

안드로이드 블루투스 통신을 이용한 도난방지 시스템

윤동환*, 김종민*, 하지완*, 공기석*, 방영철*, 이상우**

*한국산업기술대학교 컴퓨터공학과

**유니첼 주식회사

e-mail : {pariddoong^o, 1232wow, kresmar, kskong, ybang}@kpu.ac.kr

swlee@unichal.com

Anti-Theft System Using Android Bluetooth Communication

Dong-Hwan Yoon^{o*}, Jong-Min Kim^{*}, Ji-Wan Ha^{*}, Ki-Sok Kong^{*},

Young-Cheol Bang^{*}, Sang-Woo Lee^{**}

^{*}Department of Computer Engineering, Korea Polytechnic University

^{**}Unichal, Inc.

요 약

안드로이드 블루투스를 이용한 앱으로, 사용자가 도난방지 시스템을 사용하여 개인 소지품에 모듈을 부착하여 도난방지를 할 수 있는 시스템이다. 폰과 모듈간의 블루투스 수신 감도(RSSI)를 체크하여 수신 거리를 측정하게 되고, 그 거리가 10m이상 떨어지게되면 사용자에게 도난 알림을 알려준다. 또한, 사용자에게 전화가 온 것을 모듈 자체에서 수신하여 알려주는 핸드프리 기능도 갖췄다.

모듈은 ATMEGA128(MCU), FB155BC(블루투스 모듈), 부저, 진동등의 부품을 조립하여 제작하였다.

1. 서론

최근 스마트폰이 많은 인기를 끌면서 스마트폰 시장 점유율이 매년 점점 상승하고 있다[1]. 스마트폰의 운영체제 중 Apple사의 IOS와 Google사의 Android 운영체제가 대표적이며, 현재 Android OS를 탑재하는 스마트폰이 더 많은 상황이다[2].

스마트폰은 이전 세대 폰과 달리 중력센서, 감압센서 등 다양한 센서들이 부착되어 있고, 이런 센서를 이용하여 일상생활에서 편리하게 사용하거나 재미를 위한 앱을 만들 수 있다. 일반적으로 많은 이들이 외출 시에 꼭 필요한 물건을 깜박하는 일이 종종 있으며, 소중한 물건을 분실 혹은 도난당하는 경우가 발생한다. 이런 문제들을 도난 시에 알림을 통해서 알려줄 수 있게 되면, 이런 걱정을 덜어낼 수 있을 것이다.

2. 관련연구

2.1 기존 앱과 유사점 및 차이점

현재 안드로이드와 아이폰의 블루투스를 이용한 각종 앱들이 개발되고 있다. 그 중에서 본 논문의 앱과 유사한 기능을 지닌 앱들이 있다.

첫 번째 사례는 안드로이드용으로 부모와 아이가 서로 떨어지게 되면 미아를 방지하기 위해 개발된 앱으로 안드로이드 폰간의 도난방지를 위한 것이다. 두 번째 사례는 타인 사용자의 아이폰을 집을 경우에 잠금 및 경보를 작동 시켜 도난을 방지하는 기능이다. <표1>은 안드로이드

및 아이폰용 애플리케이션에 대한 사례이다.

<표1> 기존 시스템과 논문의 앱과 유사점

구분	안드로이드용 블루아이 지킴이 시스템	아이폰 진뚝개2 시스템
설명	부모와 아이가 일정거리 이상으로 떨어지면 알려주어 미아방지	주인이 아닌 사람이 가져갈 경우 잠금과 경보시스템 작동
대상	안드로이드폰간 도난방지	아이폰의 도난방지 가능
통신여부	일 대 일간의 블루투스 통신	통신 없이 자체적으로 도난방지
사례 이미지		

본 논문의 앱은 블루투스 칩모듈을 사용하여 핸드폰만이 아닌 개인소지품에도 부착이 가능토록 하여 도난이 방지하도록 개발하였다. 기존 블루투스 시스템의 경우 쌍방간의 안드로이드폰 또는 아이폰으로 도난방지, 미아방지, 또는 자체도난을 방지하지만, 본 논문의 앱은 모듈과 안드로이드폰 간의 블루투스 통신을 통해서 도난방지가 가능하다. <표2>는 기존의 시스템과 본 논문의 앱과의 차이점을 나타낸다.

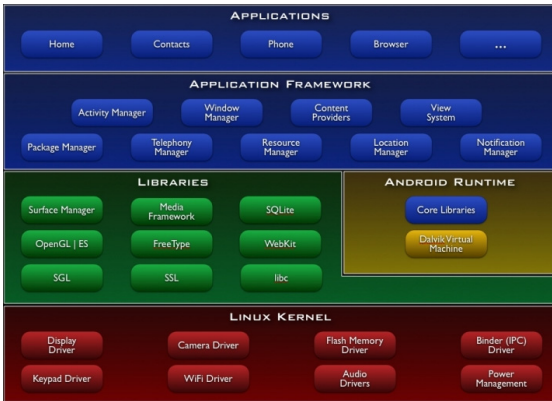
<표2> 기존 앱과 논문의 앱의 차이점

구분	기존의 시스템	개발될 시스템
가능	안드로이드폰간의 미아방지	안드로이드폰과 칩셋간의 통신으로 도난방지 가능
장치	쌍방간 안드로이드폰이 필요함	칩모듈만 부착하면 방지 가능
용도	안드로이드폰 도난방지	개인소지품 도난방지
비용	안드로이드폰 가격 X 2	안드로이드폰 가격 + 칩모듈 가격
다형성	안드로이드폰 상호간에만 가능	부착이 가능한 모든 제품에 가능

2.2.1 안드로이드 플랫폼

안드로이드는 운영체제와 미들웨어 그리고 핵심 애플리케이션을 포함하고 있는 모바일 디바이스를 위한 소프트웨어 Stack이다. 안드로이드 SDK는 Java 프로그래밍 언어를 사용하여 안드로이드 플랫폼 상의 애플리케이션을 개발하기 위한 도구들과 API를 제공한다[3][4].

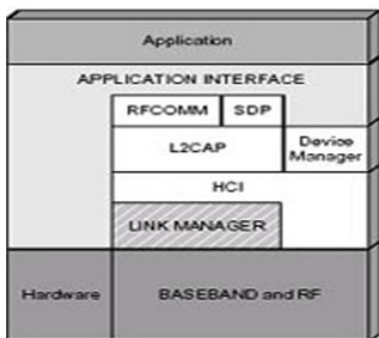
(그림 1)에서와 같이 애플리케이션 프레임워크가 존재하며 프레임워크 API를 조작할 수 있고, 개발에 필요한 라이브러리를 제공함으로 개발에 편의성이 있다.



(그림 1) 안드로이드 플랫폼

2.2.2 블루투스 프로파일

블루투스는 IEEE 802.15.1 규격을 사용하는 근거리 무선 통신이다. 블루투스는 다양한 기기들이 안전하고 저렴한 비용으로 전 세계적으로 이용할 수 있는 무선 주파수를 이용하여 서로 통신할 수 있게 한다. 본 논문의 앱은 블루투스 3.0버전을 사용하고 있다. 블루투스 3.0의 특징은 802.11 PAL(Protocol Adaptation Layer)를 채용하여 속도를 최대 24Mbps로 향상되었다. 블루투스 기기간의 대용량 파일의 송수신이 가능하고, 내장된 전력 관리기능으로 전력소모를 줄일 수 있다[5].



(그림 2) 블루투스 SPP 프로파일 스택구조

(그림 2)는 본 논문의 앱에서 사용하는 블루투스 SPP 프로파일 스택의 구조이다. 여러 가지의 블루투스 프로파일 중에 SPP를 사용하기 때문에 별도의 호스트는 존재하

지 않고, 호스트 컨트롤러인 블루투스 모듈만이 존재하고, 이 모듈 내에 프로토콜 스택이 구현되어있다[6].

2.3. 개발환경

<표3>은 본 논문의 앱 개발을 위한 개발환경이다

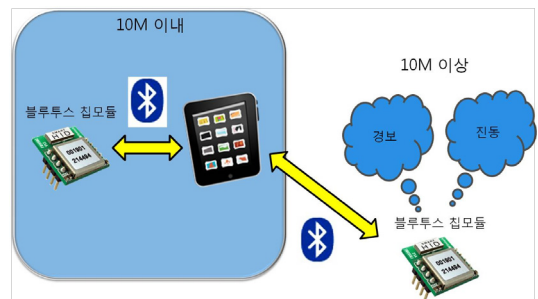
<표3> 본 논문의 앱 개발환경

기종		안드로이드폰(Samsung 갤럭시 U)
운영체제		안드로이드 프로요(2.2)
개발언어		안드로이드 2.2 SDK
개발 툴		Eclipse 3.5 버전
블루투스버전		블루투스 3.0
기종		블루투스 임베디드 모듈(FB155BC)
프로토콜		블루투스 SPP 프로파일
개발언어		C 언어
개발 툴		컴파일러-WinAVR / 에디터-AVRStudio
블루투스버전		블루투스 2.0
기종		노트북(Samsung SENS R430)
운영체제		Windows 7
블루투스버전		블루투스 3.0

3. 세부 설계 및 구현

3.1 시스템 구성

(그림 3)은 본 논문의 앱 시스템 구성도이다.



(그림 3) 도난방지 시스템 구성도

3.2 주요 구현 사항

3.2.1 안드로이드 어플리케이션 구현

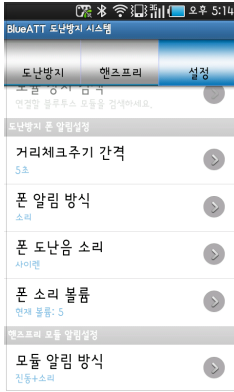
3.2.1.1 환경설정 액티비티(PreferenceActivity)

환경설정 액티비티에서는 안드로이드 폰의 블루투스의 전원관리와 도난방지 서비스, 핸드프리 서비스 등을 수행할 수 있다. 아래의 (그림 4)에서는 폰 도난 음 종류를 변경하게 되면 그것을 감지하여 환경설정창의 UI 정보 상태를 바꿔주고 도난방지 서비스 수행 시 값을 변경하게 되면 알림 다이얼로그를 띄우게 된다.

```
//공유 환경 설정 변화 감지 리스너
public void onSharedPreferenceChanged
(SharedPreferences sharedPreferences, String key) {

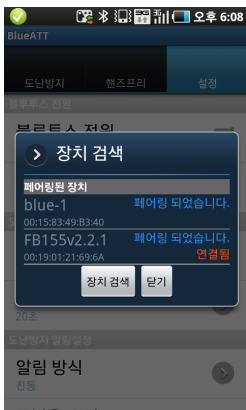
if(key.equals(PREFERENCE_ALERTSOUND)){
    alertKindValue = Integer.parseInt(sharedPreferences.getString(key, "0"));
    alert_pref.setSummary(alert_pref.getEntries()[alertKindValue]);
    if(check_anti_service_pref.isChecked())
        new ChangedValueAlertDlg(this);
}
}
```

(그림 4) 환경설정 액티비티 소스



(그림 5) 환경설정 액티비티 화면

3.2.1.2 장치 검색 다이얼로그



(그림 6) 도난방지 서비스

도난방지나 핸드프리를 수행하기 위해서는 장치 검색 다이얼로그에서 장치를 검색하고 연결해야한다. 이미 폰의 블루투스와 블루투스 모듈간의 페어링이 되어있는 경우 '페어링 된 장치'라는 리스트뷰에 나타나게 되는데 그 리스트중 하나의 항목을 선택할 경우 아래의 (그림 7)에서 선택한 장치와 블루투스 연결을 수행할 준비를 하게 된다.

```
pairedList.setOnItemClickListener(new OnItemClickListener() {
    public void onItemClick(AdapterView<?> arg0, View view,
        int index, long arg3) {
        intent.putExtra("STOP", 0);
        act.sendBroadcast(intent);
        //블루투스 검색 중지
        bluetooth.cancelDiscovery();
        device = pairedDeviceList.get(index);
        intent.putExtra("CONNECT", device);
        act.sendBroadcast(intent);
    }
});
```

(그림 7) 장치 검색 다이얼로그 소스

3.2.1.3 도난방지 서비스

(그림 8)에서는 도난방지 서비스가 수행되어 모듈과 블루투스 연결이 된 상태에서 모듈과 안드로이드 폰간의 거리가 10m이상 떨어지거나 사용자의 정상적인 종료가 아닌 비정상적인 종료가 되는 경우 도난 발생 액티비티를 띄우게 되고 모듈에서도 경보가 발생하게 된다.

```
public void run() {
    while (serviceQuitFlag == false) {
        // 블루투스 연결이 10m이상 떨어지거나 연결이 끊어질경우
        if ((module_conn_count==1 && module_disconn_flag==true) || rssi>=14) {
            // 경보액티비티 실행
            Intent i = new Intent(AntiThiefService.this,AlertActivity.class);
            i.putExtra("SEVICE_NOTIFYWAY", notifyWay);
            i.putExtra("SEVICE_ALERTKIND", alertKind);
            i.putExtra("SEVICE_SOUNDVOL", soundVolume);

            //activity가 아닌 서비스에서 startActivity 사용하기 위해서 PendingIntent를 사용
            //PendingIntent는 자동으로 FLAG_ACTIVITY_NEW_TASK를 통해 태스크를 생성한다.
            PendingIntent p = PendingIntent.getActivity
                (AntiThiefService.this, 0, i, PendingIntent.FLAG_UPDATE_CURRENT);

            if(state == STATE_CONNECTED){
                sendMessage("q"); //모듈 경보음을 울리게 한다.
            }
        }
    }
}
```

(그림 8) 도난방지 서비스 소스

3.2.1.4 도난방지 액티비티

(그림 9)에서는 도난방지 서비스에서 모듈과 폰간의 블루투스의 수신감도(RSSI)와 수신품질(LQI)를 체크하고 이를 통해 거리를 측정 한 후 도난방지 액티비티로 인텐트를 보내 그 값들을 도난방지 액티비티가 받아 띄워준다.

```
else if(intent.getAction().equals("kpu.BlueATT.action.READ")) {
    String readRssiText = intent.getStringExtra("READ_RSSI");
    String readLqiText = intent.getStringExtra("READ_LQI");
    int readCalc = intent.getIntExtra("READ_CAL_DISTANCE",-1);

    if(readRssiText!=null){
        setRssiText(readRssiText);
        setAntennaImag(RSSI_FLAG,Integer.parseInt(readRssiText));
    }
    else if(readLqiText!=null){
        setLqiText(readLqiText);
        setAntennaImag(LQI_FLAG,Integer.parseInt(readLqiText));
    }
    if(readCalc!=-1){
        setMeasureText(""+readCalc);
    }
}
```

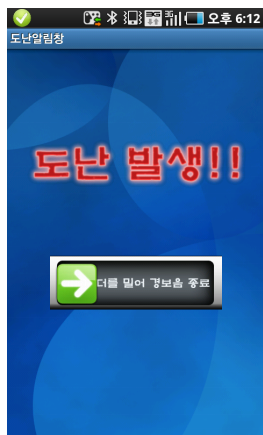
(그림 9) 도난방지 액티비티 소스

3.2.1.5 도난발생 액티비티

(그림 10)에서는 MediaPlayer 클래스를 이용하여 도난발생 음을 재생하고 정지하는 기능을 갖는 클래스이다. 도난방지서비스에서 안드로이드폰과 모듈간의 거리가 10m가 넘어가거나 비정상적인 연결종료가 발생할 경우에는 도난발생 액티비티가 수행된다.

```
class AlertSoundPlayer{
    MediaPlayer mediaPlayer;
    AlertSoundPlayer(Context context, int id){
        mediaPlayer = MediaPlayer.create(context,id);
    }
    void play(){
        mediaPlayer.seekTo(0);
        mediaPlayer.setLooping(true);
        mediaPlayer.setVolume(soundVolumeValue*0.1f,
            soundVolumeValue*0.1f);
        mediaPlayer.start();
    }
    void stop(){
        mediaPlayer.stop();
        mediaPlayer.reset();
    }
}
```

(그림 10) 도난발생 액티비티 소스



(그림 11) 도난발생 액티비티 화면

3.2.1.6 핸즈프리 서비스

(그림 12)에서는 PhoneStateListener를 통해 전화 수신 신호를 감지하여 전화가 걸려오게되면 모듈에서 진동과 비프를 울려 사용자에게 알려주도록 한다.

```
PhoneStateListener callListener = new PhoneStateListener() {
    public void onCallStateChanged(int state, String incomingNumber) {
        switch (state) {
            case TelephonyManager.CALL_STATE_RINGING:
                try {
                    Thread.sleep(500);
                } catch (InterruptedException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
                sendMessage("S");
                break;
        }
    }
}
```

(그림 12) 핸즈프리 서비스 소스

3.2.2 블루투스 모듈 구현

ATMEGA128과 FB155BC 블루투스 모듈, 부저, 진동 장치 등을 부착한 보드 모듈을 제작하여 안드로이드 폰과

의 연결을 통하여 도난방지와 핸드프리 기능을 수행하게 하였다. (그림 13)은 ATMEGA128상에서 작동하는 AVR 코드로 FB155BC에 AT-COMMAND를 보내어 블루투스 모듈을 초기화한다. 초기화후에 블루투스 모듈을 Slave로 세팅하고 블루투스 장치 검색대기를 수행하여 안드로이드 폰에서 블루투스 연결이 요청될 때까지 대기한다.

```
// UART initialize, Baud Rate Set
UART0_initialize(9600);
// Global interrupt enable
sei();

//Module initialize
TX0_string("ATZ#r");
_delay_ms(1000);

//Slave mode setting
TX0_string("AT+BTROLE=S#r");
_delay_ms(1000);

TX0_string("AT+BTSCAN#r");
_delay_ms(1000);
```

(그림 13) 블루투스 모듈 소스

4. 결론 및 향후 연구과제

이 시스템을 이용하여 꼭 필요한 물건을 체크 하거나 물건을 두고 오거나 소매치기 등 분실 시에 알람을 통해 물건과의 거리 및 방향의 대략 적인 거리와 위치를 확인할 수 있다. 또한 어린이의 목걸이나 가방 등에 부착하여 미아 방지 등 만들어진 모듈을 부착함으로써 다양한 용도로 사용 할 수 있을 것이다. 또한 스마트폰을 백팩이나, 핸드백에 넣어 두고 다니는 경우 스마트폰에 벨이 울릴 시 이를 칩에서 알람을 통해 확인할 수 있는 부가 기능도 구현 해두었다. 스마트폰에서 설정을 통해 알람 음이나, 크기를 변경하는 옵션도 설정할 수 있다.

현재 모듈의 경우 정밀한 PCB판이 아닌 만능 기판에 부품을 구성하였다. 이는 크기가 조금 크기에 부착할 수 있는 물건에 약간의 제한이 생긴다. 추후 모듈을 다양한 곳에 부착할 수 있도록, 모듈의 소형화가 필요하다. 또한 폰에서 그래픽적인 부분을 좀 더 깔끔하게 표현하고, 사용자 인터페이스를 좀 더 개선해야할 필요가 있다.

참고문헌

- [1] 스마트폰 시장 증가율 “<http://samsungamoled.net/259>”.
- [2] OS별 사용 현황 “<http://pch1286.blog.me/80131080987>”.
- [3] 안드로이드 소개 “<http://developer.android.com>”.
- [4] 안드로이드 게시판 “<http://www.androidside.com>”.
- [5] 블루투스 소개 “<http://ko.wikipedia.org>”.
- [6] 블루투스 스택 “<http://terapi.tistory.com/entry/bluetooth-profile>”.