

10th
Anniversary

2010년도

제8권 2호

동계학술대회 논문집

IWIT 2010 WINTER SMART GREEN IT CONVERGENCE CONFERENCE

일시 2010년 12월 10일(금요일)

장소 한국과학기술회관(서울 강남)

주관 및 주최 (사)한국인터넷방송통신학회(IWIT), (사)인터넷방송통신기술원(IWBC)

후원 한국과학기술단체총연합회, 방송통신위원회, 한국연구재단,

한국인터넷진흥원, 한국전파진흥원

협찬 SK C&C, 삼성 SDS, (주)GES, (주)경봉, (주)한국기술연구소,

(주)맨앤티셀, 길정보시스템(주), 하이버스(주), 한빛미디어



사단
법인

한국인터넷방송통신학회

The Institute of Webcasting, Internet and Telecommunication

www.iwit.or.kr

- PB-9 ▶ Bi-End 구조의 Inline 분산 제어가 적용된 광전송 링크에서의 WDM 신호의 BER 특성 / 224
[이성렬 (목포해양대학교)]
- PB-10 ▶ 지능형 U-Life Total Care Service 플랫폼 설계 / 228
[김찬규, 강명석, 홍인화 (전자부품연구원)]
- PB-11 ▶ 스마트TV 환경 기반 웹미디어의 콘텐츠 동적결합 / 231
[홍인화, 김정호, 김찬규 (전자부품연구원)]
- PB-12 ▶ Android 기반 무인로봇카 운행제어 시스템 개발 / 234
[장지호, 조진호, 우수정, 서채연, 김영철 (홍익대학교)]
- PB-13 ▶ 공개 소프트웨어 기반 스마트 폰 용 RSS Reader APP 및 검색 엔진 개발 / 237
[이정준, 이성민, 이현승, 서채연, 김영철 (홍익대학교)]
- PB-14 ▶ 스마트폰 기반 모바일 의학교육 시스템 / 243
[변소영, 이경록, 정용규 (울지대, 정보통신산업진흥원)]
- PB-15 ▶ USN을 이용한 PACS 영상정보의 전달 / 246
[권순일, 이경록, 정용규 (울지대, 정보통신산업진흥원)]

【네트워크/유비쿼터스/Rfid 및 USN(PC) : 13:30 - 14:30

좌장 : 이영대(IWBC), 주복규(홍익대), 조현경(동아방송대), 이기영(울지대)
발표장소 : 중회의실 2

- PC-1 ▶ 근거리 측정용 FMCW 레이더 설계 / 250
[강정진, 이종국, 김완식 (동서울대, 디지털론, LIG넥스원)]
- PC-2 ▶ VR공간에서 Haptic 디바이스를 활용한 터치-시스템에 관한 연구 / 254
[김한솔, 김선일, 김법석, 허기택, 고영혁 (동신대)]
- PC-3 ▶ 모바일과 인터넷 기반의 홈 네트워크용 가전기기 제어회로 구현 / 258
[윤달환, 연상호 (세명대)]
- PC-4 ▶ 감정인식 기반 스마트 홈 네트워크 시스템에 관한 연구 / 261
[김가람, 윤재관, 김규호, 이기영 (울지대, ETRI)]
- PC-5 ▶ 재활훈련을 위한 실시간 관절 동작 추출 기법 / 264
[김경민, 안응식, 김규호, 이기영 (울지대, 한국공간정보통신)]

Android 기반 무인로보카 운행제어 시스템 개발

Development of Unmanned RoboCar Operation Control System Based on Android Smart System

장지호*, 조진호, 우수정, 서채연, 김영철

Jang JiHo, Cho JinHo, Woo SuJeong, C. Seo, R. Y. Kim

jangjiho0418@gmail.com, wer330@hanmail.net, smartwooss@hanmail.net

요약

본 논문은 현재 스마트폰사용이 이슈화 되고 있는 실정에서 Android 스마트폰을 이용해 무인 로보카 제어에 목적을 두고 있다. 안드로이드 기반 무인으로 로보카를 제어하는 방법을 제안하고 구현하였다. 그 방법으로 Android smart phone 통해, 로보카를 제어하는 소프트웨어 개발에 있다. 그 결과, 서버를 통해 로보카로 전달하여 사용자가 언제 어디서든 무인 로보카를 제어할 수 있게 구현하였다.

키워드: RoboCar, Android, SmartPhone, Bluetooth, WiFi

I. 서론

다가오는 미래에는 다양한 로봇과 함께 인간이 공존하여서 인간이 쉽게 접근할 수 없는 화재와 붕괴 현장에서 구조 수색 활동 등에서 접근하려는 대안이 필요한 실정이다.

가장 최근에 칠레 광부 매몰 사건을 보면 사람이 살아 있거나 위험성 때문에 쉽게 접근을 하지 못하였다. 본 논문은 이러한 환경에서 쉽게 수색 탐색활동을 할 수 있는 무인제어 로봇을 요즘 핫이슈로 떠오르고 있어, 안드로이드 스마트폰을 이용해 제어하는 방안에 대해서 언급했다.

본 논문에서 제안한 무인 로보카 운행 제어시스템은 로봇부 분과 서버 그리고 안드로이드 세부부분으로 나눈다.

II. 본문

2.1 Android 기반 무인 로보카 운행제어 시스템 구성

본 장에서는 Android 기반 무인 로보카를 제어하기 위한 운행제어 시스템의 기본 아이디어와 시스템 구성을 설명한다. 그림1은 본 논문에서 제안하는 Android 기반의 무인 로보카 운행제어 시스템의 인터페이스 개념을 보여주고 있다.

Android폰에서 WiFi를 통하여 서버와 통신을 하고 서버는

Bluetooth를 통하여 RoboCar와 통신을 하도록 설계 하고, 안드로이드 상의 어플리케이션은 직관적인 인터페이스를 구현하였다.



그림 1. Android 기반의 무인 로보카 운행제어 시스템의 인터페이스 개념

III. Android 기반의 무인로보카 운행제어 시스템 인터페이스

3.1 Android 시스템 특성

Android는 Google사에서 나온 스마트폰을 위한 운영체제이다. 기존 휴대폰 운영체제들은 유료인 반면에 Android는 무료여서 개발자들이 개발하기에 적합한 환경을 가지고 있다. Android는 휴대전화를 비롯한 모든 휴대용 장치를 위한 운영체제와 미들웨어 그리고 핵심 응용프로그램을 포함하고 이쁜 소프트웨어 스택을 통틀어서 말한다. 안드로이드의 주요 특징

*충일대학교 컴퓨터정보통신공학과

은 다
-VC
-을
-데
-사
-안
-루
-안
-반
-합
-블
3.1
ATMec
ATMec
L289P
Ultrac
Accole
P8D to
Protol
Mbiors
Buzze
LED
7-8ag
Regul
Batter
Charg
Anc
Avn
Ecli
Os

은 다음과 같다.

- VGA, 2D 그래픽 라이브러리, OpenGL ES 1.0에 기반을 둔 3D 그래픽 라이브러리를 확장하기에 적용적이다
- 데이터 저장 목적의 SQLite 데이터베이스 소프트웨어가 사용됨
- 안드로이드는 GSM/EDGE, CDMA, EV-DO, UMTS, 블루투스, 와이파이를 포함하는 커넥션 기술을 지원한다.
- 안드로이드는 카메라, 터치스크린, GPS, 가속도센서, 나침반센서, 트랙볼 2D 그래픽 가속, 3D 그래픽 가속을 활용할 수 있다.
- 블루투스를 통한 파일 전송이 버전 2.0에 추가되었다.

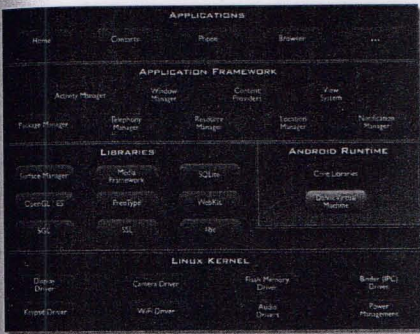


그림 2. Android Architecture

3.2 사양

구분	사양
ATmega128L	8-bit AVR, Microcontroller with 128K Bytes Flash Memory, Main control
ATmega8L	8-bit AVR, Microcontroller with 8K Bytes Flash Memory, Voltmeter control
L298P	Up to 4A DC Motor Driver 2EA
Ultrasonic sensor	40,0±0.5kHz Frequency, 2.0kHz Bandwidth, 2EA
Accelerometer sensor	Dual-Axis Accelerometer sensor 1EA, 10-80cm
PSD sensor	Distance Measuring Sensor 1EA, 10-80cm
Phototransistors	8-groups infrared rays sensor
Motors	DC geared motor 2EA, DC geared encoder motor 2EA
Buzzer	5V Input Buzzer 1EA
LED	10mm high brightness LED, White 2EA, RED 2EA
7-Segment	Voltmeter Display, 3-Digit 1EA
Regulator	+11.1V DC Input, DC Output: +5V, +3.3V
Battery	+11.1V, 5200mA Lithium Ion 1EA
Charge & Adapter	+12.6V 1.2A Battery Charger 1EA

그림 3. Robocar 시스템 사양

Android 2.2(proyo ver)
 Avr Studio
 Eclipse
 Os : Windows7

3.3 안드로이드 폰 상 운행제어 인터페이스



그림 4. Android Application Interface

제어 프로그램 인터페이스는 직관적인 십자키 형태로 전진 좌회전 우회전 정지 버튼이 배치하고, 하단에는 속도조절기능을 두어 간단하면서도 직관적으로 배치하였다.

3.4 Android기반 무인로보카 제어시스템 프로그래밍

먼저 로보카의 동작을 위한 네트워크 면에서 살펴보는 자세한 그림은 그림5와 같다. 사용자가 스마트폰에서 설계된 Interface 버튼의 버튼을 누르게 되면 Remote Controller 상에 WiFi Sender를 통해 서버상에 있는 WiFi Receiver를 통해 정보를 받게 된다. Server는 받은 데이터를 Message Parser로 보내게 되고, Message Parser는 받은 데이터를 다시 Bluetooth Sender로 보내게 된다. Bluetooth Sender는 다시 Bluetooth 망을 통해 RoboCar에 있는 Bluetooth Receiver로 데이터를 보내게 된다. Bluetooth Receiver는 받은 데이터를 Robocar 상에 있는 Message Parser로 보내게 된다. RoboCar 상에 있는 Message Parser는 RoboCar Controller로 데이터를 보내게 되고 RoboCar는 이 받은 데이터를 바로 실행하는 방식으로 RoboCar의 작동이 진행된다.

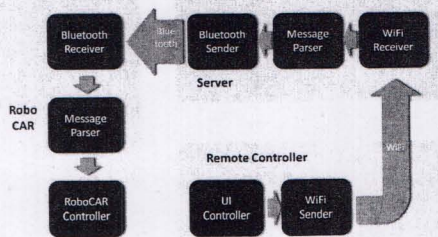


그림 5. Android 기반 무인로보카 제어시스템 네트워크 개념도

```
public void run() {
    // TODO auto-generated method stub
    try {
        InetAddress serverAddr = InetAddress.getByName(serverIP);
        Socket socket = new Socket(serverAddr, serverPort); //소켓 생성

        PrintWriter out = new PrintWriter(new BufferedWriter(
            new OutputStreamWriter(socket.getOutputStream(), true));
        out.println(msg);
        BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
        return msg = in.readLine();
    }
}
```

그림 6. 어플리케이션 소스

위 그림은 안드로이드 어플리케이션의 핵심 코드로 소켓을 생성한후 입력된 명령을 소켓을 통해 서버로 전송하는 부분이다.

```
try {
    System.out.println("S- Connecting...");
    ServerSocket serverSocket = new ServerSocket(serverPort);
    portList = DownPortIdentifier.getServerPortIdentifiers();
    while (portList.hasMoreElements()) {
        Socket client = serverSocket.accept();
        System.out.println("S- Receiving...");

        try {
            BufferedReader in = new BufferedReader(new InputStreamReader(client.getInputStream()));
            messagesString = in.readLine();
            System.out.println("S- Received: " + messagesString + "");
            PrintWriter out = new PrintWriter(new BufferedWriter(new OutputStreamWriter(client.getOutputStream(), true));
            out.println("Server Received: " + messagesString);
            outputStream.write(messagesString.getBytes());
        }
    }
}
```

그림 7. 어플리케이션 관련 소스

위 그림은 서버의 핵심 코드로 로보카로 보내질 명령어를 안드로이드 어플리케이션에서 읽어오는 부분이다.

```
while(1){
    Command = RoboCAR_getch();
    switch(Command){
        case 'v':
            PORTA &= DEFQ;
            PORTA |= FORWARD;
            break;
        case 'b':
            PORTA &= DEFQ;
            PORTA |= BACK;
            break;
        case 'j':
            PORTA &= DEFQ;
            PORTA |= LEFT;
            break;
        case 'g':
            PORTA &= DEFQ;
            PORTA |= RIGHT;
            break;
        case 't':
            PORTA &= DEFQ;
            PORTA |= LEFT_U;
            break;
        case 'u':
            PORTA &= DEFQ;
            PORTA |= RIGHT_U;
            break;
        case 's':
            PORTA &= DEFQ;
            PORTA |= STOP;
            break;
    }
    ms_delay(100);
}
```

그림 8. 로보카 관련 소스

위 그림은 로보카의 핵심코드로 각각의 케이스별 명령어에 따른 RoboCar의 행동을 정의하고 있다.

V. 결론

본 논문에서는 Android기반으로 무인 로보카 제어시스템을 제안하고 구현하였다. 먼저 로보카와 서버간의 통신을 구현하였고 Android 클라이언트에 버튼으로 구성된 User Interface를 구성하였다. 클라이언트에서 입력된 동작명령을 서버를 통해 로보카로 전달하여 사용자가 언제 어디서든 로보카를 제어할 수 있게 구현하였다.

추후 연구 방향으로는 Android플랫폼이 제공하는 환경을 충분히 이용하여 보다 인간 친화적 이면서 인지적인 인터페이스 환경을 구축하는 것이라 할 수 있다.

또한 로봇에게 명령을 내리면 로봇이 스스로 나온 방법을 생각해 명령을 수행하는 방식을 생각해 볼 수 있겠다.

본 연구는 한이음 컨소시엄사업(주관기관: 홍익대학, 참여기업(주)한백전자) 과 교육과학기술부와 한국연구재단의 지역혁신인력양성사업으로 수행된 연구결과임.

참고 문헌

1. <http://developer.android.com/index.html> 안드로이드 개발자 사이트
2. <http://www.hanback.co.kr/> 한백전자