

手机平台应用开发实验报告

学号: 14331098

班级: 周五下午班

姓名: 黄建武

实验名称: Lab10

一、参考资料

1. [Android官方开发者文档](#)
2. [百度地图Android SDK API](#)
3. [摇一摇实现](#)

二、实验环境

- 系统: Windows 10 64位
- 工具: Android Studio 2.1.3
- JDK版本: 1.8
- minSdk: 19
- targetSdk: 24

三、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。

四、实验步骤

1. 阅读实验要求文档以及对应的课件PPT。
2. 运行Demo程序, 了解实验要求细节。
3. 申请百度的开发者密钥, 添加百度地图SDK的依赖库, 安装运行百度地图demo。
4. 添加应用所需权限, 开启位置服务, 添加注册传感器事件监听器。
5. 初始化百度地图, 设置当前位置图标, 开启定位图层, 因为编程时基本在室内, 故使用NETWORK_PROVIDER进行定位, 实现默认定位在当前所在位置。此处需要将网络定位获取到的Location的原始坐标转化为百度地图的坐标。
6. 添加百度地图触摸事件以及图片按钮点击事件, 实现当前位置在地图中心与否两种效果, 此处没有使用ToggleButton, 而是使用了ImageButton根据点击的次数显示不同的图片, 可以为ImageButton设置白色的背景色。
7. 获取当前所在位置与方向, 并在地图上显示出来。当传感器发生变化时, 获取地磁传感器和加速度传感器的数值, 计算出旋转矩阵并转化为角度, 判断前后两次的旋转角度差是否大于0.5, 是则更新箭头的位置和方向, 此外还要判断当前是否处于箭头在中心的模式, 是的话就将当前位置设置在地图中心。
8. 实现摇一摇功能, 判断速度是否大于某个阈值, 如果是就切换地图的类型, 普通话地图或卫星地图, 同时手机会振动。不使用加速度是因为测试时发现加速度很不稳定而且波动特别大, 很不准确, 为防止多次触发摇一摇, 两次摇一摇之间间隔至少是两秒。
9. 进行最后的调试以及完善。

主要代码部分如下:

```
// 设置地图图标
Bitmap bitmap = Bitmap.createScaledBitmap(BitmapFactory.decodeResource(
    getResources(), R.mipmap.pointer),
    100, 100, true);
BitmapDescriptor bitmapD = BitmapDescriptorFactory.fromBitmap(bitmap);
```

```

// 开启定位图层
baiduMap.setMyLocationEnabled(true);
MyLocationConfiguration config = new MyLocationConfiguration(
    MyLocationConfiguration.LocationMode.NORMAL, true, bitmapD);
baiduMap.setMyLocationConfiguration(config);

// 位置管理
mLocationManager = (LocationManager) this.getSystemService(Context.LOCATION_SERVICE);
mLocationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0,
new LocationListener() {
    @Override
    public void onLocationChanged(Location location) {
        // 位置发生变化,构造定位数据并更新位置
        LatLng desLatLng = locationConverter(location);
        MyLocationData.Builder locdata = new MyLocationData.Builder()
            .latitude(desLatLng.latitude)
            .longitude(desLatLng.longitude)
            .direction(currentRotation);
        mMapView.getMap().setMyLocationData(locdata.build());
    }
});

// 设置初始位置,第一次获取的位置为空
mLocationManager.getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
LatLng desLatLng = locationConverter(mLocationManager.getLastKnownLocation(
    LocationManager.NETWORK_PROVIDER));
MapStatus status = new MapStatus.Builder().target(desLatLng).build();
MapStatusUpdate mapStatusUpdate = MapStatusUpdateFactory.newMapStatus(status);
baiduMap.setMapStatus(mapStatusUpdate);

// 触摸屏幕切换图标模式
baiduMap.setOnMapTouchListener(new BaiduMap.OnMapTouchListener() {
    @Override
    public void onTouch(MotionEvent motionEvent) {
        if (clicktimes == 0) {
            clicktimes = 1;
            toggle.setImageResource(R.mipmap.define_location);
        }
    }
});

```

```

@Override
public void onSensorChanged(SensorEvent event) {
    switch (event.sensor.getType()) {
        // 加速度传感器变化,计算速度,判断是否摇一摇
        case Sensor.TYPE_ACCELEROMETER:
            accValues = event.values.clone();
            if (System.currentTimeMillis() - lastUpdateTime > 50) {
                lastUpdateTime = System.currentTimeMillis();
                float deltaX = accValues[0] - lastX;
                float deltaY = accValues[1] - lastY;
                float deltaZ = accValues[2] - lastZ;
                lastX = accValues[0];
                lastY = accValues[1];
                lastZ = accValues[2];
                double speed = Math.sqrt(deltaX * deltaX + deltaY * deltaY + deltaZ * deltaZ) * 200;

                // 速度超过阈值,前后两次间隔大于2秒
                if (speed >= SPEED_SHRESHOLD && lastUpdateTime - lastShakeTime > 2000 && !first) {
                    lastShakeTime = lastUpdateTime;
                    vibrator.vibrate(500);
                }
            }
            break;
    }
}

```

```

        if (baiduMap.getMapType() == BaiduMap.MAP_TYPE_SATELLITE) {
            mMapView.getMap().setMapType(BaiduMap.MAP_TYPE_NORMAL);
        }
        else {
            mMapView.getMap().setMapType(BaiduMap.MAP_TYPE_SATELLITE);
        }
    }
    first = false;
}
break;
case Sensor.TYPE_MAGNETIC_FIELD:
    magValues = event.values.clone();
    break;
}

// 传感器的值不为空
if (accValues != null && magValues != null) {
    float[] R = new float[9];
    float[] values = new float[3];
    SensorManager.getRotationMatrix(R, null, accValues, magValues);
    SensorManager.getOrientation(R, values);
    float newRotation = (float) Math.toDegrees((double) values[0]);

    if (Math.abs(newRotation - currentRotation) > 0.5) {
        if (ActivityCompat.checkSelfPermission(
            MainActivity.this, Manifest.permission.ACCESS_FINE_LOCATION) !=
            PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
            MainActivity.this, Manifest.permission.ACCESS_COARSE_LOCATION) !=
            PackageManager.PERMISSION_GRANTED) {return;}
        // 更新箭头位置和方向
        LatLng desLatLng = locationConverter(mLocationManager.getLastKnownLocation(
            LocationManager.NETWORK_PROVIDER));
        MyLocationData.Builder data = new MyLocationData.Builder();
        data.latitude(desLatLng.latitude);
        data.longitude(desLatLng.longitude);
        data.direction(newRotation);
        mMapView.getMap().setMyLocationData(data.build());

        // 箭头处于地图中心
        if (clicktimes == 0) {
            MapStatus status = new MapStatus.Builder().target(desLatLng).build();
            MapStatusUpdate mapStatusUpdate = MapStatusUpdateFactory.newMapStatus(status);
            mMapView.getMap().setMapStatus(mapStatusUpdate);
        }
    }
    currentRotation = newRotation;
}
}
}

```

五、实验结果



初始界面



地图放大



手机旋转180°



旋转一定角度



移动地图



移动到公教楼



点击图片按钮, 箭头回到中心



摇一摇手机振动, 切换地图模式

六、实验问题

1. 设置百度地图触摸监听器的时候，一开始点击地图没有反应，输出调试发现回调函数没有被调用，查阅百度地图API 后才发现，我是使用AS的补全功能，使用的是mMapView.setOnTouchListener()，而真正的地图触摸监听是 mMapView.getMap().setOnMapTouchListener()。
2. 设置地图定位的时候抛出空指针异常，原来getLastKnownLocation函数获取上一次的定位，而第一次运行时没有定位过，第一次获取到的位置为空。
3. 在进行地图定位时，输出调试发现经纬度是广州的经纬度没错啊，可是定位一直不正确，后来发现百度地图的坐标与网络定位的坐标不一样，需要先进行坐标转换。
4. 进行了坐标转换后再次进行网络定位还是不行，一打开还是在北平，内心是崩溃的，多次运行后我发现虽然一打开显示的是北平，但是箭头不见了，难道是定位准确但是地图没有移到箭头那里，缩小地图后发现箭头已定位在广州，于是顺便实现了箭头处于地图中心的功能。
5. 实现摇一摇功能时，一开始使用的是判读加速度的绝对值是否大于每个值，可是测试后发现加速度的波动非常大，很不稳定，有时拿起手机加速度就有60几，有时摇到手要断了还是不超过30，上网查阅摇一摇的实现后，发现有人是使用速度进行判断，于是改用该方法，多次测试确定合适的阈值。为了防止短时间内多次摇一摇，限制了相邻两次摇一摇的时间至少为两秒。

七、实验总结

本次实验难度适中，主要涉及传感器以及百度地图的使用，在实验过程中还是遇到了一些问题，不过基本都能通过查阅文档得以解决，经过本次实验，学会使用了传感器，使用位置服务获取经纬度信息以及使用百度地图API进行地图应用开发，百度地图还有很多的功能，本次实验只是用到很少的一部分，其他功能将在课程project中使用到。编写地图应用时应该注意getLastKnownLocation第一次获取到的位置为空；百度地图中的当前位置处于屏幕中心其实当人移动时箭头还是会移动，只是设置了中心后会将箭头拉回中心，看起来就像箭头一直处于中心，而当地图级别放到最大的时候，可以看出箭头一直在微弱的移动。