다. 모바일 로봇의 진행 벡터 Q 는 다음과 같다.

$$Q = [x + d \cos \phi, y + d \sin \phi, \phi]^T \quad (2)$$

Q 는 모바일 로봇의 목표 위치 추종에 사용되고 ϕ 는 로봇의 위치 x, y 를 기준으로 수정 되므로 다음 식에서 ϕ 는 언급하지 않는다. 식 (2)에서 ϕ 를 제외하고 Q 를 미분하면 다음과 같다.

$$\dot{Q} = \begin{bmatrix} \dot{x} - d \dot{\phi} \sin \phi \\ \dot{y} + d \dot{\phi} \cos \phi \end{bmatrix} \quad (3)$$

식 (3)에 식 (1)을 대입하면 다음과 같은 두 개의 행을 가

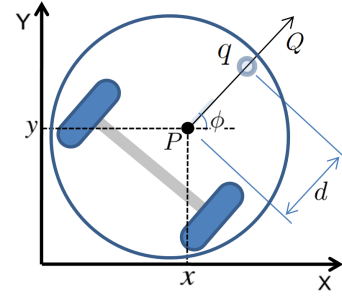


그림 3. 차등 구동 로봇 모델.

Fig. 3. Differential drive robot model.

지는 모바일로봇을 대상으로 하는 입출력 선형화 식이 나온다.

$$\dot{Q} = \begin{bmatrix} \cos \phi & -d \sin \phi \\ \sin \phi & d \cos \phi \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = ZL \quad (4)$$

여기서 $Z = \begin{bmatrix} \cos \phi & -d \sin \phi \\ \sin \phi & d \cos \phi \end{bmatrix}$ 이고 $L = \begin{bmatrix} v \\ \omega \end{bmatrix}$ 이다. 로봇의 제어 입력 L 은 $Z^{-1} \dot{Q}$ 를 해서 구할 수 있다.

Z 는 모바일 로봇의 모델링에 의해 결정되기 때문에 역변환을 할 수 있지만 로봇의 순간 움직임 \dot{Q} 은 계산해서 구하기 어렵다. 따라서 본 연구에서는 [16]과 같이 다음의 설정된 값을 사용한다.

$$\dot{Q} = \begin{bmatrix} \dot{x}_d + k_1(x_d - x) \\ \dot{y}_d + k_2(y_d - y) \end{bmatrix} \quad (5)$$

여기서 (x_d, y_d) , (\dot{x}_d, \dot{y}_d) 는 각각 목표 위치와 목표위치의 순간 속도이다. k_1 과 k_2 는 로봇과 목표위치와 오차값에 대한 이득(gain)값 이고, 추종 오차가 0으로 수렴할 수 있도록 k_1 과 k_2 는 0보다 크게 설정해야 한다.

로봇의 위치는 다음과 같이 매 스텝(step) 수정 된다.

$$P_{n+1} = P_n + \begin{bmatrix} v s_t \cos \phi \\ v s_t \sin \phi \\ \omega s_t \end{bmatrix}_{n+1} \quad (6)$$

여기서 $n+1$ 은 새로운 스텝을 의미하고 n 은 이전스텝을 의미한다. s_t 는 샘플링 시간을 의미한다. 본 논문에서 사용된 로봇 모델 파라미터 값은 $d=1$ 이고, $k_1=k_2=2$ 이다.

2. 게임 개발

Microsoft사의 XNA 게임라이브러리는 게임개발에 많이 사용된다. 라이브러리를 사용하면 객체의 충돌을 확인하는 함수가 제공되고 모든 물체들이 스프라이트(sprite)위주로 개발되기 때문에 개발이 편리한 장점이 있다. 하지만 픽셀(pixel) 단위로 게임이 설계되기 때문에 로봇모델의 적용이 어렵다. 본 논문은 로봇 모델의 적용을 위해 C++의 API (Application Program Interface)를 사용 한다.

게임 상에는 네 대의 모바일 로봇과 트랙이라는 장애물이 존재한다. 따라서 본 논문에서는 로봇과 로봇, 로봇과 트랙 벽의 충돌 회피를 위해 로봇과 로봇, 로봇과 트랙의 거리를 매 샘플링 시간마다 계산한다.

제안된 레이싱 게임은 여러 대의 아이폰이 애드혹 통신을 기반으로 PC에 무선 접속하여 실행된다. 따라서 전송되는 데이터를 손실 없이 수신하기 위해 C++언어에서 제공하는 인터럽트를 이용한다. 인터럽트 함수의 버퍼에 수신된 아이폰의 명령을 이용하여 x_d 와 y_d 를 변경하면 모바일 로봇은 그 위치를 추종하고 로봇은 제어된다. 로봇의 제어는 사용자의 핸들 조작에 의해 변경되는 식 (5)에 있는 x_d 와 y_d 값에 의해 결정된다. 로봇은 식 (6)을 이용하기 위해 v 와 ω 가 필요하다. v 와 ω 는 식 (2)의 Q 와 관련이 있다. 로봇을 이동하기 위해 로봇의 중심 P 에서 (x_d, y_d) 까지의 거리는 P 점을 기준하는 d 보다 더 큰 값으로 정해진다. 모바일 로봇의 최고 이동속도는 하드웨어 특성상 최대값을 제한한다. 표 1에서 처럼 스마트폰으로 부터 전진 명령이나 회전명령이 오면 로봇은 다음 수식에 의해 x_d 와 y_d 를 결정한다.

$$\begin{aligned} x_d &= x + (d + d_m) \cos(\phi + \phi_m) \\ y_d &= y + (d + d_m) \sin(\phi + \phi_m) \end{aligned} \quad (7)$$

여기서 d_m 은 로봇의 이동속도를 제어할 때 사용된다. ϕ_m 은 로봇의 회전각을 제어할 때 사용된다. 예를 들어 로봇이 직진하기 위해서는 $\phi_m = 0$, $d_m = 0.1$ 의 값으로 설정 한다. 만약 왼쪽 회전 명령이라면 $\phi_m > 0$ 이 되고, 오른쪽 회전 명령이면 $\phi_m < 0$ 이 된다. 실제로 d_m 을 설정할 때 d_m 이 너무 크면 모바일 로봇의 속도가 빨라져 장애물과의 충돌이 많아진다. d_m 이 너무 적으면 다음 스텝의 변화 속도가 너무 작아진다. 따라서 여기서는 $d_m = 0.01$ 로 설정한다. ϕ_m 의 설정은 스마트폰의 변화각과 상관없이 왼쪽, 정, 오른쪽 방향만을 인식하여 회전을 하도록 한다. 이것은 사용자가 가장 단순하게 모바일 로봇을 조정한다는 의미로 해석할 수 있다. 여기서 $|\phi_m| = 1$ [deg]로 설정한다.

제어 입력 v 와 ω 를 구하기 위해 식 (7)을 식 (5)에 대입하여 L 을 구하면 다음과 같다.

$$\begin{aligned} L &= Z^{-1} \dot{Q} \\ &= \begin{bmatrix} \cos\phi & \sin\phi \\ -\frac{1}{d} \sin\phi & \frac{1}{d} \cos\phi \end{bmatrix} \begin{bmatrix} \dot{x}_d + k_1(d + d_m) \cos(\phi + \phi_m) \\ \dot{y}_d + k_2(d + d_m) \sin(\phi + \phi_m) \end{bmatrix} \end{aligned} \quad (8)$$

식 (8)에서 모바일 로봇을 아이폰에서 전송하는 데이터 명령대로 모바일 로봇을 이동하기 위해서는 식 (8)의 d_m 과 ϕ_m 을 조정하여 이동할 수 있다. 만약 로봇과 로봇, 로봇과 벽면과의 거리가 일정수준 이하이면 로봇은 장애물 회피를 위해 식 (7)에서 $d_m < d$ 로 설정한다. 따라서 $d_m = -0.3$ 으

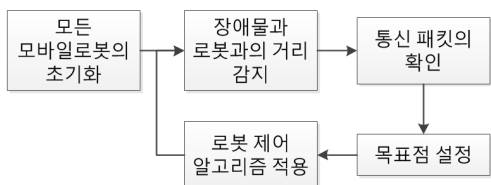


그림 4. 프로그램 흐름도.
Fig. 4. Program flow chart.

로 설정하였다.

그림 4는 게임 개발에 사용된 데이터 흐름도이다. 프로그램은 초기화 과정을 거쳐서 실행된다. 초기화 과정을 통해 모든 모바일 로봇의 로봇모델과 이득 파라미터 값을 설정하고, 샘플링시간을 통해서 주(main)함수는 반복 실행된다. 주함수에는 다음 내용이 포함된다.

- a) 로봇과 장애물과의 거리 계산
- b) 로봇에 전송된 패킷 확인
- c) 로봇의 패킷을 통해서 목표점 수정
- d) 로봇 제어 알고리즘 적용

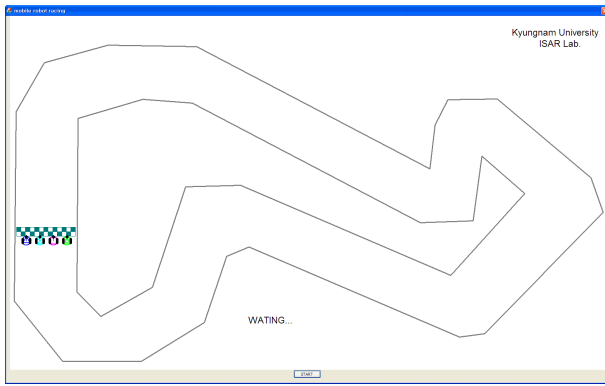
IV 장은 III 장에서 설명한 개발 방법을 통하여 하나의 모니터 화면에서 네 명의 사용자가 자신의 모바일 로봇을 제어할 수 있는 레이싱 게임 개발 내용을 다룬다.

IV. 레이싱 게임 구동 실험

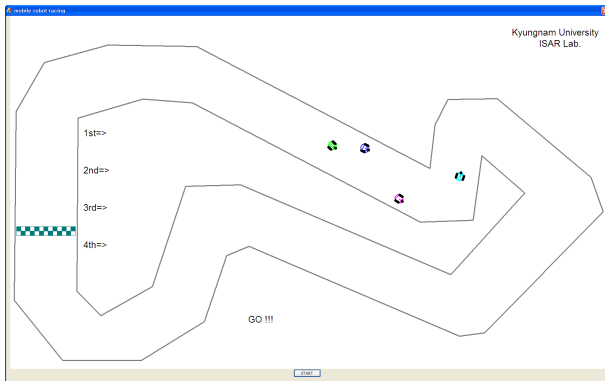
다수의 원격제어기로 게임콘솔(game console)에서 게임을 하기 위해서는 특정 게임콘솔과 원격제어기가 필요하다. 제안된 연구에서는 게임콘솔 대신 PC를 사용하고 원격제어기로 스마트폰을 사용한다는 점에서 일반 게임시스템과는 하드웨어 측면에서 다르다. 이것은 PC가 있는 일반 장소에서 다수의 사용자가 자신의 스마트폰을 사용해 하나의 모니터 화면을 보며 동시에 같은 게임을 할 수 있게 해준다. 제안된 게임 시스템에서는 한 대의 PC와 네 대의 아이폰이 필요하고, PC에 연결된 모니터에서 네 대의 아이폰을 사용하여 애드혹 무선 통신으로 각자의 모바일 로봇을 움직인다. 레이싱 게임은 다음의 루틴(routine)을 가진다.

- a) 모니터에 있는 로봇중 자신이 움직일 로봇을 아이폰을 통해 선택한다. 네 명 모두 선택이 끝났으면 모니터에 있는 “START” 버튼을 눌러서 5초간의 숫자세기(counting) 후 출발한다.
- b) 현재 사용자의 모바일 로봇이 상대 로봇들과 충돌을 피해 결승선으로 빠르게 갈 수 있는 희망루트(route)를 눈으로 확인한다. 로봇이 희망루트대로 갈 수 있도록 아이폰 인터페이스를 이용하여 로봇을 왼쪽, 진진, 오른쪽 방향으로 그림 2에 있는 핸들을 움직인다.
- c) 사용자에 의해 조작된 인터페이스는 표 1에 따라 데이터 패킷을 만들고, 아이폰과 PC와 무선 연결된 애드 혹 모듈을 통해서 아이폰의 명령패킷이 PC로 전송된다.
- d) PC에서는 아이폰으로부터 전송된 패킷을 분석한 후 패킷에 맞게 각각의 로봇을 하나의 모니터에 있는 게임 환경 안에서 움직인다.
- e) 모든 로봇이 결승선을 통과할 때까지 a)부터 d)까지의 과정이 반복된다.

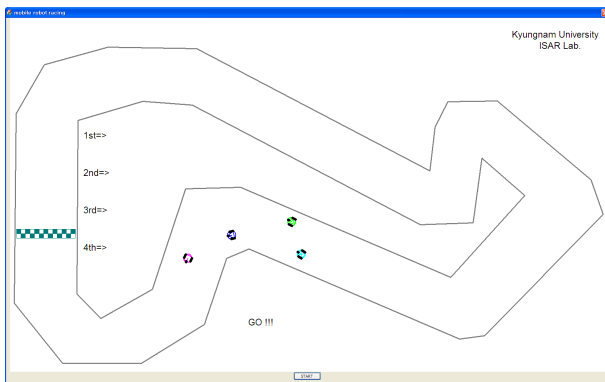
그림 5는 구현된 게임 환경 내에서 게임이 진행되는 것을 보여준다. 네 명의 사용자가 하나의 모니터 화면을 보며 각자의 스마트폰을 조정하며, 왼쪽에서 출발하여 다시 출발선으로 먼저 도착하는 사용자가 승리하게 된다. 로봇은 벽, 로봇과 로봇의 충돌이 일어나면 III 장에서 언급한 내용처럼 목표점의 위치 변화로 로봇은 충돌 지점으로 넘어가지 않게 된다. 실험에 대한 동영상은 [17]에 있다.



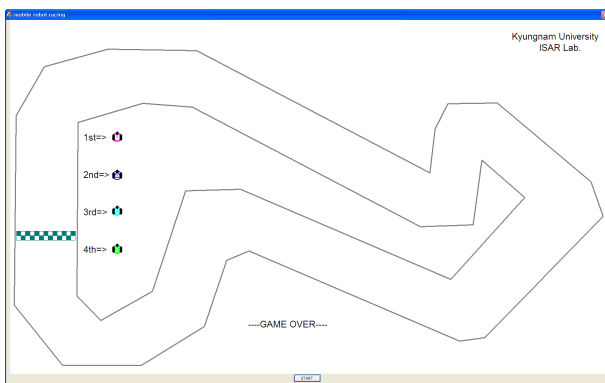
(a) 0 sec.



(b) 15 sec.



(c) 30 sec.



(d) 50 sec.

그림 5. 네 대의 스마트폰을 사용하는 레이싱 게임 구동 장면.
Fig. 5. Experiment of the proposed racing game using four smartphones.

V. 결론

본 논문에서는 애드혹 통신을 기반으로 다중스마트폰을 사용하여 PC 프로그램에 의해 구동되는 모바일 로봇을 제어하는 연구에 대하여 다루었다. 제안된 다중 스마트폰 제어에 의한 레이싱게임 구현은 저자가 기존에 연구한 내용 [13]이 확장되어서 이루어진 결과이다. 다중스마트폰과 PC와의 통신에서 게임 속 모바일 로봇 제어는 신속하게 이루어져야 하고, 데이터 손실이 없어야 한다. 개발된 게임은 바뀌가 두 개인 모바일 로봇의 운동학 모델을 이용하기 때문에 [14]처럼 모든 로봇의 위치를 측정할 수 있는 상황에서 실제 로봇을 대상으로 응용 가능하다. 제안된 다중스마트폰 제어에 의한 레이싱게임 구현은 게임콘솔이나 원격 제어가 없어도 된다. 따라서, 일반 PC와 모니터를 이용하여 스마트폰이 있는 사람들을 대상으로 실내뿐만 아니라 거리에 모인 스마트폰 사용자들이 자신의 스마트폰을 이용하여 게임이 가능하므로, 더욱 쉽고 간편하게 여러 사람이 하나의 모니터로 게임을 즐길 수 있게 해준다. 실험에서는 네 대의 아이폰이 PC에서 각자의 로봇을 위한 조정기로 사용되었고 가속도 센서 기반의 아이폰 인터페이스가 적절하게 사용될 수 있음을 보여주었다.

REFERENCES

- [1] Y. Li and L. Zeng, "Research and application of the EAN-13 barcode recognition on iphone," *International Conference on Future Information Technology and Management Engineering (FITME)*, vol. 2, pp. 92-95, 2010.
- [2] 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," *2011 IEEE 37th Annual Northeast Bioengineering Conference*, pp. 1-2, 2011.
- [3] K. R. Lee, "Development of smart fingerprint recognition system with android platform," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 11, pp. 1018-1026, 2012.
- [4] J. Kim and S. Lee, "Estimation of the user's location/posture for mobile augmented reality," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 11, pp. 1011-1017, 2012.
- [5] H. Jung, S. Y. Rhee, S. G. Gook, and D. H. Kim, "Unit control of an autonomous mobile robot using a smartphone," *Proc. of KIIS Spring Conference 2013 (in Korean)*, vol. 23, no. 1, pp. 217-218, 2013.
- [6] R. Ouch and B. Rouse, "Developing a driving training game on Windows mobile phone using C# and XNA," *2011 16th International Conference on Computer Games (CGAMES)*, pp. 254-256, 2011.
- [7] M. Joselli, J. R. Da Silva, M. Zamith, E. Clua, M. Pelegrino, E. Mendonca, and E. Soluri, "An architecture for game interaction using mobile," *2012 IEEE*

International Games Innovation Conference (IGIC), pp. 1-5, 2012.

- [8] S. Zammit, A. Muscat, and G. Gauci, "Mobile gaming on a virtualized infrastructure," *2012 16th IEEE Electrotechnical Conference (MELECON)*, pp. 745-748, 2012.
- [9] A. H. Basori, R. Sarno, and S. Widyanto, "The development of 3D multiplayer mobile racing games based on 3D photo satellite map," *WOCN '08. 5th IFIP International Conference on Wireless and Optical Communications Networks*, pp. 1-5, 2008.
- [10] J. H. Choi, S. N. Noh, E. J. Kim, S. J. Kim, and C. G. Song, "PC game system using smart phone," *2010 Korea Multimedia Society Conference (in Korean)*, pp. 630-632, 2010.
- [11] 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 IEEE SoutheastCon 2010*, pp. 9-12, 2010.
- [12] C. Fernandes, K. Y. Ng, and B. H. Khoo, "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.
- [13] D. H. Kim, "Development of a Simulator for a mobile robot based on iPhone," *Journal of Korean Institute of Intelligent System (in Korean)*, vol. 23, no. 1, pp. 29-34, 2013.
- [14] H. Jung and D. H. Kim, "Study of iPhone interface for remote robot control based on WiFi communication," *Journal of Korean Institute of Intelligent System (in Korean)*, vol. 22, no. 5, pp. 669-674, 2012.
- [15] H. Jung and D. H. Kim, "Study on communication interface of multiple smartphones for unit control in a PC," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 19, no. 6, pp. 520-526, 2013.
- [16] D. Kim and J. Oh, "Tracking control of a two-wheeled mobile robot using input-output linearization," *Control Engineering Practice*, vol. 7, no. 3, pp. 369-373, 1999.
- [17] <http://www.youtube.com/watch?v=IKrguM4rVVM>
- [18] H. Jung, Y. Kim, and D. H. Kim, "Development of a racing game controlled by multiple smartphones," *Proc. of 2013 28th ICROS Annual Conference (in Korean)*, pp. 208-209, 2013.



정 하 민

2009년 경남대학교 전기전자공학부 졸업. 2011년 동 대학원 석사. 2011년~현재 경남대학교 첨단공학과 박사 과정. 관심분야는 스웬 로봇시스템, 지능로봇제어, 이동로봇, 영상처리.



김 동 헌

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 교수. 관심분야는 스웬 시스템, 이동로봇 경로계획, 다개체 로봇, 스마트폰 응용 로봇, 지능로봇 제어, 탑승형 로봇, 국방 로봇, 비선형 적응제어.