

# 안드로이드 기반의 PC 원격제어 시스템 개발을 위한 기법 분석

김지원, \*윤은일  
충북대학교 전자정보대학 컴퓨터과학과  
e-mail : {jwonkim, yunej}@chungbuk.ac.kr

## Technique Analysis for Developing PC Remote Control System based on Android Platform

Jiwon Kim, Unil Yun  
Department of Computer Science, College of Electrical & Computer Engineering, Chungbuk National University

### 요 약

안드로이드 OS가 공개된 이후, 모바일 기기들의 발달이 가속화됨에 따라 다양한 모바일 기기들이 등장했다. 이와 더불어 다양한 안드로이드 애플리케이션이 개발되었고 사용자들은 PC의 전유물이던 인터넷 서핑, 채팅, 게임 등의 기능들을 애플리케이션을 통해 모바일 기기에서 제공받을 수 있게 되었다. 특히, 사용자 편의성을 위해 PC들 사이에 이루어지던 원격제어는 PC와 모바일 기기 사이의 원격제어로 발전해 사용자들에게 제공되고 있다. 본 논문은 안드로이드 기반의 모바일 기기와 PC간에 이용되는 원격제어 시스템 개발에 관한 기술들에 대해 분석한다.

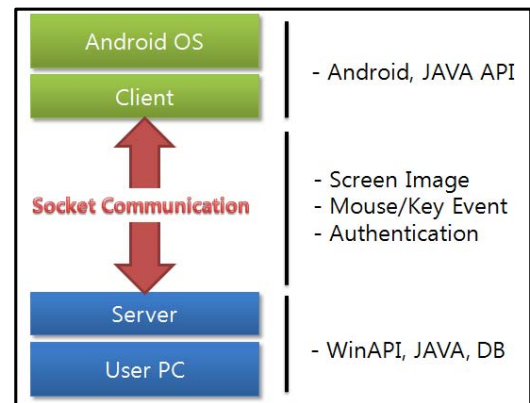
### 1. 서 론

Google사의 안드로이드 플랫폼 공개 후, 모바일 시장의 경쟁은 더욱 가속화 되었고 결과적으로 안드로이드 기반의 모바일 기기들의 등장하는 계기가 되었다. 이와 더불어 다양한 안드로이드 애플리케이션들이 개발되었으며 대부분은 PC에서 이용되던 소프트웨어들을 모티브로 한 것이었다. 안드로이드 애플리케이션은 모바일의 특성상 사용자 편의성을 극대화 시킬 수 있는 시스템들이 주로 개발되었다. 특히, 그 중에서도 PC원격제어 시스템은 많은 사용자들의 관심을 받으며 활용되고 있다. 최근의 원격제어 시스템은 과거 PC들 사이에 이루어지던 원격제어를 모바일로 확대하여 PC-모바일 사이의 원격제어의 형태를 가지게 한 것으로, 이는 사용자가 모바일기기를 통해 PC와 같은 성능을 낼 수 있다는 것을 의미할 뿐 아니라 언제 어디서든 인터넷에만 접속이 되어 있으면 PC조작이 가능하다는 특징을 가진다. 그래서 현재 안드로이드 기반 원격제어 시스템에 대해서 다양한 연구와 개발이 진행되고 있다. 또한, 시스템의 기반이 되는 안드로이드 플랫폼은 OHA(Open Handset Alliance)의 개방형 정책[9]과 함께 수많은 개발자들을 통해 발전되어 왔으며, 이로 인해 안드로이드 플랫폼에서는 다양한 API들을 계속 새롭게 제공하고 있는 상황이다. 이렇게 발전하는 시대의 특성에 맞추어 본 논문에서는 안드로이드 기반의 원격제어 시스템을 더 효율적으로 개발하기 위해 현재 안드로이드 기반의 원격제어 시스템 개발 시, 자주 사용되는 기법들을 몇 가지 기능을 기준으로 나누어 분석한다.

\* 교신저자 : 윤은일

이 논문은 2012년도 정부 교육과학기술부의 재원으로 한국연구재단의 지원을 받아 수행된 연구사업임(NRF No. 2012-0003740 and 2012-0000478).

### 2. 원격제어 시스템의 개발 기법 분석



(그림 1) 원격제어 시스템 구조도

안드로이드 플랫폼에서 원격지의 컴퓨터를 제어하기 위해서는 여러 기술들이 필요하다. 위의 그림1을 통해 살펴보면 데이터 교환을 위해 컴퓨터와 연결하는 기술(그림1은 소켓통신 이용)과 연결된 컴퓨터의 입력장치들(마우스, 키보드 등)을 제어할 수 있는 기술, 추가적으로 원격지의 상황을 알 수 있는 화면전송기술 등이 필요하다. 본 장에서는 이러한 기술들이 안드로이드 플랫폼에서 어떻게 제공되고 사용되는지를 분석한다.

#### 2.1 연결 기술

안드로이드 모바일 기기(이하 안드로이드 기기)에서 컴퓨터와 연결하는 기술은 크게 유선 및 무선기술로 나눌 수 있다. 현재, 유선을 통해 안드로이드 기기와 컴퓨터를 연결하는 방법은 USB 인터페이스를 이용하는 것이다. 안드로이드 기기의 경우 MINI-USB 또는 MICRO-USB 포트가 장착되어 있다.[4] USB 포트들은 기기에 전력을 공급하는 통로가 되며 연결된 기기들 사이에 파일전송을 할

수 있게 도와준다. 또한, 안드로이드의 버전이 향상됨에 따라 해당 포트를 이용해 다른 기기들(카메라, 키보드, 마우스 등)과 연결하여 통신을 가능하게 해주는 프로토콜(안드로이드 버전 3.1부터 기본 API로 제공되며 2.3.4부터는 추가 라이브설치 필요)을 제공한다.[8] 즉, USB 인터페이스를 활용하면 핸드셰이크(HandShake) 과정의 구축을 통해 안드로이드 기기와 컴퓨터가 통신이 가능하게 되어 원격제어 시스템 구현 활용하면 . 또, 현재 안드로이드 기기에서 지원하는 다양한 무선제공한다(블루투스, 3G, Wi-Fi, Wi-Fi Direct 등)을 활용하여 소켓통신을 할 수 있다. 소켓통신은 TCP/IP 및 UDP 프로토콜을 지원하는 통신기술이다. 안드로이드 TCP/IP JAVA의 핵심 라이브러리들이 포함된 SDK(Software Development Kit)가 제공되기 때 포함 JAVA의 소켓관련 라이브러리(JavaSocket 포함)를 활용할 수 있다. 이것은 통신을 주목적으로 하는 안드로이드 기기의 특성을 잘 나타낸다. 이러한 JAVA API를 통해 서버-클라이언트 모델의 원격제어 시스템을 구축하게 되면 컴퓨터에 설치된 서버 프로그램은 JVM(Java Virtual Machine)을 통해 실행되는 동안 소켓을 열어 안드로이드 기기에 설치된 클라이언트의 접속요청을 기다린다. 이 때, 스레드를 통해 다수의 클라이언트가 서버에 접속하는 것도 가능하다. 클라이언트가 서버의 소켓에 접속하면 클라이언트의 접속종료시점까지 서로 메시지를 교환하며 원격제어를 수행하게 된다.

## 2.2 화면전송

안드로이드 기기에서 원격제어 시스템 사용 시, 원격지와 멀리 떨어져 있을 경우 원격지 컴퓨터의 화면을 볼 수 없는 경우가 발생하게 된다. 이러한 사용자의 편의성 하락을 방지하고 효율적인 원격제어를 위해서 필요한 것이 화면전송 기능이다. 화면을 전송하기 위해 사용되는 기본 아이디어는 화면을 캡처하여 전송하는 것이다. 서버-클라이언트 모델 관점에서 살펴보면, 서버 프로그램은 OS에 따라 API를 활용해 주기적으로 컴퓨터의 화면 이미지를 그림파일로 추출하여 클라이언트로 전송한다.(예로 윈도우 OS의 경우 WinAPI를 이용하여 현재 화면을 HBITMAP 형태의 bmp파일로 추출할 수 있다) 이 때, 추출된 그림파일을 압축할 경우 파일의 전송시간이 감소하게 된다. 결과적으로 클라이언트 측에서는 실시간에 가깝게 원격지 컴퓨터의 화면을 볼 수 있게 된다. 예로, VNC(Virtual Network Computing)라는 시스템이 있다. VNC는 RFB(Remote Frame-Buffer) 프로토콜을 이용하여 연결된 두 기기의 화면을 공유하는 원격제어 시스템이다. VNC는 서버의 화면을 암호화하여 클라이언트로 전송하고 클라이언트에서는 암호화된 데이터를 복호화 한다. 이 때, RFB 프로토콜에서는 RRE, CopyRect, Hextile, ZRLE 등 여러 방법을 통해 암호화를 수행할 수 있다.[2] 대부분 변경된 부분을 최소화시켜 전송되는 파일의 용량을 줄이는데 그 목적이 있다. 클라이언트 환경에서 좋은 품질의 화면을 보고 싶다면 서버 프로그램의 사용 환경(OS, 인터넷속도, 화면 해상도 등)에 따라 다른 방법을 이용해야 한다.

## 2.3 입력장치(마우스, 키보드) 제어

입력장치의 제어는 원격제어 시스템에서 사용자들이 주

로 사용하는 기능으로 관련기술을 크게 두 부분으로 나누어 분석한다. 첫 번째는 안드로이드 기기 상에서의 이벤트 분석 기법이다. 사용자는 클라이언트 애플리케이션을 통해 서버에 이벤트 메시지를 송신한다. 안드로이드 플랫폼은 이러한 이벤트를 처리하기 위해 각종 인터페이스를 제공한다. 그 예로 안드로이드 플랫폼에는 MotionEvent, OnClickListener, OnKeyListener 등과 같은 인터페이스들이 있으며 이를 활용해 각종 이벤트 정보를 서버로 보낼 수 있다. 두 번째로 서버 프로그램에서 전달 받은 이벤트 값들을 이용해 처리하는 기법이다. 이 기법은 서버가 운용되는 컴퓨터의 운영체제와 서버 프로그램의 제작언어에 따라 다르게 이용된다. 먼저 윈도우즈의 경우에는 WinAPI를 운영체제에서 제공한다. 하지만 리눅스의 경우에는 직접 마우스 관련 프로토콜(PS/2, IMPS/2 등)[3]을 이용하여 제어를 해야 한다. 또한 프로그래밍 언어별로 마우스 및 키보드 이벤트 제어에 관한 인터페이스를 제공하기도 한다. 예를 들어, JAVA의 경우에는 AWT 라이브러리의 Robot 클래스를 이용하여 마우스 제어가 가능하다. 마우스 및 키보드 이외의 입력장치는 PC를 통해 내부적으로 제어가 가능한 것들이 많기 때문에 본 논문에서는 분석하지 않는다.

## 3. 결론

본 논문에서는 안드로이드 기반의 모바일 기기에서 PC를 제어할 수 있는 원격제어 시스템을 개발하기 위한 기법들에 대해 기능적 관점으로 다양하게 분석했다. 추후 연구로는, 본 논문에서 분석한 개발 기법들을 활용해 안드로이드 기기에서 PC를 원격제어 하는 시스템을 직접 개발하고, 추가로 발생할 수 있는 문제점들(예로 보안 및 호환성 관련 문제)에 대한 해결방법을 찾는 방향으로 진행할 것이다.

## 참고문헌

- [1] Guiran Chang, Chunguang Tan, Guanhua Li, Chuan Zhu, "Developing Mobile Applications on the Android Platform", Book: Mobile Multimedia Processing, pp.264-286, 2010.
- [2] Tristan Richardson, "The RFB Protocol", RealVNC Ltd, version 3.8, 2010.
- [3] Roderick W. Smith, "LPIC-1: Linux Professional Institute Certification Study Guide", John Wiley&Sons, 2009.
- [4] Angel Gonzalez Villan, Josep Jorba Esteve, "Remote Control of Mobile Devices in Android Platform", IEEE transactions on mobile computing, pp.1-9, 2011.
- [5] 이문구, 박성원, "실시간 무선 원격제어 시스템의 설계 및 구현", 『전자공학회지』 제26권 1호, pp.1589-1592, 2003.
- [6] 이문구, "실시간 무선 원격 제어 시스템에 관한 연구", 『전자공학회지』 제46권 CI편 제6호, pp.63-69, 2009.
- [7] 이자인, 최경희, 정기현, "안드로이드 디바이스의 효과적인 활용을 위한 wifi기반의 원격제어 시스템", 『한국컴퓨터종합학술대회 논문집』 제39권 1호, pp.328-330, 2012.
- [8] Android Developers, "USB Host and Accessory", <http://developer.android.com/guide/topics/usb>, 2011.
- [9] OHA, "Industry Leaders Announce Open Platform for Mobile Devices." <http://www.openhandsetalliance.com/>, 2007.