
# 《手机平台应用开发》

# 传感器及地图相关应用

**学 院 名 称** ： 数据科学与计算机学院

**成　　员** ： 陈伟宸

**时　　间** ： 2016 年 12 月 3 日

## 一、实验环境

操作系统：Windows 10

IDE：Android Studio 2.2.2

## 二、实验过程

1. 新建 CheckPermissionActivity.java，检查并申请权限

```
public class CheckPermissionActivity extends AppCompatActivity {
    private Handler mHandler = new Handler();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        RxPermissions rxPermissions = new RxPermissions(this);
        rxPermissions
                .request(Manifest.permission.ACCESS_FINE_LOCATION,
                        Manifest.permission.WRITE_EXTERNAL_STORAGE)
                .subscribe((Action1) (aBoolean) -> {
                    if (aBoolean) {
                        Toast.makeText(CheckPermissionActivity.this,
                                "Granted", Toast.LENGTH_SHORT).show();
                        Intent intent = new Intent(CheckPermissionActivity.this, MainActivity.class);
                        startActivity(intent);
                        finish();
                    }
                    else {
                        Toast.makeText(CheckPermissionActivity.this,
                                "App will finish in 3 seconds", Toast.LENGTH_SHORT).show();
                        mHandler.postDelayed(() -> {
                                finish();
                            }, 3000);
                    }
                });
    }
}
```

在此之前需要在 build.gradle 中添加相应的依赖

```
compile 'com.tbruyelle.rxpermissions:rxpermissions:0.9.0@aar'
compile 'io.reactivex:rxandroid:1.2.1'
compile 'io.reactivex:rxjava:1.1.6'
```

2. 按照实验文档申请百度地图 API 等

3. 设计地图页面的样式

```xml
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@mipmap/center_on" android:state_checked="true"/>
    <item android:drawable="@mipmap/center_off"/>
</selector>
```

```xml
<com.baidu.mapapi.map.TextureMapView
    android:id="@+id/bmapView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:clickable="true" />

<ToggleButton
    android:id="@+id/tb_center"
    android:layout_width="48dp"
    android:layout_height="48dp"
    android:layout_alignParentBottom="true"
    android:layout_alignParentStart="true"
    android:layout_marginBottom="20dp"
    android:layout_marginStart="20dp"
    android:background="@drawable/toggle_center"
    android:checked="true"
    android:textOff=""
    android:textOn=""/>
```

4. 在 MainActivity.java 中实现地图的功能

（1）首先获取到各控件和传感器

```
mMapView = (TextureMapView) findViewById(R.id.bmapView);
mSensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
mMagneticSensor = mSensorManager.getDefaultSensor(Sensor.TYPE_MAGNETIC_FIELD);
mAccelerometerSensor = mSensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
mLocationManager = (LocationManager) getSystemService(LOCATION_SERVICE);
mToggleButton = (ToggleButton) findViewById(R.id.tb_center);
mConverter = new CoordinateConverter();
```

（2）选择一个较好的 provider

```
Criteria criteria = new Criteria();
criteria.setAccuracy(Criteria.ACCURACY_FINE);
criteria.setAltitudeRequired(false);
criteria.setBearingRequired(false);
criteria.setCostAllowed(true);

provider = mLocationManager.getBestProvider(criteria, true);
```

（3）获得最后得到的位置

```
if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_COARSE_LOCATION)
                != PackageManager.PERMISSION_GRANTED) {
    //...
    return;
}
mCurrentLocation = mLocationManager.getLastKnownLocation(provider);
```

（4）将获取到的位置转换为百度地图的坐标系并在地图上用箭头显示出来

```
Bitmap bitmap = Bitmap.createScaledBitmap(BitmapFactory.decodeResource(getResources(),
        R.mipmap.pointer), 100, 100, true);
BitmapDescriptor bitmapDescriptor = BitmapDescriptorFactory.fromBitmap(bitmap);

mMapView.getMap().setMyLocationEnabled(true);
MyLocationConfiguration configuration = new MyLocationConfiguration(
        MyLocationConfiguration.LocationMode.NORMAL, true, bitmapDescriptor);
mMapView.getMap().setMyLocationConfigeration(configuration);

if (mCurrentLocation != null) {
    mConverter.from(CoordinateConverter.CoordType.GPS);
    mConverter.coord(new LatLng(mCurrentLocation.getLatitude(), mCurrentLocation.getLongitude()));
    LatLng desLatLng = mConverter.convert();
    MyLocationData.Builder data = new MyLocationData.Builder();
    data.latitude(desLatLng.latitude);
    data.longitude(desLatLng.longitude);
    data.direction(mCurrentDirection);
    mMapView.getMap().setMyLocationData(data.build());
    MapStatus mMapStatus = new MapStatus.Builder().target(desLatLng).build();
    MapStatusUpdate mMapStatusUpdate = MapStatusUpdateFactory.newMapStatus(mMapStatus);
    mMapView.getMap().setMapStatus(mMapStatusUpdate);
}
```

（5）实现地图拖动

```
mMapView.getMap().setOnMapTouchListener(new BaiduMap.OnMapTouchListener() {
    @Override
    public void onTouch(MotionEvent motionEvent) {
        switch (motionEvent.getAction()) {
            case MotionEvent.ACTION_MOVE:
                mToggleButton.setChecked(false);
                break;
            default:
                break;
        }
    }
});
```

## （6）实现点击 ToggleButton 回到当前位置

```
mToggleButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (mToggleButton.isChecked()) {
            mConverter.from(CoordinateConverter.CoordType.GPS);
            mConverter.coord(new LatLng(mCurrentLocation.getLatitude(),
                    mCurrentLocation.getLongitude()));
            LatLng desLatLng = mConverter.convert();
            MapStatus mMapStatus = new MapStatus.Builder().target(desLatLng).build();
            MapStatusUpdate mMapStatusUpdate = MapStatusUpdateFactory.newMapStatus(mMapStatus);
            mMapView.getMap().setMapStatus(mMapStatusUpdate);
        }
    }
});
```

## （7）在 onResume 函数中注册传感器监听器

```
@Override
protected void onResume() {
    super.onResume();
    //在activity执行onResume时执行mMapView. onResume (), 实现地图生命周期管理
    mMapView.onResume();

    mSensorManager.registerListener(mSensorEventListener, mMagneticSensor,
            SensorManager.SENSOR_DELAY_GAME);
    mSensorManager.registerListener(mSensorEventListener, mAccelerometerSensor,
            SensorManager.SENSOR_DELAY_GAME);
    if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION)
            != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        //...
        return;
    }
}
```

（8）在 onPause 函数中注销传感器监听器

```
@Override
protected void onPause() {
    super.onPause();
    //在activity执行onPause时执行mMapView. onPause (),实现地图生命周期管理
    mMapView.onPause();

    mSensorManager.unregisterListener(mSensorEventListener);
    if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION)
            != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        //...
        return;
    }
    mLocationManager.removeUpdates(mLocationListener);
}
```

（9）实现 LocationListener，在位置发生变化时，更新地图

```
LocationListener mLocationListener = new LocationListener() {
    @Override
    public void onLocationChanged(Location location) {
        mCurrentLocation = location;
        mConverter.from(CoordinateConverter.CoordType.GPS);
        mConverter.coord(new LatLng(mCurrentLocation.getLatitude(),
                mCurrentLocation.getLongitude()));
        LatLng desLatLng = mConverter.convert();
        MyLocationData.Builder data = new MyLocationData.Builder();
        data.latitude(desLatLng.latitude);
        data.longitude(desLatLng.longitude);
        data.direction(mCurrentDirection);
        mMapView.getMap().setMyLocationData(data.build());
        MapStatus mMapStatus = new MapStatus.Builder().target(desLatLng).build();
        MapStatusUpdate mMapStatusUpdate = MapStatusUpdateFactory.newMapStatus(mMapStatus);
        mMapView.getMap().setMapStatus(mMapStatusUpdate);
        mToggleButton.setChecked(true);
    }
```

（10）实现 SensorEventListener，在手机方向转动时箭头跟着转动，在摇晃手机时，弹出 BUG 反馈对话框

```java
@Override
public void onSensorChanged(SensorEvent sensorEvent) {
    switch (sensorEvent.sensor.getType()) {
        case Sensor.TYPE_ACCELEROMETER:
            accValues = sensorEvent.values;
            float x = accValues[0];
            float y = accValues[1];
            float z = accValues[2];
            int medumValue = 19;
            if (Math.abs(x) > medumValue || Math.abs(y) > medumValue || Math.abs(z) > medumValue) {
                Message message = new Message();
                message.what = SENSOR_SHAKE;
                mHandler.sendMessage(message);
            }
            break;
        case Sensor.TYPE_MAGNETIC_FIELD:
            magValues = sensorEvent.values;
            break;
        default:
            break;
    }
```

```java
float[] R = new float[9];
float[] values = new float[3];

if (accValues != null && magValues != null) {
    SensorManager.getRotationMatrix(R, null, accValues, magValues);
    SensorManager.getOrientation(R, values);

    mConverter.from(CoordinateConverter.CoordType.GPS);
    mConverter.coord(new LatLng(mCurrentLocation.getLatitude(), mCurrentLocation.getLongitude()));
    LatLng desLatLng = mConverter.convert();
    MyLocationData.Builder data = new MyLocationData.Builder();
    data.latitude(desLatLng.latitude);
    data.longitude(desLatLng.longitude);
    float direction = (float) Math.toDegrees(values[0]) + 180;
    data.direction(direction);
    mMapView.getMap().setMyLocationData(data.build());

    mCurrentDirection = direction;
}
```

（11）在检测到手机晃动时，使用一个 Handler 来弹出对话框

```
Handler mHandler = handleMessage(msg) -> {
        super.handleMessage(msg);
        switch (msg.what) {
            case SENSOR_SHAKE:
                if (!SHAKING) {
                    SHAKING = true;
                    AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
                    builder.setTitle("BUG反馈")
                            .setMessage("发送邮件给开发者")
                            .setPositiveButton("是", new DialogInterface.OnClickListener() {
                                @Override
                                public void onClick(DialogInterface dialogInterface, int i) {
                                    Intent data = new Intent(Intent.ACTION_SENDTO);
                                    data.setData(Uri.parse("mailto:530415489@qq.com"));
                                    data.putExtra(Intent.EXTRA_SUBJECT, "MapSensor BUG反馈");
                                    data.putExtra(Intent.EXTRA_TEXT, "我在使用 MapSensor 的时候遇到了 BUG");
                                    startActivity(Intent.createChooser(data, "发送邮件"));
                                    SHAKING = false;
                                }
                            })
                            .setNegativeButton("否", new DialogInterface.OnClickListener() {
                                @Override
                                public void onClick(DialogInterface dialogInterface, int i) {
                                    SHAKING = false;
                                }
                            }).create().show();
                }
                break;
        }
};
```

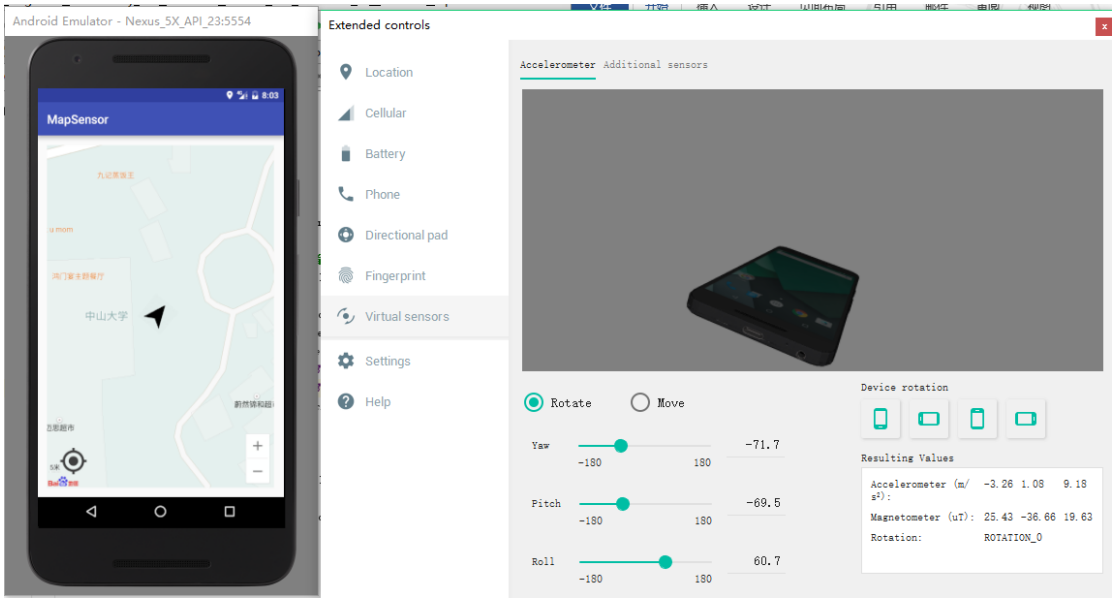
## 三、实验结果

### 1. 运行程序


2. 拖动地图


3. 点击 ToggleButton

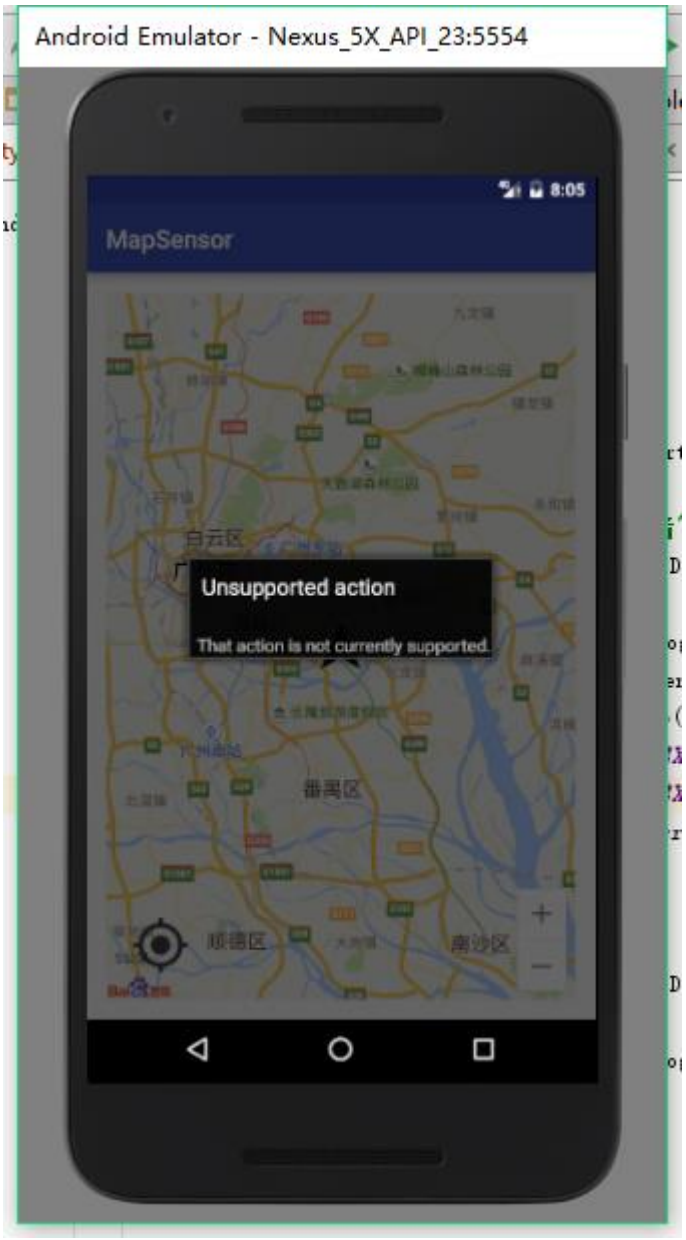

4. 改变手机朝向


Android Emulator - Nexus_5X_API_23:5554
MapSensor
中山大学
Extended controls
Location
Cellular
Battery
Phone
Directional pad
Fingerprint
Virtual sensors
Settings
Help
Accelerometer
Additional sensors
Rotate
Move
Yaw
-71.7
-180
180
Pitch
-69.5
-180
180
Roll
60.7
-180
180
Device rotation
Resulting Values
Accelerometer (m/s²): -3.26 1.08 9.18
Magnetometer (uT): 25.43 -36.66 19.63
Rotation: ROTATION_0

5. 晃动手机


Android Emulator - Nexus_5X_API_23:5554
8:04
MapSensor
BUG反馈
发送邮件给开发者
否
是
番禺区
顺德区
南沙区

6. 点击“是”




由于在该模拟器上没有绑定 gmail 账号，所以暂时不能成功发送邮件

## 四、实验心得

1. 使用 com.baidu.mapapi.map.MapView 会出错，应该使用 com.baidu.mapapi.map.TextureMapView

2. 在低版本的 API ( API 19 ) 中，无法使用百度地图的定位功能，高

版本（API 23）不存在此问题

3. 第一次调用 getLastKnownLocation 时会返回 null，需要使用模拟器传递一个 Location 给应用程序

4. 传感器直接获取到的方向与箭头指向相反，应该将该角度加上 180 度

5. 在实现“摇一摇”时，传感器会连续监测到许多满足条件的事件，因此，应该设置一个标志位，在第一次处理该事件时将标志位置位，表示已经有线程在处理“摇一摇”事件，并在处理结束后把标志位复位