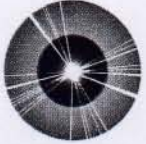


제13권 제1호
Vol. 13 No. 1



한국정보과학회
Korean Institute of Information Scientists and Engineers



한국정보처리학회
Korean Information Processing Society



2011

한국 소프트웨어공학 학술대회 논문집(안)

(Preliminary)

Proceedings of 2011 Korea Conference on
Software Engineering

- 일시 : 2011년 2월 9일(수)~11일(금)
- 장소 : 한화 휘닉스 파크(강원도 평창)

주최 : 한국정보과학회, 한국정보처리학회

주관 : 한국정보과학회 소프트웨어공학 소사이어티
한국정보처리학회 소프트웨어공학 연구회
한국전자통신연구원

후원 : 삼성 SDS, 비트컴퓨터, (주)모아소프트, (주)이웨이파트너즈,
단국대학교SERC(소프트웨어공학연구센터), 고려대학교 고신뢰
융합소프트웨어 연구센터, 서강대학교 SW 요구 및 검증공학기술
연구센터, 숭실대학교 모바일서비스 SW공학센터, 포항공대
융합소프트웨어개발 연구센터

민대규, 김동성 (승실대학교)

Morphology-pair 에 기반한 ECG 신호의 기초선변동 제거 알고리즘-----438

구혜은, 김세윤, 이희열, 정건희, 박길흠 (경북대학교)

이동 단말기를 이용한 교통 정보 생성-----444

오제한, 이은석(성균관대학교)

F3: 국방 소프트웨어 프로세스 및 관리

함정 전투관리 시스템에서의 이기종 분산 시스템을 위한 AMSM 기반 통합 관리 시스템
구현-----453

민법기, 국승학, 김현수 (충남대학교)

김정수(국방과학연구소)

무기체계 내장형 소프트웨어 품질 향상을 위한 온라인 동료검토(Peer Reviews) 기능의
구현 및 적용사례-----463

김영균, 정지운, 유석진, 김대용, 안효철, 한병학, 김영수 (현대로템)

전지구 가시화를 위한 지형 렌더링 방법에 관한 연구-----467

정지환, 이재영, 김필수 (픽소니어)

황선명 (대전대학교)

김인철, 김성호 (국방과학연구소)

G1: 소프트웨어 아키텍처 및 설계

다기종 스마트폰을 적용한 다중 제어방법의 서빙 로봇 설계 및 구현-----473

장우성, 최태웅, 김영수, 김영철 (홍익대학교)

파일의 의미정보 표현을 이용한 파일관리 기법에 대한 설계와 구현-----478

조용균, 박원규, 박용범 (단국대학교)

변수의 가중치를 이용한 응집도 척도의 개선-----487

박철현, 이은석 (성균관대학교)

G2: 모바일 소프트웨어 시험 및 검증

다기종 스마트폰을 적용한 다중 제어방법의 서빙 로봇 설계 및 구현

장우성*, 최태웅**, 김영수***, 김영철*

홍익대학교 컴퓨터정보통신공학과 소프트웨어공학전공*

(주)한백전자**, SEC@NIPA***

jang@selab.hongik.ac.kr, choitw@hanback.co.kr, ysgold@nipa.kr, bob@hongik.ac.kr

요약: 서빙 로봇은 손님에게 주문을 받아 음식을 전달하는 로봇이다. 서빙 로봇을 개발하기 위한 방법은 두 가지가 있다. 첫 번째는 서빙 로봇이 모든 일을 할 수 있도록 하는 방법이다. 이 경우 서빙 로봇 자체의 기능이 복잡해져서 개발하기가 매우 어렵다. 두 번째는 각 기기마다 역할을 분담하고 상호 협력하여 작업을 처리할 수 있도록 하는 방법이다. 예를 들어 손님이 스마트폰으로 음식을 주문하면, 서버가 주문을 접수하고, 서빙 로봇이 음식을 손님에게 전달하는 것과 같다. 이 경우 각각의 역할분담만 잘하면 매우 효과적으로 작업을 처리하는 것이 가능하다. 본 논문에서는 두 번째 방법을 사용하여 다기종의 스마트폰으로 주문이 가능한 서빙 로봇을 설계 및 구현한다.

핵심어: 스마트폰, 서빙 로봇, 다중 컨트롤러, 다중 접속, RoboCAR

1. 서론

근래 들어 로봇은 다양한 종류가 개발되었으며, 청소나 요리 등의 실생활에 적용되어 쓰이고 있다. 이 중 서빙 로봇은 음식점에서 손님에게 주문을 받아 음식을 전달하는 로봇이다. 서빙 로봇을 개발하기 위한 방법은 두 가지가 있다. 첫 번째는 서빙 로봇이 손님이 음식점으로 들어오는 것을 인식해서 주문을 받아 음식이 나오면 음식을 전달하는 것이다. 이와 같은 서빙 로봇은 주위 환경을 볼 수 있는 카메라, 사물을 구별할 수 있는 인공지능, 손님의 주문을 들을 수 있는 음성인식 기능 등 복잡한 기술이 필요하기 때문에 개발이 어렵다. 두 번째는 각 기기마다 역할을 분담하고 상호 협력하여 작업을 처리할 수 있도록 하는 방법이다. 예를 들어 손님이 스마트폰으로 음식을 주문하면, 서버가 주문을 접수하고, 서빙 로봇이 음식을 손님에게 전달하는 것과 같다. 이 경우 각각의 역할분담만 잘한다면 매우 효과적으로 작업을 처리하는 것이 가능하다.

기존의 로봇은 UPnP(Universal Plug and Play),

Web, Jini 를 적용하여 구현하였다. 하지만 다중 접속을 지원하지 않거나, 스마트폰 환경에 적합하지 않은 단점이 있어 적용에 어려움이 있었다. UPnP 는 기능적 기기나 무선 기기들 간의 peer-to-peer 네트워크를 쉽게 구성할 수 있도록 도와준다[1]. UPnP 기반의 로봇은 쉽게 다른 기기와 연결할 수 있는 장점이 있다. 하지만 여러 기기들을 다중 지원하는데 어려움이 있다. Web 기반의 로봇은 웹 브라우저를 통해 로봇을 제어한다. 웹 브라우저가 설치된 스마트폰이라면 쉽게 연결할 수 있는 장점이 있지만, 스마트폰의 GPS 기능을 활용할 수 없는 단점이 있다. Jini 는 썬 마이크로시스템즈사에서 발표한 분산 환경의 네트워크 자원공유를 위한 Java 기반의 플랫폼이다. Jini 를 이용하면 네트워크 상의 모든 종류의 장치와 소프트웨어 자원의 통합체를 구성하여 서비스와 자원을 공유할 수 있다. 하지만 Java 가 기반이기 때문에 JVM 을 설치할 수 있는 OS 가 있는 기기에만 적용 가능한 단점이 있다[6].

본 논문은 다기종의 스마트폰으로 다중 제어가 가능한 서빙 로봇을 설계 및 구현하였다. 클래스 다이어그램 기반의 서빙 로봇, 스마트폰, 서버 모델을 설계하였으며, 적용 사례로 (주)한백전자의 RoboCAR 의 펌웨어, 안드로이드폰 어플리케이션, 서버를 구현하였다.

본 논문의 구성은 다음과 같다. 2 장에서는 관련 연구에 대해서 설명한다. 3 장에서는 서빙 로봇의 설계에 대해서 설명한다. 4 장에서는 RoboCAR 와 안드로이드폰을 이용한 시스템 구축에 대해서 설명한다. 5 장에서는 결론 및 향후 연구를 언급한다.

2. 관련연구

2.1 UPnP 기반의 서빙 로봇 제어 방법

UPnP(Universal Plug and Play) 기술은 지능적 기기나 무선 기기들 간의 peer-to-peer 네트워크를 쉽게 구성할 수 있도록 한다. XML, HTTP 와 같은 표준

† 이 논문은 2010 년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업(2010-0012117)과 교육과학기술부와 한국연구재단의 지역혁신인력양성사업으로 수행된 연구결과임.

프로토콜을 이용하고, 스스로 필요한 디바이스를 찾을 수 있는 Auto-discovery 기능을 지원하기 때문에 로봇이 외부 스마트 디바이스와 연동하는데 편하다 [1].

UPnP의 진행과정은 자신의 IP와 Port를 획득하는 Addressing, SSDP 프로토콜을 이용하여 디바이스를 찾아내는 discovery, 검색된 디바이스의 XML 정보를 획득하는 description, SOAP 프로토콜을 이용하여 원하는 서비스를 제어하는 control, GENA 프로토콜을 사용하여 상태정보를 알려주는 Eventing과 웹을 통해 상태를 알고, 제어가 가능한 Presentation 등 6 단계로 이루어진다. UPnP의 구조는 분산형, 개방형 네트워킹 구조이며, 가정이나 사무실 어디에서나 제어와 데이터 전송을 가능하게 한다. 이러한 장점 때문에 로봇 미들웨어로써 UPnP를 사용하면 많은 장점을 가질 수 있다[2]. 하지만 UPnP는 peer-to-peer 네트워킹을 기반으로 구성되었기 때문에 기기들의 다중 접속을 처리하기가 어려운 단점이 있다..

2.2 Web 기반의 로봇 제어 방법

현재 이동로봇 제어를 할 때 가장 많이 사용되는 방법은 IEEE 802.11b를 기반으로 한 wireless LAN Socket 통신방식이다[3]. 이러한 Internet 방식 중에 TCP/IP 프로토콜의 WWW(World Wide Web)으로 구성 시 로봇이 Web에 연결되어 있다면 로봇의 IP를 통해 로봇과 웹브라우저의 연결이 가능하다. 하지만 웹브라우저는 자바스크립트나 HTML 태그 등을 통해서만 처리가 되기 때문에 로봇의 제어에 많은 어려움이 있다. 이때 Java나 Microsoft사의 ActiveX나 COM 같은 기법을 적용하면 좀 더 고급스러운 프로그래밍이 가능하며, 클라이언트에서 직접 데이터베이스에 접근할 수 있다[4]. 데이터베이스에 접근하는 방법으로는 RDO(Remote Data Objects), DAO(Data Access Objects), ADO(ActiveX Data Objects), OLE DB, ODBC(Open Database Connectivity), JDBC(Java Database Connectivity) 등이 있다[5]. 자세한 구조는 <그림 1>과 같다.

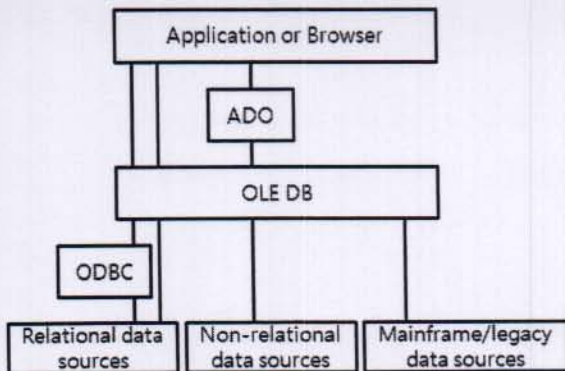


그림 1 DataBase 연결 라이브러리 구조[5]

Web 기반의 로봇 제어 방법은 다양한 기능을 제공하지만, 웹 브라우저 내에서 실행되기 때문에 스마트폰의 GPS 기능을 사용할 수 없는 단점이 있다.

2.3 Jini 기반의 이동 로봇 제어 방법

Jini는 1998년에 썬 마이크로시스템즈사에서 발표한 분산 환경의 네트워크 자원공유를 위한 Java 기반의 플랫폼이다. Jini를 이용하면 네트워크 상의 모든 종류의 장치와 소프트웨어 자원의 통합체를 구성하여 서비스와 자원을 공유할 수 있다. 그리고 사용자의 위치 변화에 관계없이 네트워크 상의 자원에 대한 접근 및 네트워크의 개설이 용이하고 갱신 및 변경 작업이 단순해진다. 로봇 제어 시스템에 Jini 기술을 도입하면 고가의 로봇 시스템을 여러 작업장에서 공유하여 사용할 수 있으므로 로봇의 활용도를 높이고 시설투자 비용을 줄일 수 있다[6]. 로봇을 공동으로 사용하는 환경에 대한 그림은 <그림 2>와 같다.

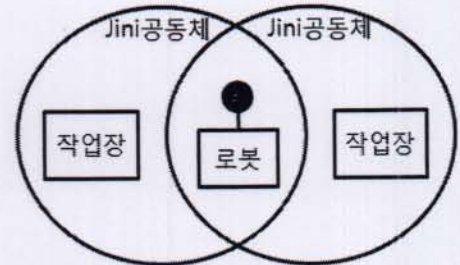


그림 2 로봇의 공동사용환경[7]

하지만 Jini 기반의 이동 로봇 제어 시스템은 Java 기술이 기반이 되어야 하기 때문에 OS가 없는 서버로봇은 적용이 어렵다.

3. 서버 로봇의 설계

3.1 시스템 구조

전체 시스템은 서버 로봇, 스마트폰, 서버로 구성된다. 서버 로봇은 위치 파악을 위한 센서와 주위의 장애물을 감지하기 위한 센서, 서버와 통신을 하기 위한 센서를 갖추고 있어야 한다. 로봇은 항상 자신의 위치 정보를 파악하고 있어야 하며, 장애물을 발견했을 때는 회피를 한다. 스마트폰은 주문자가 주문을 할 수 있게 메뉴 리스트를 제공하고, 선택된 메뉴를 서버로 전송한다. 서버는 모든 스마트폰에 메뉴 리스트를 제공하고, 스마트폰에게 메뉴 주문을 받아 내부 큐(Queue)에 저장한다. 주문을 한 테이블의 위치가 순차적으로 로봇에게 전달된다.

이에 대한 그림은 <그림 3>과 같다.

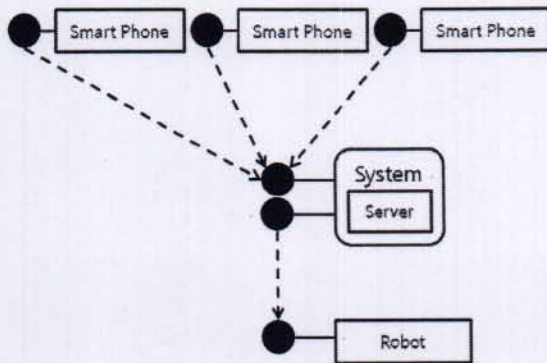


그림 3 시스템 전체 구조도

3.2 서빙 로봇 구조

서빙 로봇은 SMMModules, RBCMOTOR, RBCUltra, RBCLed, RobocarInit, AvrLib 클래스로 구성된다. SMMModules 클래스는 처음 서빙 로봇이 처음 실행될 때 함께 실행되는 클래스이며, 위치 정보를 관리한다. RBCMOTOR 클래스는 모터를 제어하는 클래스이다. RBCUltra 클래스는 블루투스 통신 기능을 제어하는 클래스이다. RBCLed 클래스는 LED 불빛을 제어하는 클래스이다. RobocarInit 클래스는 모듈 정보를 초기화 시키는 클래스이다. AvrLib 클래스는 UART 번호, 블루투스 통신속도 등의 정보를 가지고 있는 클래스이다. 다른 클래스는 AvrLib 클래스 내의 변수값들을 참조하여 서빙 로봇을 제어한다. 서빙 로봇 시스템 구조는 아래의 <그림 4>와 같다.

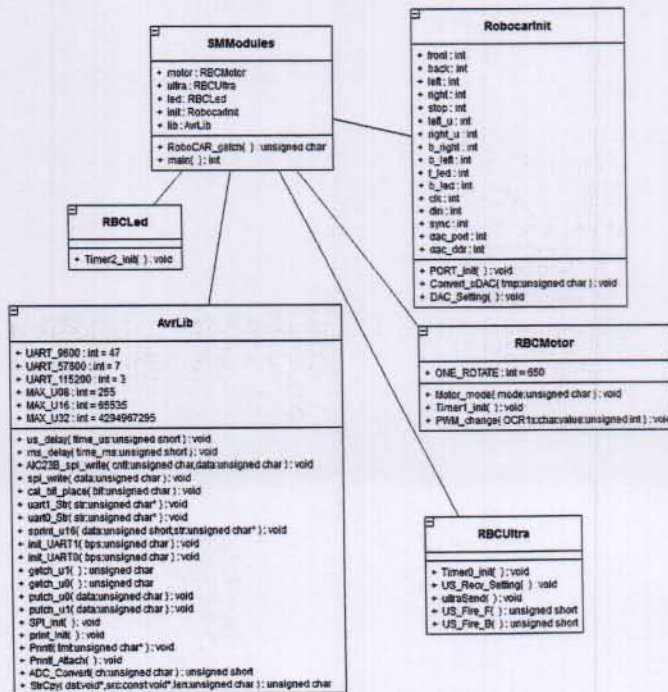


그림 4 서빙 로봇의 클래스 다이어그램

3.3 스마트폰 어플리케이션 구조

스마트폰 어플리케이션은 Main, View, ServerComm, EfficientAdapter, ViewHolder 클래스로 구성된다. Main 클래스는 어플리케이션 실행 시 가장 처음 실행되는 클래스로 View, EfficientAdapter 클래스를 변수로 선언하고, View 클래스를 실행하여 화면을 생성한다. View 클래스는 어플리케이션 내의 화면그리기, 터치이벤트를 처리하고, ServerComm 클래스를 생성한다. ServerComm 클래스는 서버와 소켓 통신을 하여 메뉴 리스트를 받아오며, View 에서 선택된 주문 리스트를 서버에 전송한다. EfficientAdapter 클래스는 View 클래스에 List 기능을 추가하기 위한 클래스이며, ViewHolder 클래스를 생성한다. ViewHolder 클래스는 EfficientAdapter 클래스를 통해 리스트에 삽입될 텍스트뷰 정보를 관리한다. 스마트폰 어플리케이션 시스템의 구조는 아래의 <그림 5>와 같다.

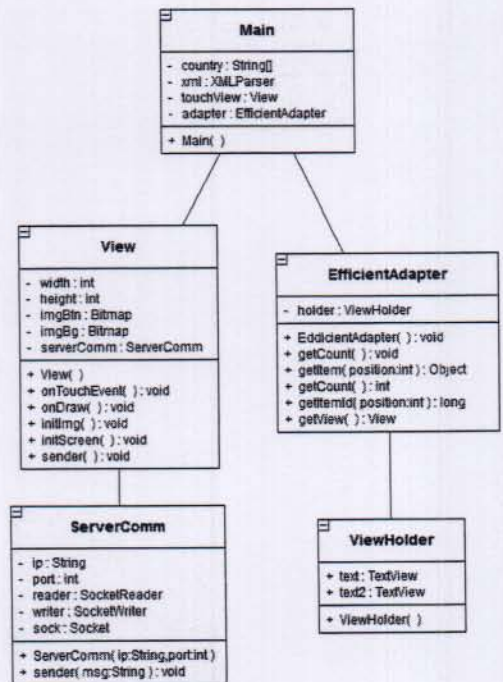


그림 5 스마트폰 어플리케이션의 클래스 다이어그램

3.4 서버 구조

서버는 Main, MainQueue, SmartPhoneComm, SmartPhoneThread, RobotComm, RoborCommThread 클래스로 구성된다. Main 클래스는 서버가 실행될 때 가장 먼저 실행되는 클래스로 MainQueue, SmartPhoneComm, RobotComm 클래스를 변수로 선언한다. MainQueue 클래스에 저장된 주문 내역을 읽어 음식을 서빙해야 하는 테이블의 위치를

RobotComm 클래스에게 전달한다. MainQueue 클래스는 스마트폰에서 보낸 주문 내역을 순차적으로 저장하는 클래스이다. SmartPhoneComm 클래스는 스마트폰이 보낸 주문 내역을 MainQueue 클래스에 저장하는 클래스로 서버 소켓을 사용하며, SmartPhoneThread 클래스를 변수로 생성한다. SmartPhoneThread 클래스는 스마트폰의 메시지를 받는 클래스이다. 서버에 접속한 스마트폰의 개수만큼 증가한다. RoboComm 클래스는 RoboCAR 에게 테이블 위치를 전달하는 클래스이며, RobotCommThread 클래스를 변수로 생성한다. RobotCommThread 클래스는 RoboCAR 의 메시지를 받는 클래스이다. 자세한 구조는 <그림 6>과 같다.

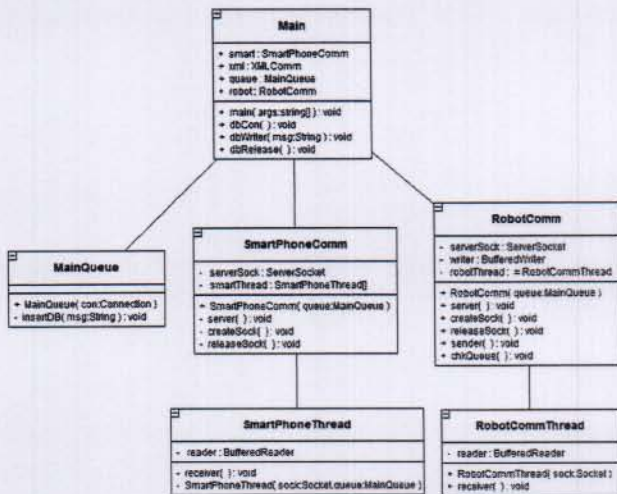


그림 6 서버의 클래스 다이어그램

4. 서빙 로봇의 구현

서빙 로봇으로 ㈜한백전자의 RoboCAR 를 적용하였다. RoboCAR 의 기본 외형은 <그림 7>과 같다.



그림 7 RoboCAR 의 기본 외형

RoboCAR 는 본체에 내장되어 있는 8Bit 마이크로 프로세서 (ATmega128L)를 이용하여 센서 값 제어 및 구동부(DC 모터) 제어를 할 수 있도록 설계되어 있다. 본체 단독으로도 구동이 가능하며, 초음파 센서, 적외선 센서, 가속도 센서, 포토 센서 등 로봇제어용 센서를 지원한다. 또한 UART 포트를 통해 다양한 모듈을 확장할 수 있는 확장성을 제공한다. 두 뇌보드를 연결하여 OS 를 설치할 수 있으며, 기본

차체에 블루투스, 와이파이 모듈을 연결하여 다른 기기와 커뮤니케이션을 할 수 있다. 하드웨어 사양에 대한 자세한 내용은 <표 1>과 같다.

표 1 RoboCAR 의 상세사양[8]

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, Duty Cycle
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

RoboCAR 는 세 가지 기능을 수행한다. 자신의 위치를 파악하고, 서버에게서 테이블의 위치를 받아 움직이고, 주위에 장애물이 있을 시 장애물을 회피한다. 자신의 위치를 파악하고, 주위의 장애물을 회피하기 위해서 바닥의 적외선 센서를 사용한다. 이를 위해서 RoboCAR 가 다니는 길을 검은색 라인으로 표시하였다. 그리고 일정 구간마다 마크를 표시하였다. RoboCAR 는 검은색 라인으로 표시된 길로만 통행이 가능하며, 일정구간마다 새겨진 마크를 통해 자신의 위치를 파악한다. 만약 검은색 라인 위에 장애물이 존재한다면, 장애물이 사라질 때까지 제자리에서 대기한다. 장애물을 파악하기 위한 센서는 RoboCAR 의 앞뒤에 위치한 적외선 센서를 이용한다. 앞뒤의 적외선 센서가 미치는 범위는 <그림 8>과 같다.

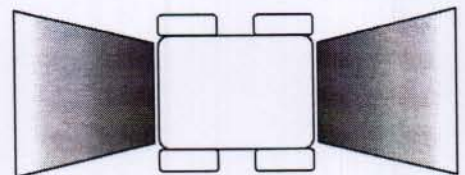


그림 8 RoboCAR 의 센서범위

서버의 메시지를 받기 위해 블루투스 모듈을 사용

한다. 기본 사양의 RoboCAR 는 통신을 위한 모듈이 존재하지 않는다. 그렇기 때문에 블루투스 모듈을 연결하여 서버와의 통신을 가능하게 하였다.

전체 시스템은 스마트폰, 서버, 서빙 로봇으로 구성된다. 주문자가 스마트폰을 통해 서버에 음식을 주문하면 서버는 주문 내용을 내부 큐에 저장한 후 주문을 차례대로 읽어 서빙 로봇을 컨트롤한다. 스마트폰의 적용사례는 안드로이드폰을 사용하였다. 서버는 일반 PC 를 사용하였다. 이에 대한 그림은 <그림 9> 와 같다.

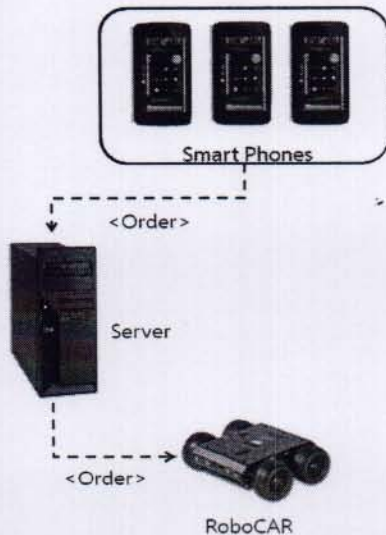


그림 9 전체 시스템 구성도

시스템의 통신구조는 스마트폰과 서버는 WiFi 통신을 하고, 서버와 RoboCAR 는 블루투스 통신을 한다. 이러한 구조 때문에 스마트폰은 WiFi 가 가능해야 하며, 서버는 WiFi 와 블루투스 통신이 가능해야 하며, RoboCAR 는 블루투스 통신이 가능해야 한다. 자세한 그림은 <그림 10>과 같다.

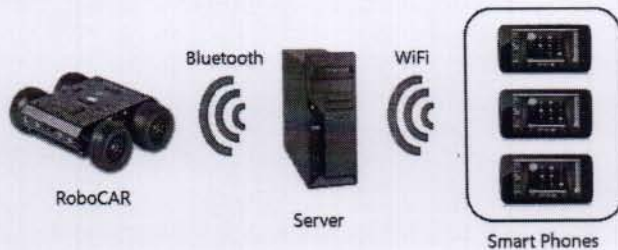


그림 10 전체 시스템 통신 구조

결과적으로 서빙 로봇은 다기종의 스마트폰으로부터 주문을 받아 쉽게 서빙을 할 수 있었다.

스마트폰은 어플리케이션을 다운받아 쉽게 메뉴를 주문 하였고, 서버에 저장된 메뉴 리스트를 받아와 항상 새로 업데이트된 메뉴를 확인할 수 있었다

5. 결론

서빙 로봇에는 혼자 스스로 생각해서 주문을 받는 방법과 주위의 기기와 상호 협력하여 동작하는 방법이 있다. 두 번째 방법의 경우 각각의 역할분담만 잘 한다면 첫 번째 방법에 비해 적은 비용으로 효율적인 작업을 진행하는 것이 가능하다.

본 논문에서는 두 번째 방법을 적용하여 다기종의 스마트폰으로 주문을 받아 서빙을 하는 로봇을 설계 및 구현하였다. 전체 시스템은 서빙 로봇, 스마트폰, 서버로 구성된다. 스마트폰이 서버에게 주문 리스트를 전송하면 서버는 주문 리스트를 큐에 저장한 후 순서대로 출력하여 주문한 테이블의 위치를 서빙 로봇에게 전송한다. 테이블 위치를 전송 받은 로봇은 서빙을 시작한다. 로봇은 서빙이 끝났다면 다음 테이블 위치를 전송 받는다. 이러한 구조를 클래스 다이어그램으로 설계하였다. 서빙 로봇은 ㈜한백전자의 RoboCAR, 스마트폰은 안드로이드 폰, 서버는 일반 PC 를 사용하였다. 여러 대의 안드로이드폰을 이용하여 주문을 한 결과 RoboCAR 가 주문한 순서대로 테이블에 서빙하는 것을 확인할 수 있었다.

향후 연구로 다른 기종의 스마트폰 어플리케이션을 제작하여 적용할 것이며, 로봇의 현재 위치 인식과 장애물 회피에 대한 방법을 개선할 것이다.

참고문헌

- [1] 임기웅, 안상철, 권용무, "UPnP 기반 로봇 미들웨어를 이용한 서빙 로봇 제어", 대한전자공학회, 2179-2182, 2010.
- [2] Snag Chul Ahn, Jin Hak Kim, Kiwoong Lim, Heedong Ko, Young-Moo Kwon, and Hyoung-Gon Kim, "UPnP Approach for Robot Middleware", ICRA2005, 1959-1963, Apr.2005.
- [3] H. Hirukawa, I. Hara, "Web-top robotics", IEEE Robotics & Automation Magazine, Vol.7, 40-45, 2000.
- [4] Chris H. Pappas, William H., "III Murray Debugging C++ : Troubleshooting for Programmers", McGraw-Hill, 2000.
- [5] 정성호, 김성주, 서재용, 김용택, 전홍태, "Web 기반 이동로봇의 다중 접속 제어 처리법", 한국지능시스템학회, 239-242, 2002.
- [6] 김선환, 송윤철, 이봉주, 유준호, "인터넷 정보가전 기술개발 기획연구", 정보통신연구진흥원, 2000.
- [7] 이동일, 한동창, 김종무, 박주현, "Jini 기반의 이동 로봇 제어 시스템에 관한 연구", 대한전기학회, 2476-2478, 2001.
- [8] ㈜한백전자 기술연구소, "HBE-RoboCAR 로 배우는 이동 로봇 제어", ㈜한백전자, 2008.