感知技术与应用

实验指导

——方向传感器

目录

[实验：方向传感器 3](#_Toc40791128)

[实验目的 3](#_Toc40791129)

[实验要求 3](#_Toc40791130)

[基础知识 3](#_Toc40791131)

[实验内容 6](#_Toc40791132)

[一、 指南针程序 6](#_Toc40791133)

[作业： 12](#_Toc40791134)

[注意事项 13](#_Toc40791135)

# 实验：方向传感器

## 实验目的

本次实验的目的是让大家了解Android中方向传感器的基本知识，掌握Android中方向传感器的使用方法。

## 实验要求

1. 了解Android中方向传感器基本知识
2. 掌握Android中方向传感器使用方法

## 基础知识

方向传感器基础

方向传感器也被称为姿态传感器，与加速度传感器不同，方向传感器主要用于感应设备方位的变化。本节将详细讲解方向传感器的基本知识。

方向传感器必备知识

在现实世界中，方向传感器通过对力敏感的传感器，感受手机等设备在变换姿势时的重心变化，使手机等设备光标变化位置从而实现选择的功能。方向传感器运用了欧拉角的知识，欧拉角的基本思想是将角位移分解为绕3个互相垂直轴的3个旋转组成的序列。其实，任意3个轴和任意顺序都可以，但最有意义的是使用笛卡儿坐标系并按一定的顺序所组成的旋转序列。

在学习欧拉角知识之前先介绍几种不同概念的坐标系，以便于读者理解欧拉角知识。

（1）世界坐标系

世界坐标系是一个特殊的坐标系，建立了描述其他坐标系所需要的参考框架。能够用世界坐标系描述其他坐标系的位置，而不能用更大的、外部的坐标系来描述世界坐标系。例如，“向西”“向东”等词汇就是世界坐标系中的描述词汇。

（2）物体坐标系

物体坐标系是和特定物体相关联的坐标系，每个物体都有它们独立的坐标系。当物体移动或改变方向时，和该物体相关联的坐标系将随之移动或改变方向。例如，“向左”“向右”等词汇就是物体坐标系中的描述词汇。

（3）摄像机坐标系

摄像机坐标系是和观察者密切相关的坐标系。在摄像机坐标系中，摄像机在原点，*x* 轴向右，*z* 轴向前（朝向屏幕内或摄像机方向），*y* 轴向上（不是世界的上方而是摄像机本身的上方）。

（4）惯性坐标系

惯性坐标系是为了简化世界坐标系到物体坐标系的转换而引入的一种新的坐标系。惯性坐标系的原点和物体坐标系的原点重合，但惯性坐标系的轴平行于世界坐标系的轴。

在欧拉角中，表示一个物体的方位用“Yaw-Pitch-Roll”约定。在这个系统中，一个方位被定义为一个Yaw角、一个Pitch角和一个Ron角。欧拉角的基本思想是让物体开始于“标准”方位，目的是使物体坐标轴和惯性坐标轴对齐。在标准方位上，让物体做Yaw、Pitch和Roll旋转，最后物体到达我们想要描述的方位。

（5）Yaw轴

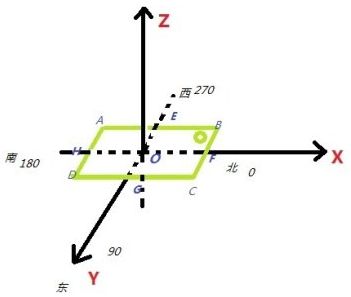
Yaw轴是3个方向轴中唯一不变的轴，其方向总是竖直向上，和世界坐标系中的*z*轴是等同的，也就是重力加速度g的反方向。如图12-5所示，为在图12-4的基础上手机绕Yaw轴旋转的效果。

（6）Pitch轴

Pitch轴方向依赖于手机沿Yaw轴的转动情况，即当手机沿Yaw转过一定的角度后，Pitch轴也相应围绕Yaw轴转动相同的角度。Pitch轴的位置依赖于手机沿Yaw轴转过的角度，好比Yaw轴和Pitch轴是两根焊死在一起成90度。

12.1.2　Android中的方向传感器

在Android系统中，方向传感器的类型是TYPE\_ORIENTATION，用于测量设备围绕3个物理轴（*x* ，*y* ，*z* ）的旋转角度，在新版本中已经使用SensorManager.getOrientation()替代。Android系统中的方向传感器在生活中的典型应用例子是指南针，接下来先来简单介绍一下传感器中3个参数*X* 、*Y* 、*Z* 的含义，如下图所示。



▲图　参数\_X\_、\_Y\_、\_Z\_

如上图所示，矩形ABCD表示一个手机，带有小圈那一头是手机头部，各个部分的具体说明如下所示。

传感器中的*X* ：如上图所示，规定*X* 正半轴为北，手机头部指向OF方向，此时*X*的值为0。如果手机头部指向OG方向，此时*X* 值为90，指向OH方向，*X* 值为180，指向OE，*X* 值为270。

传感器中的*Y* ：现在将手机沿着BC轴慢慢向上抬起，即手机头部不动，尾部慢慢向上翘起来，直到AD跑到BC右边并落在XOY平面上，*Y* 的值将从0～180之间变动，如果手机沿着AD轴慢慢向上抬起，即手机尾部不动，直到BC跑到AD左边并且落在XOY平面上，*Y* 的值将从0～180之间变动，这就是方向传感器中*Y* 的含义。

传感器中的*Z* ：现在将手机沿着AB轴慢慢向上抬起，即手机左边框不动，右边框慢慢向上翘起来，直到CD跑到AB右边并落在XOY平面上，*Z* 的值将从0～180之间变动，如果手机沿着CD轴慢慢向上抬起，即手机右边框不动，直到AB跑到CD左边并且落在XOY平面上，*Z* 的值将从0～180之间变动，这就是方向传感器中*Z* 的含义。

## 实验内容

### 指南针程序

本实验将演示在Android中使用方向传感器开发指南针应用程序的方法。在本实例中首先准备一张指南针素材图片，该图片上方向指南针指向北方。接下来开发一个检测方向的传感器，传感器程序可以检测到手机顶部绕*Z* 转过的多少度。在实例中需添加了一张图片，并让图片总是反转方向传感器返回的第一个角度值。



指南针图片素材

#### 1编写布局文件，具体实现代码如下所示。

编写布局文件main.xml，功能是插入准备好的素材图片，主要实现代码如下所示。

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:background="#fff"

>

<ImageView

android:id="@+id/znzImage"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:scaleType="fitCenter"

android:src="@drawable/compass" />

</LinearLayout>

#### 2编写程序文件，具体实现代码如下所示。

编写程序文件，使用传感器获取设备的旋转角度值，并根据这个值返回指南针的角度。

package zhinanzheng.com;

import android.app.Activity;

import android.hardware.Sensor;

import android.hardware.SensorEvent;

import android.hardware.SensorEventListener;

import android.hardware.SensorManager;

import android.os.Bundle;

import android.view.animation.Animation;

import android.view.animation.RotateAnimation;

import android.widget.ImageView;

public class Zhinanzheng extends Activity implements SensorEventListener{

ImageView image; //指南针图片

float currentDegree = 0f; //指南针图片转过的角度

SensorManager mSensorManager; //管理器

/\*\* Called when the activity is first created. \*/

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.main);

image = (ImageView)findViewById(R.id.znzImage);

mSensorManager = (SensorManager)getSystemService(SENSOR\_SERVICE); //获取管理服务

}

@Override

protected void onResume(){

super.onResume();

//注册监听器

mSensorManager.registerListener(this

, mSensorManager.getDefaultSensor(Sensor.TYPE\_ORIENTATION), SensorManager.SENSOR\_DELAY\_GAME);

}

//取消注册

@Override

protected void onPause(){

mSensorManager.unregisterListener(this);

super.onPause();

}

@Override

protected void onStop(){

mSensorManager.unregisterListener(this);

super.onStop();

}

//传感器值改变

@Override

public void onAccuracyChanged(Sensor sensor, int accuracy) {

// TODO Auto-generated method stub

}

//精度改变

@Override

public void onSensorChanged(SensorEvent event) {

// TODO Auto-generated method stub

//获取触发event的传感器类型

int sensorType = event.sensor.getType();

switch(sensorType){

case Sensor.TYPE\_ORIENTATION:

float degree = event.values[0]; //获取z转过的角度

//穿件旋转动画

RotateAnimation ra = new RotateAnimation(currentDegree,-degree,Animation.RELATIVE\_TO\_SELF,0.5f

,Animation.RELATIVE\_TO\_SELF,0.5f);

ra.setDuration(100);//动画持续时间

image.startAnimation(ra);

currentDegree = -degree;

break;

}

}

}

# 作业：

按照实验的内容进行操作，掌握Android中方向传感器的使用方法等内容。

完成实验内容后，设计一个测试用户拿着手机是否走直线的app，至少包含如下功能：

* + 显示当前手机x，y，z三个方向的方向值；
  + 通过方向值判断用户拿着手机是否一直处于直线行走状态。

# 注意事项

1. 独立自主完成实验任务，**切勿抄袭！**

如实验报告或代码有雷同现象则视为抄袭，**被抄袭者**与**抄袭者**实验**总成绩**均按**0分**处理。

1. 本次实验需提交如下内容：
2. 完整可运行的程序源代码。
3. 实验报告电子版。
4. 提交时间：

在规定之间内提交

1. 评分方法：
   * 达到基本作业要求，85分，根据实验报告质量和源码质量上下浮动5分
   * 作业的每项要求，少完成一项，分数减少3-5分，如控件数量不足、类别不足、没有消息相应处理等，最低分为0分
   * 超过基本作业要求，酌情加分，最高分100
   * 发现抄袭，0分
   * 作业未按时提交，0分