

《手机平台应用开发 (与 Google 共建) 实验》

实验十：传感器及地图相关应用 实验报告

学 院 名 称 : 数据科学与计算机学院

专 业 : 软件工程 (计应)

学 生 姓 名 : 张凯鑫

学 号 : 14331362

班 级 : 周三上午 4-5 节、周五下午 7-8 节

【实验目的】

1. 学会使用 android 手机不同的传感器，包括磁强计、加速度传感器等。
2. 学会使用 android 位置服务获得经纬度信息。
3. 学会使用百度地图基础 API 进行地图应用的开发。

【实验内容】

1. 实验效果如图 1 所示：

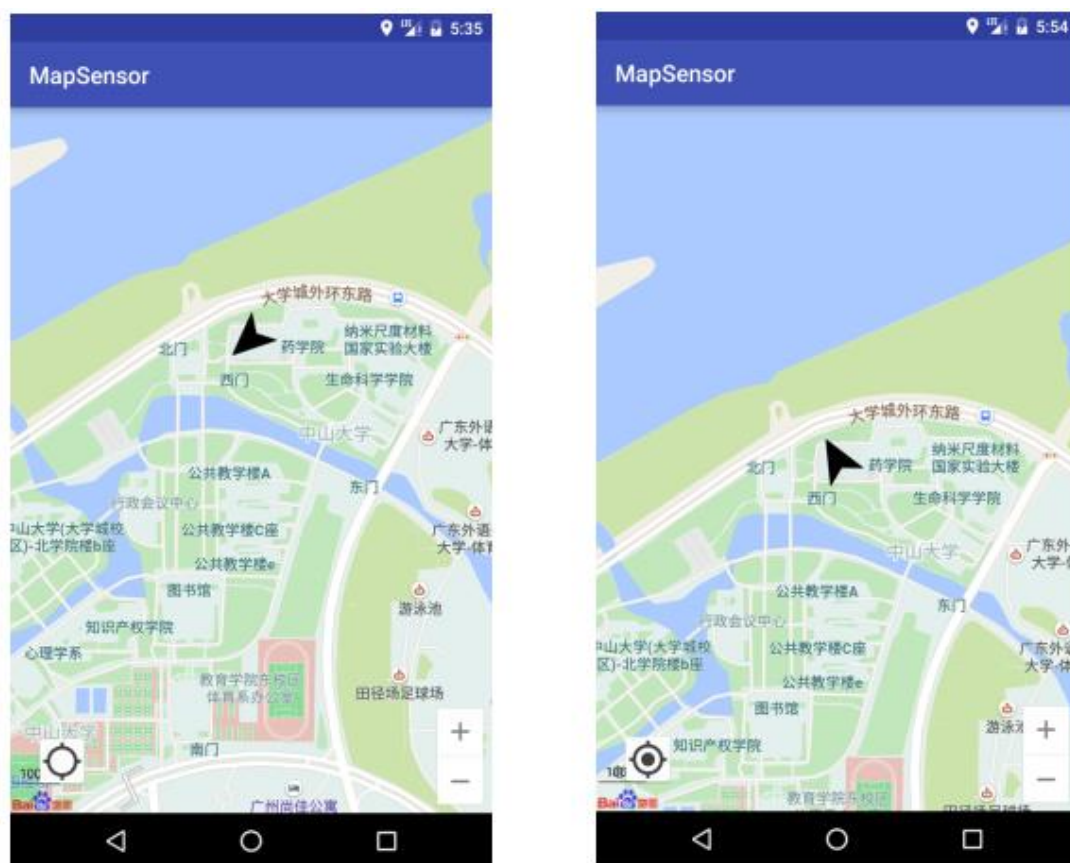







图 1 实验效果图

2. 主要分为以下三部分

- a) 调用百度地图 SDK 实现地图的展示以及交互，其中包括左下角的按钮要求实现的功能如下：

- i. 当按钮处于  状态时：如果 GPS 位置有更新，那么更新用户当前所在位置，并使得当前位置的箭头图标一直保持在屏幕中心。
- ii. 按钮处于  状态时，更新当前位置的箭头的时候不移动地图。
- iii. 点击按钮可以切换这两种状态，当按钮进行  状态的时候，使得表示当前位置的箭头位于屏幕中央。
- iv. 如果按钮处理  的状态手动平移地图的时候，地图进行平移并且按钮变为  状态。

- b) 获取当前所在位置与方向，并在百度地图上显示出来，如图中黑色箭头所示，箭头中心位置为定位结果，箭头方向为手机的朝向。
- c) 实现简单摇一摇功能，当晃动手机的时候作出相应的响应，其中，摇一摇之后会发生什么事情，可发挥创造力自己实现（实验中实现为：摇一摇后显示当前位置详细的经纬度信息）。

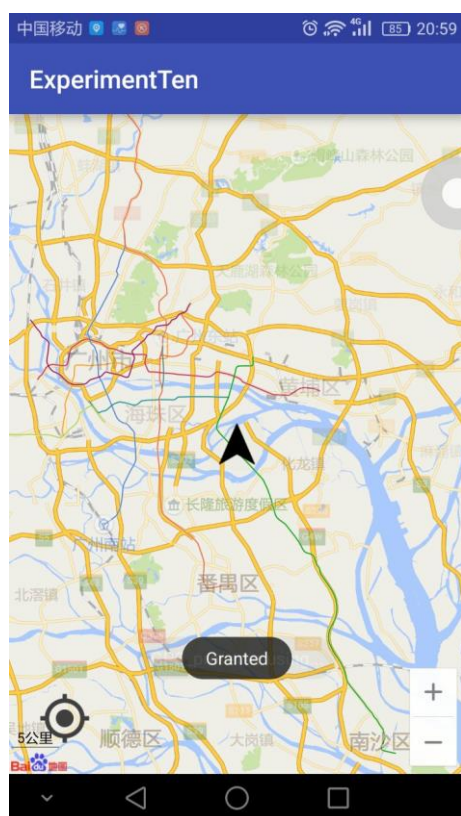
【实验过程】

（以下过程为简单步骤介绍，具体实验效果与关键代码在“实验结果”中结合截图分析给出）

1. 创建新的 Android Studio 项目，命名为：ExperimentTen。
2. 申请百度的开发者密钥：获得与 app 包名相同的 AK。
3. 下载百度地图 SDK 的依赖库，按照“百度地图 SDK 配置环境与发布”，把依赖库放到相应的文件夹并添加到工程中。
4. 按要求创建布局文件：activity_main.xml，添加控件 MapView,ToggleButton。在 AndroidManifest.xml 文件里添加权限以及应用 AK，实现百度地图 API 的基础显示。
5. 实现 MainActivity.java，主要是传感器的注册、监听、注销以及百度地图的 API 的简单开发等。依次可为：使用 LocationManager 通过 GPS、NETWORK 等方式获取用户的经纬度信息、使用 SensorManager 来通过监听地磁传感器、加速度传感器的数据变化来获取手机的方向信息、在百度地图中添加当前所在位置的箭头，并设置箭头的方向和位置、实现 ToggleButton 开关的功能、通过加速度传感器的数据变化来实现摇一摇的功能。
6. 不断运行并调整实验代码直到实验完成。

【实验结果】

实验效果图及部分代码分析如下（实验代码详见 lab10_code 文件夹）：



验证权限是否开启



显示使用的是 GPS 还是 NETWORK 定位：GPS

【关键代码】

使用 RxAndroid, 验证权限是否开启:

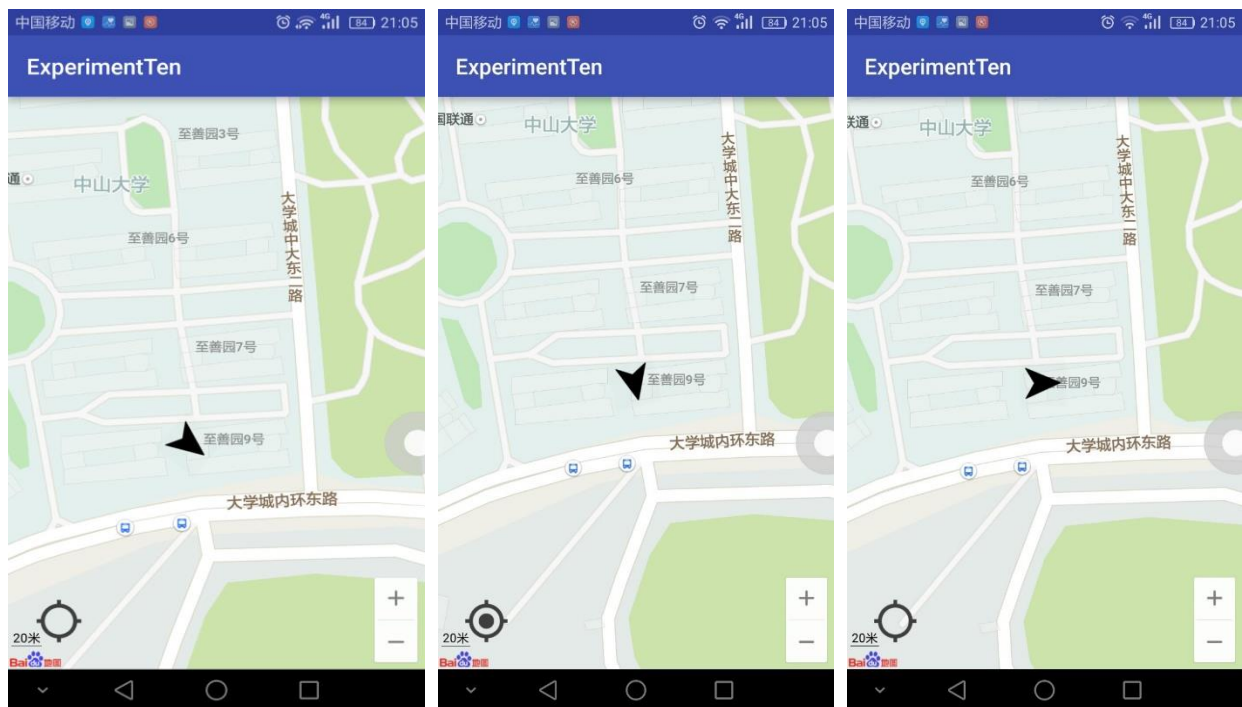
```
RxPermissions rxPermissions = new RxPermissions(this);
rxPermissions
    .request(Manifest.permission.ACCESS_FINE_LOCATION,
            Manifest.permission.ACCESS_COARSE_LOCATION,
    .subscribe(new Action1<Boolean>() {
        @Override
        public void call(Boolean granted) {
            if (granted) {
                Toast.makeText(MainActivity.this, "Granted",
Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(MainActivity.this, "App will finish in 3 secs...",
Toast.LENGTH_SHORT).show();
                finish();
            }
        }
    });
```

获取当前经纬度信息:

```
locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
locationProvider = LocationManager.GPS_PROVIDER;
try {
    location = locationManager.getLastKnownLocation(locationProvider);
    locationManager.requestLocationUpdates(locationProvider, 3000, 1, locationListener);
} catch (SecurityException e) {
    e.printStackTrace();
}
```

将经纬度信息转换为百度地图坐标, 并在百度地图中定位箭头位置:

```
if (location != null) {
    MyLocationData.Builder builder = new MyLocationData.Builder();
    CoordinateConverter coordinateConverter = new CoordinateConverter();
    coordinateConverter.from(CoordinateConverter.CoordType.GPS);
    coordinateConverter.coord(new LatLng(location.getLatitude(), location.getLongitude()));
    LatLng desLatLng = coordinateConverter.convert();
    builder.latitude(desLatLng.latitude);
    builder.longitude(desLatLng.longitude);
    builder.direction(rotation);
    mMapView.getMap().setMyLocationData(builder.build());
    MapStatus mapStatus = new MapStatus.Builder().target(desLatLng).build();
    MapStatusUpdate mapStatusUpdate = MapStatusUpdateFactory.newMapStatus(mapStatus);
    mMapView.getMap().setMapStatus(mapStatusUpdate);
}
```



放大地图，左下角按钮自动切换； 点击左下角按钮，地图定位； 箭头朝向随手机手机朝向改变

【关键代码】：监听器部分

按钮自动切换的实现是在百度地图显示控件添加监听器：

```
mMapView.getMap().setOnMapTouchListener
```

箭头位置的改变，是设置 location 监听器，使位置改变时，箭头随之改变：

```
private LocationListener locationListener = new LocationListener() {
    @Override
    public void onLocationChanged(Location location) {
        MyLocationData.Builder builder = new MyLocationData.Builder();
        CoordinateConverter coordinateConverter = new CoordinateConverter();
        coordinateConverter.from(CoordinateConverter.CoordType.GPS);
        coordinateConverter.coord(new
LatLng(location.getLatitude(), location.getLongitude()));
        LatLng desLatLng = coordinateConverter.convert();
        builder.latitude(desLatLng.latitude);
        builder.longitude(desLatLng.longitude);
        builder.direction(rotation);
        mMapView.getMap().setMyLocationData(builder.build());

        if (toggleButton.isChecked()) {
            MapStatus mapStatus = new MapStatus.Builder().target(desLatLng).build();
            MapStatusUpdate mapStatusUpdate =
MapStatusUpdateFactory.newMapStatus(mapStatus);
            mMapView.getMap().setMapStatus(mapStatusUpdate);
        }
    }
}
```


箭头方向的改变，是设置传感器监听器，使手机的朝向改变时，箭头朝向也改变，它是通过地磁传感器和加速度传感器共同计算出来的：

```
private SensorEventListener mSensorEventListener = new SensorEventListener() {
    float[] accValues = null;
    float[] magValues = null;

    @Override
    public void onAccuracyChanged(Sensor sensor, int accuracy) {}

    @Override
    public void onSensorChanged(SensorEvent event) {
        switch (event.sensor.getType()) {
            case Sensor.TYPE_ACCELEROMETER:
                isShark();
                break;
            case Sensor.TYPE_MAGNETIC_FIELD:
                getRotation();
                break;
            default:
                break;
        }
    }
};
```

传感器的注册是在 onResume() 中：

```
mSensorManager.registerListener(mSensorEventListener,
    mSensorManager.getDefaultSensor(Sensor.TYPE_MAGNETIC_FIELD),
    SensorManager.SENSOR_DELAY_GAME);
mSensorManager.registerListener(mSensorEventListener,
    mSensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER),
    SensorManager.SENSOR_DELAY_GAME);
```

传感器的注销是在 onPause() 中：

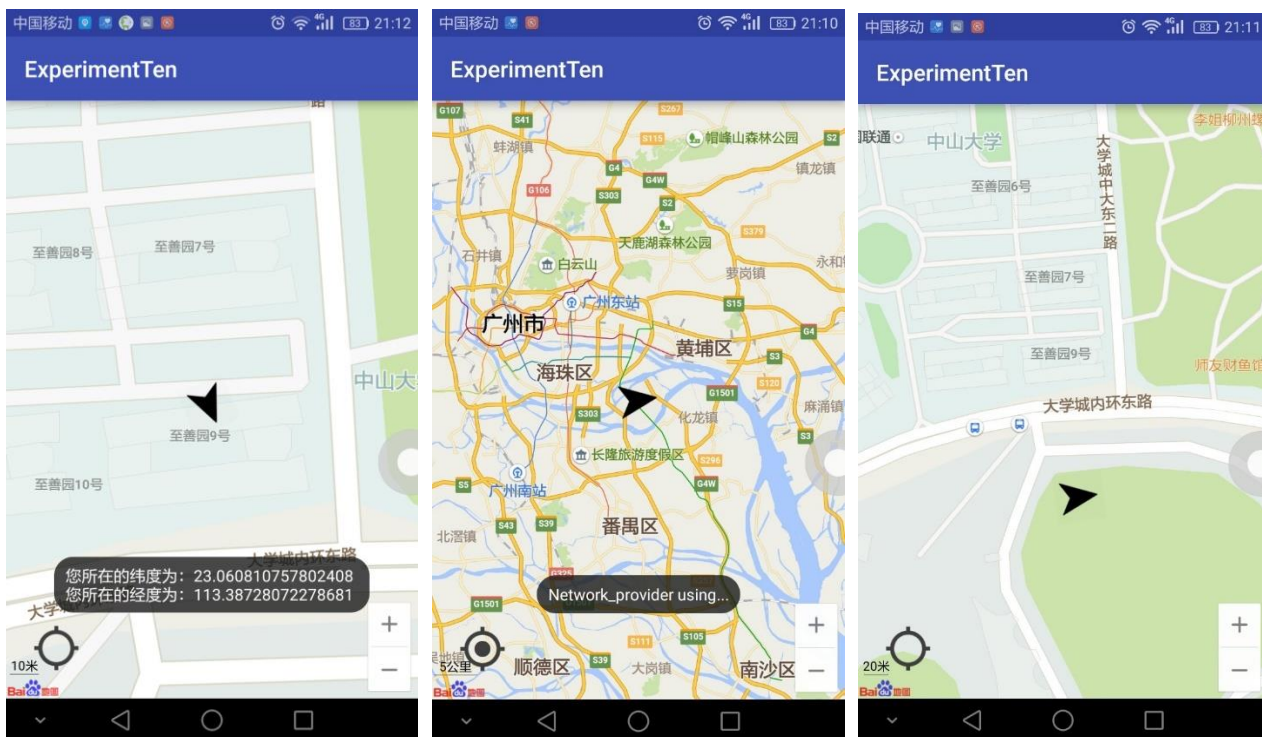
```
mSensorManager.unregisterListener(mSensorEventListener);
```

位置 LocationManager 的监听在 onCreate() 中使用：

```
locationManager.requestLocationUpdates(locationProvider, 3000, 1, locationListener);
```

位置 LocationManager 须在 onDestroy() 中取消监听：

```
if (locationManager != null) {
    try {
        locationManager.removeUpdates(locationListener);
    } catch (SecurityException e) {
        e.printStackTrace();
    }
}
```



摇一摇显示当前位置的经纬度信息； 无 GPS 时，使用 NETWORK 定位；NETWORK 定位显示大致位置

【关键代码】

摇一摇：可在加速度传感器中比较 X、Y、Z 轴方向的变量是否大于 18（一般非主观意识地摇手机时，变量小于 18）

```
if (accValues != null) {
    for (int i = 0; i < 3; i++) {
        if (Math.abs(accValues[i] - event.values[i]) > 18) {
            if (location != null) {
                Toast.makeText(MainActivity.this, "您所在的纬度为: " +
location.getLatitude() + "\n" +
                    "您所在的经度为: " + location.getLongitude(),
Toast.LENGTH_SHORT).show();
            }
            vibrator.vibrate(500);
            break;
        }
    }
}
accValues = event.values.clone();
```

【遇到的问题与解决方法】

1. 申请百度地图密钥时，按文档所说在“百度地图开放平台”中注册账号，但手机一直不能收到验证码，而且每天一个手机号码获得验证码是有次数限制的，注册不了账号，使得工程一直不能继续下去，后来，在“百度开发者中心”中注册了“百度开发者”账号，居然也成为了“百度地图开放平台开发者”，成功申请到密钥，工程得以继续。

2. 使用百度地图一打开就出现错误闪退，以为是导入包过程出错，反复检查发现并没错，后来发现报错是 `IllegalArgumentException`，在网上找到解决方法：将控件 `MapView` 改为 `TextureMapView`，百度地图成功被添加显示出来。
3. GPS 定位获得经纬度信息时，最初因为是在宿舍内，难以定位，误以为代码有问题，后来经过反复查看和网上求解，在宿舍阳台成功获得经纬度信息，但所得的定位与实际不符，细读文档发现是 `LocationManager` 返回的 `Location` 中的坐标系与百度地图的坐标系不一样，需要进行坐标系的转换。通过查看文档得以解决问题。
4. NETWORK 定位时，由于原先不知道测试时使用的手机默认只开启 GPS 定位，耗费了太多无用功，后来在手机的“设置——定位服务”中开启了所有的定位服务：GPS、WLAN 和移动网络。

【实验心得与体会】

1. 实验文档看着内容很多，最初很难看懂，又加上需要注册账号申请密钥等操作，给人一种很难很麻烦的感觉，但认真梳理思路，发现其实很有头绪，很容易实现。
2. 位置信息的获取是通过 `LocationManager` 来实现，首先确定使用的是何种定位，GPS 还是 NETWORK；接着，使用 `getLastKnownLocation()` 来获取 `location`，含经纬度，给 `locationManager` 设置监听：`requestLocationUpdates(locationProvider, 3000, 1, locationListener)`，定时获取位置信息；最后将 `location` 的坐标系转换为百度地图的坐标系，在百度地图上定位。
3. 传感器信息的获取是通过 `SensorManager`，需要注册监听：`registerListener()`，不使用时注销监听：`unregisterListener()`。
4. 通过本次实验，我对手机上不同的传感器，如磁强计、加速度传感器等有了一定的了解，基本掌握了它们的使用方法，同时也掌握了 android 位置服务获得经纬度信息的方法。并且学会了使用百度地图基础 API 进行地图应用的开发。