

로봇과 안드로이드 장치의 연동을 위한 소프트웨어 프레임워크 설계

이평화*, 김병규, Wu Peng, 유민수
*한양대학교 전자컴퓨터통신공학과
e-mail : {phlee, bkkim, wupeng, msryu}@hanyang.ac.kr

Extending and Modifying Android Software Platform for Effective Control of Service Robots

Pyoung-Hwa Lee*, Byung-Kyu Kim, Peng Wu, Min-Soo Ryu
*Dept. of Electronics Computer Engineering, Han-Yang University

요 약

최근 지능형 서비스 로봇을 유비쿼터스 환경에 적용하기 위해 다른 장치들과 연동시키는 기술이 부각되고 있다. 이에 본 논문에서는 로봇 플랫폼과 안드로이드 플랫폼 간의 연동을 통해 로봇과 안드로이드 기반의 협력적인 서비스를 사용자에게 제공할 수 있는 소프트웨어 프레임워크 기법을 제시한다. 그리고 본문에서 소프트웨어 프레임워크의 각 구성요소를 설명하고 동작 예시를 통해 실제 동작하는 방식을 기술한다.

1. 서론

기존의 로봇 기술이 산업에서의 생산성 향상을 위해 단순 정밀 작업에 대한 하드웨어 제어를 중심으로 연구가 진행되어 왔다면 근래에는 일상에서 사람과 함께 생활하며 사람에게 유용한 서비스를 제공하기 위해 지능형 서비스 로봇[1]에 대한 연구가 진행되고 있다[2]. 그리고 최근에는 이러한 지능형 서비스 로봇을 유비쿼터스 환경에 적용하여 다른 장치들과 연동시키는 기술이 요구되고 있다.

이에 본 논문에서는 로봇과 안드로이드 장치의 연동 소프트웨어 프레임워크 기법을 제시한다. 본 소프트웨어 프레임워크 기법은 안드로이드 플랫폼과 로봇 플랫폼을 서로 연동시켜 안드로이드가 탑재된 장치로 로봇 서비스를 이용하고, 로봇으로 안드로이드가 탑재된 장치의 안드로이드 서비스를 이용할 수 있게 한다. 그리고 궁극적으로 본 소프트웨어 프레임워크를 적용한 로봇과 안드로이드 기반 장치 간에 상호 협력적인 서비스를 사용자에게 제공한다.

본 소프트웨어 프레임워크는 로봇 플랫폼과 연동하여 동작하는 안드로이드 플랫폼의 어플리케이션을 위해 직관적(intuition)이고 사용하기 쉬운(easy-to-use) 프로그래밍 인터페이스를 제공하며, 로봇 플랫폼과 안드로이드 플랫폼 간의 통신 인터페이스와 매커니즘을 제시한다. 더불어 안드로이드 플랫폼과 연동하여 동작하는 로봇의 서비스 관련 태스크 관리 기법과 본 소프트웨어 프레임워크가 로봇의 HW 플랫폼에 구애받지 않고 동작하도록 하는 로봇 HW 추상화 계층을 제공한다.

본 논문의 구성은 다음과 같다. 2 장에서는 관련 연

구에 대해서 살펴보고, 3 장에서는 제시하는 로봇 플랫폼과 안드로이드 플랫폼 간의 연동 소프트웨어 프레임워크에 대해 설명한다. 4 장에서는 동작 예시에 대해 기술하며 5 장에서 향후 연구 계획과 함께 결론을 맺는다.

2. 관련연구

2.1. URC 로봇 원격 모니터링 기술

URC 로봇 원격 모니터링 기술[4]은 실시간 운영체제인 Q+와 HTTP 웹 서버인 BOA 를 기반으로 하는 로봇 모니터링 및 원격 제어 기술이다. 이 기술은 원격지의 웹 브라우저를 통해 로봇의 카메라와 액츄에이터 등의 제어 장치 제어, 로봇의 내부 자원 및 상태 모니터링, 영상 로깅 및 탐색 등의 작업을 가능하게 한다. 하지만 사용자는 클라이언트인 웹 브라우저를 통해 서버인 로봇이 제공하는 서비스만을 이용할 수 있다는 한계를 가지고 있다.

2.2. RUPI 서버 소프트웨어 프레임워크

RUPI 기반 서버 소프트웨어 프레임워크[5]는 상이한 로봇 클라이언트 플랫폼이 RUPI 서버와 연동하여 다양한 로봇 서비스들을 수행하도록 지원하는 네트워크 기반 로봇 소프트웨어 프레임워크 규격이다. 이 소프트웨어 프레임워크는 원격에서 로봇 클라이언트의 상태를 진단하고 복구하는 1) 로봇 클라이언트 관리 기술, 로봇 클라이언트에서 수행되기 어려운 응용 컴포넌트를 로봇 클라이언트를 대신하여 적재, 관리, 실행하는 2) 로봇 클라이언트 응용 관리기술, 상황인지 기반의 원격 로봇 응용 개발 및 운용을 위한 3)

원격 로봇 응용 플랫폼 기술, 다수의 이중 로봇 클라이언트가 안정적으로 접속할 수 있도록 고가용성을 보장하는 4) RUPI 서버 가용성 기술로 구성된다. 운영 체제 및 프로그래밍 언어 독립적으로 동작하는 것을 목표로 하고 있다.

2.3. WIPI 기반의 지능형 홈서비스 로봇의 구현

WIPI 기반의 지능형 홈서비스 로봇[6]은 무선 인터넷 플랫폼인 WIPI의 어플리케이션을 통해 사용자로부터 화재 및 가스누출 등의 위험을 통보 받을 수 있도록 하는 서비스 로봇이다. 구성은 위험을 감지하고 사용자의 요구대로 움직이는 1) 로봇부, 위험에 대한 정보를 로봇으로부터 사용자에게 제공해주는 2) 모바일부, 로봇과 모바일 간의 통신을 위한 3) 미들웨어부로 이루어진다. 하지만 WIPI 기반 모바일 장치의 단순한 로봇 클라이언트 기능으로는 사용자에게 로봇의 기능을 활용한 제한적인 서비스만을 제공할 수 있다는 한계를 가지고 있다.

3. 로봇 플랫폼과 안드로이드 플랫폼 간의 연동 소프트웨어 프레임워크

3.1. 개요

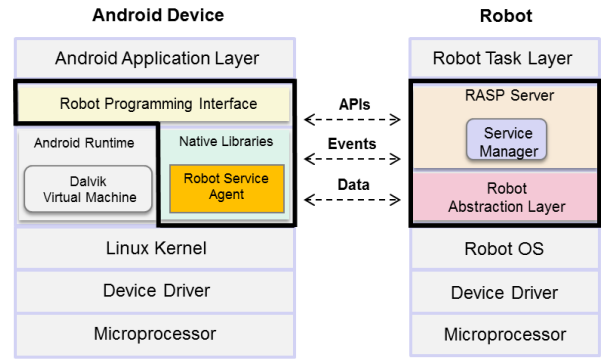
본 장에서 소개하는 로봇 플랫폼과 안드로이드 플랫폼 간의 연동 소프트웨어 프레임워크는 로봇 소프트웨어 플랫폼과 안드로이드 플랫폼을 연동시켜 로봇과 안드로이드 기반 장치가 상호 협력적인 서비스를 제공하도록 하는 기법이다.

본 소프트웨어 프레임워크는 (그림 1)와 같은 구조를 가진다. 안드로이드 장치와 로봇은 제시하는 소프트웨어 프레임워크 기법을 사용하여 안드로이드와 로봇에서 제공하는 서비스 관련 API와 이벤트, 그리고 데이터를 서로 교환할 수 있으며, 다수의 로봇 대 다수의 안드로이드 장치들 간에 연동을 지원한다. 본 소프트웨어 프레임워크를 구성하는 각각의 컴포넌트들은 안드로이드 플랫폼의 어플리케이션을 위한 API (Application Programming Interface), 안드로이드 장치가 로봇과 통신하고 서비스를 원격으로 요청/제공하기 위한 클라이언트 에이전트, 로봇이 안드로이드 장치와의 연결과 서비스 관련 태스크를 관리하기 위한 서버 관리자, 본 소프트웨어 프레임워크가 로봇 HW 플랫폼 독립적으로 동작하기 위한 추상화 계층으로 구분된다.

본 소프트웨어 프레임워크는 궁극적으로 로봇 플랫폼과 안드로이드 플랫폼 간의 연동을 통해 상호 협력적인 서비스를 사용자에게 제공함을 목표로 한다.

3.2. 구성요소

본 절에서는 (그림 1)에서 표현하는 로봇 플랫폼과 안드로이드 플랫폼 간의 연동 소프트웨어 프레임워크의 각 구성요소를 소개한다.



(그림 1) 소프트웨어 프레임워크

- **RPI (Robot Programming Interface):** 안드로이드 어플리케이션에서 로봇 서비스를 원격으로 이용하기 위한 API (Application Programming Interface)이다. 안드로이드 기반의 Java method 로서 제공되며, 안드로이드 어플리케이션으로 직접 노출되는 컴포넌트이다.
- **RSA (Robot Service Agent):** 안드로이드 어플리케이션이 로봇과 통신하고 서비스를 원격으로 요청/제공하기 위한 에이전트로서 RPC (Remote Procedure Call) 매커니즘의 Stub 이자 안드로이드 기반 장치의 태스크 관리자 역할을 한다. 본 컴포넌트는 안드로이드 어플리케이션이 로봇 서비스를 원격으로 이용하기 위한 런타임이기 때문에 빠른 처리 속도를 위해 JNI (Java Native Interface)[3]를 이용하여 C 언어로 설계되었고, 로봇으로부터의 안드로이드 장치에 대한 서비스 요청과 로봇 이벤트 감지 및 데이터 교환을 위해 별도의 태스크를 생성하기도 한다.
- **Service Manager:** 안드로이드 어플리케이션으로부터 로봇 서비스와 관련한 요청을 받고 이를 처리하기 위한 연결과 태스크의 매니저로서 RPC (Remote Procedure Call) 매커니즘의 Skeleton 역할을 한다. 본 컴포넌트는 로봇이 안드로이드 어플리케이션과 통신하기 위한 백그라운드 프로세스로 설계되었고, 로봇 서비스와 관련한 태스크를 생성하고 관리한다.
- **RAL (Robot Abstraction Layer):** 본 소프트웨어 프레임워크가 로봇 HW 플랫폼 독립적으로 동작하기 위한 로봇 HW 추상화 계층으로서 단일화된 인터페이스를 제공한다.

3.3. 로봇 서비스

본 절에서는 제시하는 소프트웨어 프레임워크가 지원하는 로봇 서비스를 <표 1>과 같이 분류하였다.

<표 1> 로봇 서비스 분류

정보 서비스	멀티미디어 서비스	제어 서비스	이벤트 서비스
- 로봇 센서 정보 - 로봇 배터리 정보 - 로봇 액추에이터 정보 - 로봇 LED 정보	- 스트리밍 - TTS - 음성 - 영상	- 주행 - 모션	- 센서 알람 - 배터리 알람 - 시간 알람

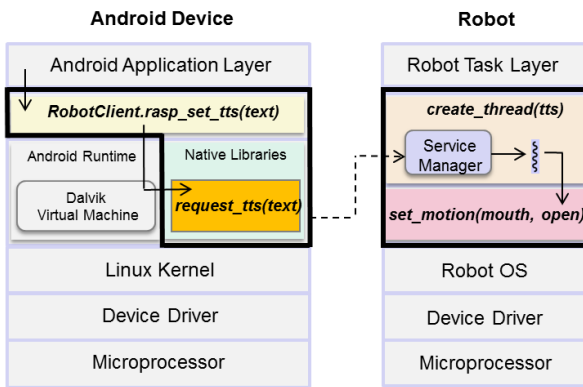
- **정보 서비스:** 단순히 안드로이드 장치의 어플리케이션이 로봇의 현재 상태를 모니터링하기 위해 요청

하는 서비스로서 로봇의 HW 상태 값을 읽어서 제공하는 방식으로 동작한다.

- 멀티미디어 서비스: 안드로이드 장치의 어플리케이션이 로봇을 통해 원격으로 멀티미디어 서비스를 요청하고 제공 받는 방식으로 동작하며, 타 서비스에 비해 시스템 자원을 많이 사용하기 때문에 효과적인 관리 기술이 요구된다.
- 제어 서비스: 안드로이드 장치의 어플리케이션이 실제 로봇의 움직임을 제어하기 위해 요청하는 서비스로서 복잡한 로봇 HW 제어 기술과 다양한 로봇 HW 플랫폼에서 적용될 수 있는 공통의 인터페이스가 요구된다.
- 이벤트 서비스: 안드로이드 장치의 어플리케이션에서 로봇에게 특정 조건의 이벤트에 대한 알람을 요청하는 서비스로서 실시간성과 함께 다양한 이벤트 요청들을 효과적으로 관리할 수 있는 기술이 요구된다.

4. 동작 예시

본 장에서는 제시하는 소프트웨어 프레임워크가 실제 동작하는 방식에 대해 기술한다. 예시로서 안드로이드 장치의 어플리케이션이 로봇에게 TTS (Text to Speech) 서비스를 요청하면 로봇이 실제 말하듯이 입을 벌리면서 수신한 텍스트를 음성으로 변환하여 출력하는 상황을 설정하였다.



(그림 2) TTS 서비스 동작 예시

상기 설정한 TTS (Text to Speech) 서비스 요청 예시는 (그림 2)과 같이 동작하며, 동작 흐름은 다음과 같다.

- ① 안드로이드 장치의 어플리케이션은 로봇에게 TTS (Text to Speech) 서비스를 요청하기 위해 RobotClient 라는 Java 클래스의 Robot Programming Interface 인 rasp_set_tts() method 를 호출한다.
- ② rasp_set_tts() method 는 Dalvik Virtual machine 을 통해 JNI (Java Native Interface)로 매핑된 C 언어 함수 request_tts()를 Robot Client Agent 로서 호출한다.
- ③ request_tts() 함수는 요청하는 TTS 서비스 관련 데이터를 마샬링(marshaling)하고, 로봇의 Service Manager 에게 서비스 요청 메시지를 전송한다.
- ④ 메시지를 수신한 Service Manager 는 요청 받은 서

비스가 TTS 라는 것을 인식하고 태스크를 수행할 담당 쓰레드를 생성한다. 만약 이미 TTS 서비스를 수행할 담당 쓰레드가 존재한다면 해당 쓰레드에게 태스크 요청을 알린다.

- ⑤ TTS 서비스를 수행하는 쓰레드는 텍스트를 음성으로 변환하여 출력하고, 동시에 Robot Abstraction Layer 를 통해 set_motion() 함수를 호출하여 로봇의 입 동작을 제어한다.

5. 결론

본 논문에서 제시한 로봇과 안드로이드 장치의 연동 소프트웨어 프레임워크는 안드로이드 플랫폼과 로봇 플랫폼을 서로 연동시켜 로봇과 안드로이드 기반 장치 간의 상호 협력적인 서비스를 사용자에게 제공하는 기법이다.

본 논문에서는 관련 기술과 함께 소프트웨어 프레임워크 및 각 구성요소들을 소개했고, 로봇이 제공하는 서비스를 분류했으며, 예시를 통해 소프트웨어 프레임워크의 동작 흐름을 기술했다.

향후 스마트 장치의 소프트웨어 플랫폼으로서 안드로이드 뿐만 아니라 애플의 iOS 와 MS 의 Windows Phone 7 에도 본 소프트웨어 프레임워크를 적용하는 연구가 요구된다. 그리고 더 나아가 로봇을 스마트 디바이스이자 유비쿼터스 장치로서 실생활에서의 활용도를 높이기 위해 안드로이드 플랫폼을 로봇에 직접 도입하는 연구가 요구된다.

참고문헌

- [1] Robert Todd pack, "IMA: The Intelligent Machine Architecture," PhD Thesis, Vanderbilt University, Nashville, Tennessee, 2003.
- [2] 김혜진, 이재연, "지능형 서비스 로봇으로의 이동," 정보통신연구진흥원 학술정보 주간기술동향 1181호, 2005.
- [3] Sheng Liang, "The Java native interface: programmer's guide and specification," Sun Microsystems, 1999
- [4] 김주만, "URC 로봇 원격 모니터링기술 개발," 한국콘텐츠학회논문지 '06 Vol.6 No.8, pp 8-19, 2006
- [5] 김록원, 이강우, 서영호, 김형신, "RUPI 서버 소프트웨어 프레임워크," 한국통신학회지 제 25 권 제 4 호, pp 2-86, 2008
- [6] 김진환, "WIPI 기반의 지능형 홈서비스 로봇의 구현," 한국콘텐츠학회논문지 제 8 권 제 5 호, pp 19-28, 2008.