

## Android 가상머신을 위한 디바이스 드라이버 구현

김호성<sup>1</sup> · 서종균<sup>2</sup> · 박한수<sup>2</sup> · 정희경<sup>1\*</sup>

### Implementation of Device Driver for Virtual Machine Based-on Android

Ho-Sung Kim<sup>1</sup> · Jong-Kyoun Seo<sup>2</sup> · Han-Su Park<sup>2</sup> · Hoe-Kyung Jung<sup>1\*</sup>

<sup>1</sup>Department of Computer Engineering, Paichai University, Daejeon 302-735, Korea

<sup>2</sup>UbiTech Co.,Ltd, Daejeon 305-340, Korea

#### 요 약

스마트폰의 보급량은 기하급수적으로 증가하고 있지만, 주기적인 고성능 스마트폰 및 업그레이드된 운영체제의 출시로 인하여 1년~2년 사이에 구형 스마트폰이 된다. 이러한 스마트폰의 환경적인 제약 부분을 해결하기 위하여 Thin-Client 단말을 이용한 가상화 기술이 개발되고 있다. 하지만 가상머신(VM: Virtual Machine)의 경우 센서 및 GPS 장치가 장착되지 않아 센서와 GPS 장치와 관련된 응용프로그램을 구동할 수 없다.

본 논문에서 x86 기반의 시스템에서 가상 머신으로 동작하는 Android용 가상머신의 디바이스 드라이버를 구현하여 가상머신 환경에서 최신 스마트폰을 사용하는 것 같은 Android 가상화 기능을 제공하고자 한다. 가상 디바이스 드라이버는 실제로 동작하는 구형 Android 스마트 폰(Thin-Client)으로부터 센서 및 GPS 정보를 수신하여 가상 장치에 실제 디바이스가 있는 것처럼 동작하는 방법을 제안하고자 한다.

#### ABSTRACT

The amount of smart phones has increased exponentially. Due to the periodic release of high-performance smart phones and upgraded operating system, new smart phones become out-dated over 1 or 2 years. In order to solve environmental constraints of these smart phones, virtualization technology using Thin-Client terminal has been developed. However, in the case of Virtual Machine(VM), the applications associated with sensors and a GPS device can not run because they are not included.

In this paper, by implementing the device driver for Android running in a virtual machine in the x86-based systems, it is to provide Android virtualization capabilities such as using the latest smart phones in the virtual machine environment. It would like to propose a method that the virtual device driver receives sensors and GPS information from the old Android smart phones(Thin-Client) that actually work and run as if the real device exists.

**키워드** : 가상 디바이스 드라이버, 가상 머신, 센서, 안드로이드, GPS

**Key word** : Virtual Device Driver, Virtual Machine, Sensor, Android, GPS

Received 02 January 2015, Revised 29 January 2015, Accepted 16 February 2015

\* Corresponding Author Hoe-Kyung Jung(E-mail:hkjung@pcu.ac.kr, Tel:+82-42-520-5640)

Department of Computer Engineering, Paichai University, Daejeon 302-735, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2015.19.4.1017>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.  
Copyright © The Korea Institute of Information and Communication Engineering.

## I. 서 론

전세계 스마트폰의 판매량은 꾸준히 증가하고 있다. Gartner 자료에 의하면, 2013년도까지 스마트폰 판매량은 968만대로 전년도 대비 42.3% 판매량이 증가되었고, 올해 안에 스마트폰 판매량이 10억대에 도달할 것으로 예상하고 있다[1]. 판매되는 스마트폰의 OS별 점유율은 Android가 약 85% 정도를 차지하고 있으며, 국내의 경우 거의 90%를 차지하고 있다[2,3].

Android OS는 주기적인 업데이트가 이루어지고 있지만, 기존 사용자들이 직접 Android OS를 업데이트 하기에는 상당한 어려움이 발생한다. 또한 제조사의 새로운 단말 출시는 고사양의 H/W를 적용하고 고성능 3D 게임 등을 지원함으로써 기존에 사용하던 스마트폰은 점차 구형 스마트폰이 되어 고사양을 지원하는 어플리케이션을 사용하지 못하게 되는 문제점이 발생하게 된다. 그리고, IDC 모바일 연구팀의 보고서에 따르면 신흥 시장에서 저가형 스마트폰이 성과를 내고 있으며, 저가형 스마트폰 시장이 앞으로도 계속 성장할 것이라고 예상하였다. 하지만 저가형 스마트폰의 경우 제한된 H/W 성능으로 고성능 어플리케이션 구동에 제약사항이 발생한다[4-7].

본 논문에서는 위에서 언급한 H/W 및 S/W 환경에 제약 없이 Thin-Client(저가형 스마트폰 또는 Low Android OS 버전)에서도 사용자에게 최신 OS 버전에서 제공되는 고성능 어플리케이션 구동이 가능하도록 모바일 클라우드 기반 가상화 서비스를 제공하기 위한 방법과, 단순한 Android OS 환경뿐만 아니라 센서 및 GPS등의 가상 디바이스 드라이버 구현 방법을 제안한다.

## II. 관련연구

모바일 클라우드 기반 가상화 서비스 기술은 스마트 단말에 가상화 기술을 적용하는 방법과 클라우드 상에서 가상 단말 또는 에뮬레이터를 제공하는 경우로 나누어진다. 단말에 적용하는 가상화 기술은 스마트 단말에 다수의 OS를 동작시키는 기술로 Red Bend, VMware Mobile Virtual Platform(MVP), Motorola Atrix 4G 등이 이에 속한다.

단말에 가상화 기술을 사용하는 VMware Mobile Virtual Platform의 경우 스마트 단말에서 두 개의 OS를 구동할 수 있도록 하는 기술이다. 각각의 OS는 개인용도(Normal Domain)와 업무 및 사생활 보호용(Secure Domain)으로 나뉘어 별도로 동작함으로써 개인정보 및 회사 업무와 관련된 데이터를 Secure Domain에 저장하여 보안 기능을 강화한다. 이 기술은TYPE1 가상화와 TYPE2 가상화로 분류되며, TYPE1 가상화는 프로세서 모듈에서 제공하는 가상화 기능을 이용하여 구현된 것으로 각각의 OS가 완전히 독립적으로 동작한다. TYPE2 가상화는 기존의 OS위에 가상화 기능을 구현하는 것이다. 그림1은 VMware Mobile Virtual Platform의 구성도를 나타낸다.

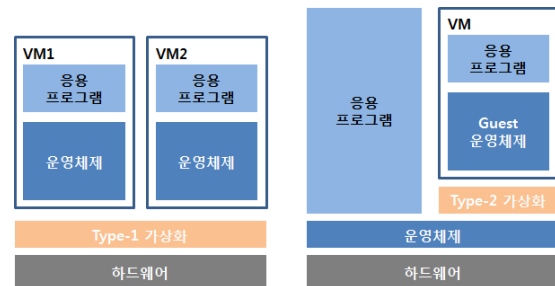


그림 1. VMware MVP  
Fig. 1 VMware MVP

클라우드 상에서 가상 단말 또는 에뮬레이터를 제공하는 기술은 카네기 멜론 대학에서 연구된 Cloudlet상의 서버를 통하여 가상머신을 실행시키고 실제 단말과 연동한다.

Cloudlet은 스마트 단말 H/W의 자원제약, 네트워크 환경의 높은 지연율, 낮은 대역폭 문제를 해결하기 위하여 구현된 기술이다. 스마트 단말에서 직접적으로 클라우드 환경과 연결되지 않고 Cloudlet과 연결된다. 이때 네트워크 환경은 무선 환경으로 구성되며, Cloudlet 은 낮은 지연율, 높은 대역폭을 보장하고 풍부한 H/W 리소스 자원을 제공한다.

Cloudlet은 기본VM과 오버레이VM으로 구성되며, 기본 VM은 Cloudlet에 동작하고 오버레이 VM은 스마트 단말에서 동작한다. Cloudlet의 개요는 아래 그림 2와 같다.

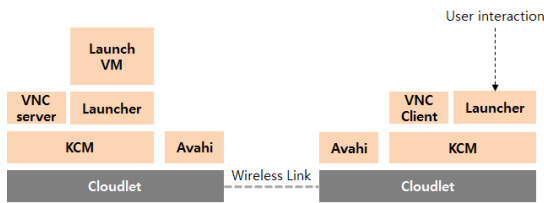


그림 2. Cloudlet 개요  
Fig. 2 Overview of the Cloudlet

### III. 시스템 설계

본 논문에서는 클라우드 상에서 가상 단말을 구현하고, 실제 스마트폰에 클라이언트를 구현한다.

클라이언트는 센서 및 GPS 값을 가상 단말의 서버 프로그램으로 주기적(100ms)으로 전달하게 되며, 가상 단말에는 센서 및 GPS 값을 처리하는 서버 프로그램, 가상 디바이스 드라이버 및 가상 HAL 라이브러리를 구현한다. 이는 저가형 스마트폰 및 노후화된 스마트폰의 S/W와 H/W 특성에 제한받지 않고 고성능 앱 콘텐츠를 원활하게 이용할 수 있도록 할 수 있다. 전체 시스템의 구성도는 그림 3과 같다.

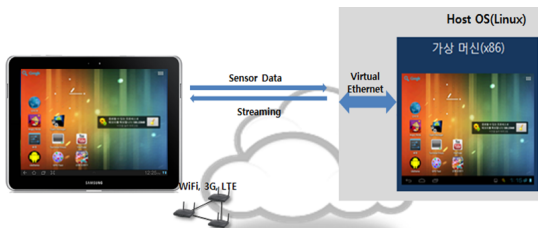


그림 3. Android 가상머신 구조도  
Fig. 3 Android Virtual machine Structure

본 장에서는 x86기반의 Android 가상머신에서 동작하는 서버 프로그램, 가상 디바이스 드라이버, 가상 HAL 라이브러리 및 실제 스마트폰에서 구성되는 클라이언트의 구현에 관한 내용을 기술한다.

실제 스마트폰과 x86 기반의 Android OS 와 연동하기 위해서는 가상 라이브러리와 가상의 디바이스 드라이버를 구현해줘야 한다. Android 의 경우 센서 및 GPS 값은 HAL을 통해서 Android Framework에 전달하게 되는데, x86 기반의 Android 의 경우 실제 센서 및 GPS

물리장치가 없기 때문에 장치들이 있는 것처럼 인식하도록 해야 하기 때문이다. x86 기반의 Android의 전체 S/W 구조와 가상 드라이버가 추가된 구조는 그림 4와 같다.

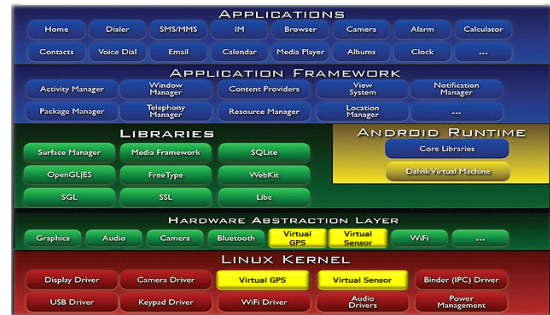


그림 4. Android 구조도  
Fig. 4 Android Structure

가상 디바이스에 대한 드라이버를 리눅스 커널에 구현하고 Android와의 연결을 위하여 HAL에 라이브러리를 구현 하며, 실제 스마트폰에서 데이터를 송수신 할 수 있는 클라이언트 프로그램과 서버프로그램을 구현한다. 본 논문에서 가상 머신과 실제 스마트폰의 동작 구조는 그림 5와 같다.

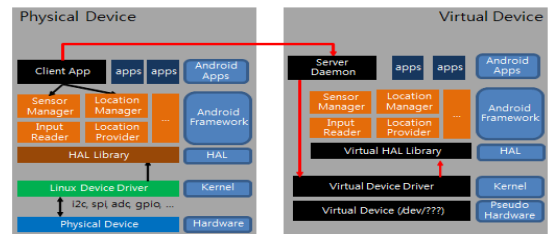


그림 5. Android 가상 디바이스 드라이버  
Fig. 5 Android Virtual Devices Driver

본 논문에서 구현하는 프로그램은 다음과 같다.

- 클라이언트 프로그램
- 서버 프로그램
- 가상 디바이스 드라이버
- 가상 라이브러리

클라이언트 프로그램은 스마트폰에서 동작하는 프로그램이며, 가상머신에서 필요로 하는 센서 및 GPS 값을 캡처해서 가상 머신으로 전달하는 역할을 한다(그림 6).

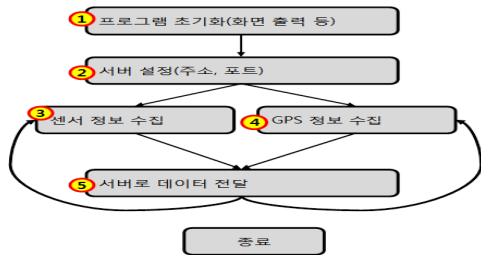


그림 6. 클라이언트 프로그램 동작 절차  
Fig. 6 Client Program operation process

서버 프로그램은 스마트폰의 클라이언트 프로그램으로부터 센서 및 GPS 값을 수신하는 역할을 한다. Daemon 형태로 동작하고 있으며, Android 부팅 시 자동으로 실행되어 클라이언트 프로그램의 연결을 대기한다. 또한 수신된 값들을 파싱(Parsing) 해서 알맞은 가상 디바이스 드라이버로 데이터를 전달하는 역할을 한다(그림 7).

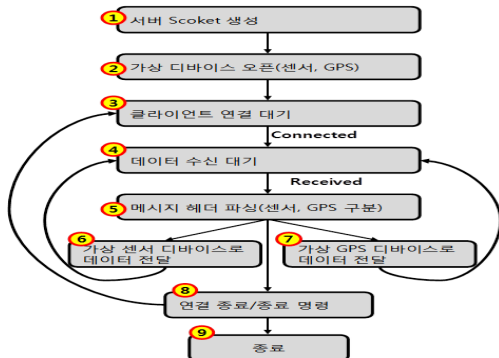


그림 7. 서버 프로그램 동작 절차  
Fig. 7 Server Program operation process

가상 디바이스 드라이버는 리눅스 커널에 포함된 디바이스 드라이버이며, 서버 프로그램과 HAL 라이브러리에서 접근한다. Server 프로그램에서 클라이언트 프로그램으로부터 데이터를 수신하면 적절한 파싱을 거쳐서 해당 디바이스 드라이버에게 전달한다. 디바이스 드라이버는 수신된 값을 바로 HAL에 전달하거나 HAL에서 가져갈 수 있는 준비를 해놓고 기다린다. 이러한 것은 HAL 동작하는 방식에 따라 달라진다. 또한 x86 기반의 Android의 경우 실제 센서 및 GPS 장치가 없지만 해당 H/W가 있는 것처럼 인식하기 위하여 “/dev/pxxx” 노드 장치 파일을 생성한다(그림 8).

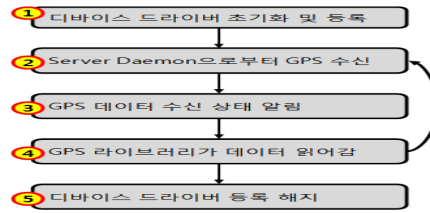


그림 8. GPS 디바이스 드라이버 동작 절차  
Fig. 8 GPS device driver operation process

가상 라이브러리는 HAL에 존재하며, Android는 디바이스 드라이버를 직접 접근하기보다는 다양한 H/W를 접근할 수 있는 공통 메커니즘을 제공한다(그림 9).

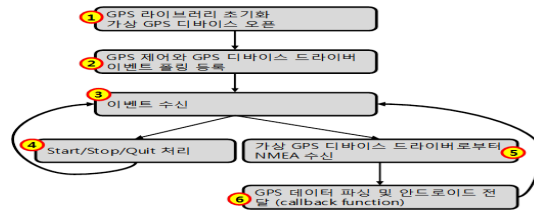


그림 9. GPS 라이브러리 동작 절차  
Fig. 9 GPS library operation process

본 논문에서는 기존의 앱 콘텐츠들 수정 없이 연동될 수 있도록 해당 디바이스 드라이버에 대한 가상 라이브러리를 추가한다.

#### IV. 시스템 구현 및 성능평가

본 논문에서 Android 가상 디바이스 드라이버 구현에 사용된 개발환경의 운영체제로는 Ubuntu 12.04 LTS, 클라이언트로는 Google Nexus7, 서버로는 인텔 코어 i7, 가상 머신으로 오라클 Virtual Box, 개발 언어로는 C와 Java를 이용하였다.

Google Nexus7에서는 100ms 단위로 센서 및 GPS 값을 보내주기 위한 클라이언트 프로그램을 구현하였고, 가상 단말에서는 센서 및 GPS 값을 수신하기 위한 서버 프로그램을 구현하였다. 이때 양쪽의 통신은 TCP 소켓 통신을 이용하여 데이터를 송수신한다. Android OS와 센서 및 GPS를 사용하는 응용프로그램과 연동하기 위하여 가상 HAL 라이브러리와 가상 디바이스 드라이버를 추가로 구현하였다.

가상 단말의 가상 디바이스 드라이버들이 센서 및 GPS 장치들을 제어할 수 있도록 장치 노드들을 생성한다.

- 가상 GPS : /dev/vgps
- 가상 SENSOR : /dev/vsensor

가상 장치들을 제어할 수 있도록 커널의 센서 및 GPS 에 대한 디바이스 드라이버를 포함하였다. 센서와 GPS 데이터들을 Android 프레임워크에서 읽어 갈 수 있도록 그림 10과 같이 가상 HAL 라이브러리를 추가하였다.

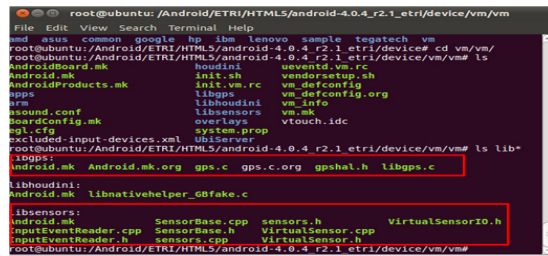


그림 10. 가상 HAL Library  
Fig. 10 Virtual HAL Library

스마트폰에서 클라이언트 프로그램을 실행하면 센서 및 GPS 값을 가상단말 서버 프로그램으로 전달한다. 본 논문에서는 가상단말에서 센서 및 GPS 정보 값을 확인하기 위한 데모프로그램을 구현 하였다. 가상머신 실행 화면을 그림 11에 보이며, 그림 12는 Google Nexus7으로부터 센서 및 GPS 정보값을 받아와 가상머신의 Android 화면에 값을 보여준다.

본 논문에서는 센서 및 GPS 정보 값을 출력하기 위한 예제프로그램을 작성하였다. 프로그램이 시작되면 가상머신은 GPS 정보값 및 마그네틱 센서, 가속도 센서, 방향 센서 값을 수신 받는다.



그림 11. 가상 머신 실행화면  
Fig. 11 Run Screen of Virtual Machine

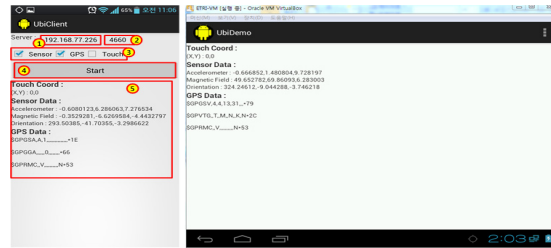


그림 12. UbiDemo 실행화면  
Fig. 12 Run Screen of UbiDemo

본 논문에서 제안한 방식은 클라우드 환경에서 저비용 서비스를 제공하기 위한 것으로 저사양 스마트 단말 및 노후화 스마트 단말 사용자에게 최신 기능을 제공하고자 한다. 또한 가상 디바이스 드라이버 기술을 제공함으로써 마이크로소프트, 구글, 다음, 네이버등 기존에 제공되는 클라우드 서비스와는 다르게 사용자는 가상 단말에서도 센서 및 GPS를 이용하는 기존의 앱 콘텐츠를 수정 없이 사용 가능한 장점이 있다.

## V. 결론 및 향후 연구방향

주기적으로 Android OS 버전이 출시되는 것과는 다르게 시장에 출시된 스마트 단말에 대한 Android OS 업그레이드 지원은 상당히 저조한 편이다. 또한, 최신 발매된 Android OS 버전을 사용자가 업데이트를 하는 것 역시 거의 불가능하며, 최근에 출시되는 중저가 스마트폰의 경우 H/W 성능 제약으로 인하여 고성능 앱 및 멀티미디어 서비스를 원활하게 이용하기 힘들다.

기존에 연구되고 있는 가상머신 VMware Mobile Virtual Platform의 경우 사생활 보호를 위한 두개의 도메인을 사용하거나, cloudnet의 경우 H/W 자원 제약 및 네트워크 성능에 대해서 연구가 진행되고 있다. 하지만, 위에서 언급한 연구들은 센서 및 GPS 장치의 정보 수신에 대한 부분을 고려되지 않고 있다. 따라서, 센서 및 GPS를 이용하는 관련 서비스를 사용하지는 못하였다.

본 논문에서 제안한 방식은 가상머신과 실제 스마트 단말의 정보들 값을 이용하여 성능이 낮은 H/W 및 구형 S/W에서도 최신 Android OS 및 고성능 앱을 구동하도록 하고자 하였다. 최신 Android OS의 경우 풍부한

H/W 리소스 자원을 보유하고 있는 가상 머신에서 동작하도록 하고, 실제 스마트 단말의 주변정보값(센서 및 GPS) 정보를 가상 머신으로 전달하여 Android와 연동이 가능하도록 프로그램을 구현하였다. 그 결과 가상화 서비스를 통하여 고성능 게임, 고화질 미디어, 최신 Android OS를 낮은 H/W 성능 및 구버전 Android OS 버전의 스마트폰에서도 사용 가능하게 되었다.

향후 터치스크린, 영상 전송 기능 기술을 추가하여 MDM(Mobile device Magement)분야로 기술적으로 확장하여 구현할 예정이다.

### 감사의 말

본 논문은 교육부의 ‘산업단지캠퍼스 조성사업’ 국고지원금으로 수행한 산학공동연구과제의 연구 결과입니다.

### REFERENCES

- [1] Gartner web site, “Gartner Says Annual Smartphone Sales Surpassed Sales of Feature Phones for the First Time in 2013”. Available: <http://www.gartner.com/newsroom/id/2665715>, 2014.
- [2] IDC web site, “Smartphone OS Market Share, Q4 2014” Available:<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>, 2014.
- [3] Android web site, “Platform Versions”. Available: [https://developer.android.com/about/dashboards/index.html?utm\\_source=ausdroid.net#Screens](https://developer.android.com/about/dashboards/index.html?utm_source=ausdroid.net#Screens)
- [4] H. C. Go, H. M. Yoo, "Everything analysis and porting of Android", HanBitMedia, 2011.
- [5] Greg Milette, Adam Stroud, “Professional Android Sensor Programming”, WROX Press, 2012.
- [6] Mahadev Satyanarayanan, “The Case for VM-Based Cloudlets in Mobile Computing”, Carnegie Mellon University, 2009.
- [7] Eric Y. Chen, Mistutaka Itoh, “Virtual Smartphone over IP”, NTT, 2010.



김호성(Ho-Sung Kim)

1995년 동아대학교 전자공학과(공학사)  
2008년 충남대학교 전자정보통신공학과(공학석사)  
2015년 ~ 현재 배재대학교 컴퓨터공학과(박사과정)  
1995년 ~ 현재 K-water 감사실 정보보안 차장  
※ 관심분야 : IT융합기술, ICT, Cloud Computing, Big data, Ubiquitous Computing , 수처리 알고리즘



서종균(Jong-Kyoun Seo)

1998년 건양대학교 정보통신공학과(공학사)  
2014년 충남대학교 산업시스템공학과(공학석사)  
2003년 ~ 현재 (주)유비테크 부설연구소 부장  
※ 관심분야 : 무선랜 보안, 차세대 이동통신(WiBro, LTE), 버티컬 핸드오버



박한수(Han-Su Park)

2011년 한밭대학교 컴퓨터공학과(공학사)  
2014년 한밭대학교 컴퓨터공학과(공학석사)  
2012년 ~ 현재 (주)유비테크 부설연구소 과장  
※ 관심분야 : Embedded, 차세대 이동통신(WiBro, LTE), 가상화



**정회경(Hoe-Kyung Jung)**

1985년 광운대학교 컴퓨터공학과(공학사)

1987년 광운대학교 컴퓨터공학과(공학석사)

1993년 광운대학교 컴퓨터공학과(공학박사)

1994년 ~ 현재 배재대학교 컴퓨터공학과 교수

※관심분야 : 멀티미디어 문서정보처리, XML, SVG, Web Services, Semantic Web, MPEG-21, Ubiquitous Computing, USN