

一种基于多传感器数据融合的移动主体运行轨迹捕捉机制

毕朝国 徐利敏

(南京财经大学江苏省现代服务业研究院 南京 210003)

摘要 目前军事、医疗、科技、电影、游戏等很多应用领域都需要对移动主体的运行轨迹进行捕捉。现有的移动主体轨迹识别与绘制机制一般对设备的要求较高,且算法复杂,实时性不够理想。为此,提出一种基于多传感器数据融合的移动主体运行轨迹捕捉机制,以智能移动终端为载体,联合采用加速度传感器和姿态传感器采集数据,通过对加速度传感器和姿态传感器采集的数据进行处理和融合,并应用物理学中加速度和位移以及数学中曲线和直线的关系,准确识别移动主体的行动轨迹;然后利用光学透视投影的原理将轨迹投影到二维空间,并在智能终端的屏幕上绘制出来。实测结果表明,该机制具有较高的准确性和实时性,且具有理想的时空复杂度。

关键词 加速度传感器,姿态传感器,数据融合,运行轨迹捕捉

中图分类号 TP212.9 **文献标识码** A

Mobile Subjects Moving Track Capture Mechanism Based on Multi-sensor Data Fusion

BI Chao-guo XU Li-min

(Jiangsu Institute of Modern Service Industry, Nanjing University of Finance and Economics, Nanjing 210003, China)

Abstract At present, many fields, such as military, medical care, science and technology, movies, games and many other applications need to capture moving trajectories of moving objects. Existing mobile trajectory recognition and mapping methods generally have high requirements for equipment and have complex algorithms. What's more, their real-time performance is not ideal. This paper presented a multi-sensor data fusion based mobile trajectories capture mechanism with the intelligent mobile terminal as the carrier, combined with the use of acceleration sensor and attitude sensor to collect data. Through the fusion of the data and the application of the relationship between acceleration and displacement and the relationship between curve and straight line, accurate identification of moving body movement can be acquired. After that, the mobile trajectory can be projected onto a two-dimensional space using the principle of optical perspective projection and mapped on the screen of the intelligent terminal. Experimental results show that the mechanism with high accuracy and real-time performance has the ideal time and space complexity.

Keywords Acceleration sensor, Attitude sensor, Data fusion, Moving track capture

1 引言

利用传感器识别移动物体的姿态应用广泛。在军事上,通过获取目标的实时轨迹,可以实现战略目的;在医疗上,通过脉搏传感器和脑电波传感器分别获取患者的实时脉搏和脑电波变化图,有助于对患者的疾病进行病理性分析;在交通驾驶中,利用加速度传感器识别方向盘的运动,从而进一步分析出驾驶员的疲劳状态,及时提醒驾驶员远离危险驾驶;在游戏娱乐中,互动性比较强的拳击、棒球、网球、投篮等游戏就是利用各种传感器捕捉到参与者的移动轨迹、移动频率、力量大小等特点,进而做出相应的反应,如出拳、击球、投篮等。

本文提出了一种基于多传感器移动主体轨迹捕捉方法,主要包括数据采集模块、轨迹识别模块和轨迹捕捉显示模块。首先,利用加速度传感器采集移动物体的对应空间坐标系中

x, y, z 轴的三轴加速度,利用姿态传感器采集移动物体运动中 $Yaw, Pitch, Roll$ 轴的三轴旋转过的角度;然后对采集到的数据进行数据平滑处理;基于姿态传感器中三轴的角度和加速度传感器中三轴的加速度的角度以及加速度的数学关系,进行数据融合,得到移动物体对应空间坐标系三轴的实时运动加速度,进而通过物理学中加速度和速度的关系,得到移动物体实时运动的速度;用大量曲线模拟直线,实时捕捉移动物体运行轨迹并通过透视投影的方法在显示屏上显示。

2 相关研究

文献[1]利用时延神经网络获取物体的位置、形状、方向等参数,进而动态识别物体的运动轨迹。其主要方法是利用一段足够长并且可用的网络,目标类型通过输出层确定,通过调整偏移量以及轨迹平移不变性的特性实现物体轨迹识别。

本文受江苏省高校自然科学研究计划项目(14KJB520014),江苏高校优势学科建设工程项目(YXK999),江苏现代服务业协同创新中心项目,江苏现代服务业研究院专项科研基金项目(LSZ200003)资助。

毕朝国(1978—),男,硕士,讲师,主要研究方向为 Web 技术和物联网技术等;徐利敏(1980—),男,博士,副教授,主要研究方向为智能系统、智能商务、移动互联网。

文献[2]发明了一种数字笔,用户利用这个笔可以写出数字或者做出手势,手的移动加速度可以通过这个笔中内嵌的加速度传感器测量出来,通过无线通信的方式将数据传送到服务器,用服务器对笔的移动进行轨迹识别。该算法首先提取时间和频域特征的加速度信号,然后通过一种混合方法进一步确定重要的特征,这种方法是基于内核类的可分性来选择重要特征,通过线性判别分析来减少特征的维度。文献[3]基于MEMS三轴加速度传感器构建了3个不同的手势识别模型来识别7种手势。文献[4]也基于加速度传感器,利用随机投影理论实现手势识别。文献[5]提出了一种基于图像的轨迹识别方法。文献[6]通过在轨迹识别算法中改进数据分布构建了一种新的轨迹识别描述符。

上述方法的优点是都可以准确获取移动主体的移动轨迹,但是实时性不佳,对设备的要求比较高,而且时空复杂度也比较高。针对这种情况,本文提出了一种基于移动平台的实时性较好的移动主体轨迹识别方法。

3 基于多传感器的移动主体运行轨迹捕捉

3.1 多传感器数据采集

传感器是一种检测装置,能感受被测量的信息,并能将检测的感到的信息按一定的规律变换为电信号或其他所需形式的信息输出,以满足信息的传输、处理、存储、显示、记录和控制的要求。它是实现自动检测和自动控制的首要环节。

传感器的种类有很多,常见的传感器有加速度传感器、陀螺仪传感器、光传感器、姿态传感器、磁力传感器、温度传感器、距离传感器、压力传感器。人们说传感器是人类五官的延伸,这是对传感器的巨大作用的形象描述,不仅因为传感器可以获取准确可靠的信息,可以监视和控制生产过程中的各个参数,更因为借助于传感器,可以推动科学的发展,进而推动人类社会的向前发展。

加速度传感器就是这众多传感器的一种,它是一种能够测量加速力的电子设备^[7]。加速力就是当物体在加速过程中作用在物体上的力。加速度传感器包括压电式、压阻式、电容式和伺服式4种。本文用到的Android中的加速度传感器属于压电式加速度传感器,该类型传感器的原理是利用压电陶瓷或石英晶体的压电效应,在加速度计受振动时,质量块加在压电元件上的力也随之变化。当被测振动频率远低于加速度计的固有频率时,力的变化与被测加速度成正比。

Android中的加速度传感器通过获取空间坐标系X、Y、Z3个分量来计算手机在某一时刻、某一方向上的加速度。在注册传感器监听器后,加速度传感器主要捕获3个参数Values[0]、Values[1]、Values[2],分别对应于空间坐标系中的X、Y、Z3个分量,如图1所示。

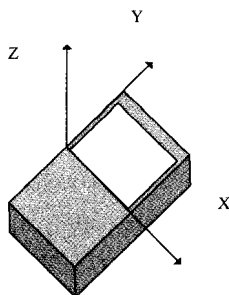


图1 Android中加速度传感器三轴的方向

如图1所示,当手机处于水平面时,加速度传感器三轴方向和标准空间坐标系保持一致,当手机不在处于水平面时,X、Y相对手机的位置保持不变,Z轴还是垂直向上,因此手机移动时可以实时测量出手机沿着3个方向的运动加速度,测量值以 m/s^2 为单位,但是测量数据包含重力 g 的影响。实验中终端圆周移动时获取的加速度数据如图2所示。

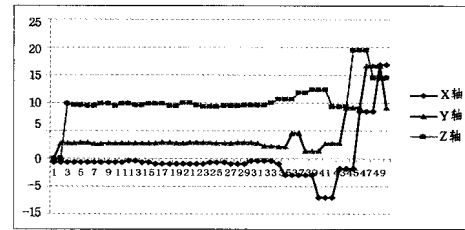


图2 加速度传感器各轴监测数据

姿态传感器也是一种很重要的传感器。百度百科对姿态传感器^[8]有如下定义:姿态传感器(E. T-ahrs)是基于MEMS技术的高性能三维运动姿态测量系统。它包含三轴陀螺仪、三轴加速度计(即IMU)、三轴电子罗盘等辅助运动传感器,通过内嵌的低功耗ARM处理器输出校准过的角速度、加速度、磁数据等,通过基于四元数的传感器数据算法进行运动姿态测量,实时输出以四元数、欧拉角等表示的零漂移三维姿态数据。本文使用的Android中的姿态传感器和百度百科中定义的传感器不同,该传感器主要感应手机方位的变化,捕获的同样是3个数,分别代表手机沿Yaw轴、Pitch轴和Roll轴转过的角度,如图3所示。

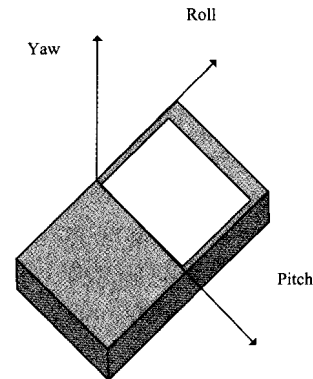


图3 Android中姿态传感器示意图

Yaw轴、Pitch轴和Roll轴与平时我们理解的空间坐标系有所不同,下面分别对这3个轴所表示的含义进行详细介绍^[9]。

Yaw轴:该轴是3个轴中最简单的一个,其表示的方向是不变的,一直是重力加速度 g 的反方向,即一直是竖直向上的,与手机的姿态无关。

Pitch轴:该轴的方向并不是固定不变的,而是会随着手机沿Yaw轴旋转而改变,唯一不变的关系是该轴永远与Yaw轴成 90° 。实际上Yaw轴与Pitch轴相当于焊到一起的一个 90° 支架,无论手机怎么旋转,其与Yaw轴的角度都为 90° 。

Roll轴:该轴是沿着手机屏幕向上的轴,无论手机是何种姿态,Roll轴都是沿着手机的屏幕向上的,其方向是与手机绑定的。

实验中终端进行圆周移动时获取的姿态数据如图4所示。

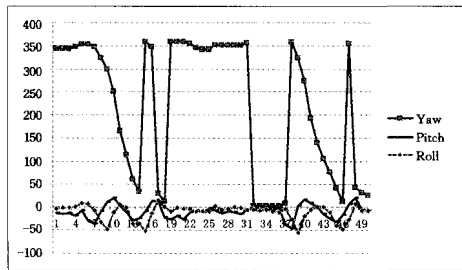


图4 姿态传感器各轴监测数据

3.2 数据平滑处理

大部分时间序列都存在一个重要问题即存在噪音,也就是某个值的大小随机变化。现实世界中很多数据集里都存在噪音,但我们通常只通过改善测量数据的仪器或采集一个大一点的样本再求平均的方法来减少噪音。然而事实上这种做法并不理想,并不能很好地平滑数据。

我们利用传感器采集到的数据或多或少存在着误差,所以在利用采集到的数据进行移动物体轨迹识别之前,有必要对数据进行基本的数据平滑处理。这里采用加权移动平均法^[10,11],越靠近平滑窗口边缘的点权值越小。使用这种加权平均值时,进入平滑窗口的点都是逐渐地计入平均值中,再逐渐消除:

$$s_i = \sum_{j=-k}^k w_j x_{i+j} \quad (1)$$

式中, $\sum_{j=-k}^k w_j = 1$, s_i 代表位置 i 的平滑值, x_{i+j} 代表数据点, w_j 代表权重因素。例如对于 5 点移动平均法,我们可以使用 $(1/16, 1/8, 1/2, 1/4, 1/16)$ 作为权值。选择什么样的权值并不重要,只要中间为最大值,越靠近边缘的值越小,并让总和为 1。为了得到更好的数据平滑处理效果,本文使用了 3 点加权移动平均法,使用 $(1/4, 1/2, 1/4)$ 作为权值,对于采集到的加速度和姿态传感器数据,每 3 个点保存为一个 float 数组,进行平滑处理后返回一个新的 float 数组,该数组保存到相应的数据集中。具体平滑处理算法实现代码如下所示。

```
public float[] SmootheData(float[] a){
    //定义 3 点移动平均法的权重
    float w1=0.25f;
    float w2=0.5f;
    float w3=0.25f;
    //改变 3 个点中中间的数据的值
    a[1]=w1 * a[0]+w2 * a[1]+w3 * a[2];
    return a;
}
```

3.3 数据融合

多传感器信息融合^[12] 又称多传感器数据融合,指的是对不同知识源和多个传感器所获得的信息进行综合处理,消除多传感器信息之间可能存在的冗余和矛盾,利用信息互补,降低不确定性,以形成对系统环境相对完整一致的理解,从而提高智能系统决策、规划的科学性、反应的快速性和正确性,进而降低决策风险的过程。

由于加速度传感器采集到的数据包含重力 g ,而我们实际应用中需要排除重力 g 对数据的影响,因此我们将加速度传感器和姿态传感器采集的数据融合起来,消除重力 g 对加

速度采集到的数据的影响。对于加速度传感器如何与姿态传感器融合才能消除重力的影响,文献^[13]采用了如下的算法:

手机已经经过了任意翻转,我们设定其中的某一个状态为手机的初始状态,由于重力加速度的方向是始终竖直向下,此时手机屏幕平面与重力加速度的方向会呈现一定的夹角。

根据算法思想,需要求出重力加速度向量在手机屏幕上的投影,将手机沿着 *Yaw* 轴、*Pitch* 轴和 *Roll* 轴依次进行旋转。我们设定重力加速度的方向与手机屏幕所在平面保持相对关系不变,相当于把重力加速度向量固定在手机初始状态的屏幕平面上。因此,我们沿着 *Yaw* 轴逆方向旋转和刚刚同样的角度。

在沿着 *Pitch* 轴逆向旋转进行角度恢复之后,此时的 *Roll* 轴便对应于标准空间坐标系中的 *Y* 轴。而原先在手机屏幕平面上的重力加速度向量现在也于手机屏幕有了一定的夹角。

在沿着 *Roll* 轴进行旋转以后,此时 *Roll* 轴和 *Pitch* 已经处于手机屏幕平面上,此时的手机屏幕平面也就是标准空间坐标系的 *X-Y* 平面。不难发现,手机初始状态下的重力加速度向量此时已经位于手机屏幕的平面上,并与 *Roll* 轴和 *Pitch* 轴相交有一定的夹角。

受到该方法的启发,本文对该方法进行改进,应用了一种更加简便的方法。具体如下:

假设任意状态下,重力 g 与手机平面有个夹角 α ,如图 5 所示。

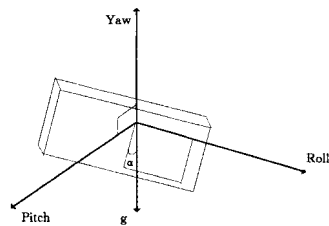


图5 初始状态下重力与手机屏幕的位置关系

重力对手机的作用力仅和手机的相对位置有关,手机沿着 *Roll* 转过 α 角度就落在了平面上,这个投影就是作用在手机上的作用力。对应水平状态下重力对手机的作用力如图 6 所示。

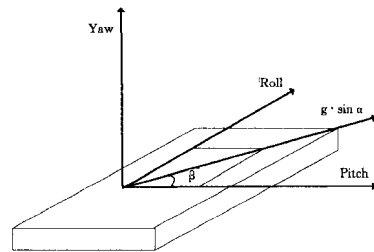


图6 水平状态下重力与手机屏幕的位置关系

投影向量 $g \cdot \sin \alpha$ 与 *Pitch* 轴的夹角为 β , β 为沿着 *Yaw* 轴旋转的角度。重力 g 在 *Pitch* 轴上的作用力 $g \cdot \sin \alpha \cdot \cos \beta$ 对应应在加速度传感器 *X* 轴上的作用力,在 *Roll* 轴上的作用力 $g \cdot \sin \alpha \cdot \cos \beta$ 对应应在加速度传感器上 *Y* 轴的作用力。

```
//collection1 保存 X 轴的加速度
ArrayList<Float> collection1=new ArrayList<Float>();
```

```

//collection2 保存 Y 轴的加速度
ArrayList<Float> collection2=new ArrayList<Float>();
//collection3 保存沿 Roll 轴旋转的角度
ArrayList<Float> collection3=new ArrayList<Float>();
//collection4 保存沿 Yaw 轴旋转的角度
ArrayList<Float> collection4=new ArrayList<Float>();
//保存融合后 X 轴的加速度
ArrayList<Float> collection1_1=new ArrayList<Float>();
//保存融合后 Y 轴的加速度
ArrayList<Float> collection2_1=new ArrayList<Float>();
//得到轨迹的集合
public void CatchTrajectory(){
    for(int i=0;i<collection1.size();i++){
        //作用力方向判定
        if(collection3.get(i)>0){
            //重力在 X 轴上的作用力
            collection1_1.add((float)((float)
(collection1.get(i)-g*Math.sin(collection3.get(i)))*Math.abs
(Math.cos(collection4.get(i)))));
            //重力在 Y 轴上的作用力
            collection2_1.add((float)((float)
(collection2.get(i)-g*Math.sin(collection3.get(i)))*Math.abs
(Math.sin(collection4.get(i)))));
        }
    }
}

```

3.4 轨迹绘制

本文利用物理学的相关知识,构建了一种轨迹绘制的方法。经过数据的平滑处理,可以利用这些处理过的数据进行轨迹绘制。由前面的叙述可知,加速度传感器和姿态传感器的融合主要是排除重力 g 的影响,绘制的具体方法的原理是利用物理学中位移 s 和加速度 a 的关系 $s_t = v_0 * t + \frac{1}{2} at^2$, 得到移动物体三轴实时运动的坐标点,而其中的每个 v_0 是经过单位 t 时间里移动物体的实时速度,该实时速度由加速度与速度的关系 $v_{t_2} = v_{t_1} + at$ 得到,其中 t_1, t_2 的间隔时间取 5ms (当然可以取更短的时间间隔,这样绘制的轨迹更加准确,取 5ms 的原因是手机中 CPU 处理数据水平有限),加速度 a 为 t_1 时刻的加速度,因为在很短的时间内,物体的加速度变化可以忽略不计。

由此我们得到了移动物体运动的大量的空间点,相邻两点之间连线就可以表示移动物体空间运动的轨迹,因为需要显示在 2D 的显示设备上,所以接下来对绘制的轨迹利用透视投影的思想。

透视投影^[14]属于非平行投影,观察者的视线在远方是相交的,视线的相交处即为不可见的点。透视投影是用中心投影法将形体投射到投影面上,从而获得一种较为接近视觉效果的单面投影图。它具有消失感、距离感、相同大小的形体呈现出有规律的变化等一系列的透视特性,能逼真地反映形体的空间形象。

选择透视投影的主要原因是通过透视投影可以等比例缩小物体移动轨迹,方便在移动设备上显示。

4 原型系统构建与实验分析

4.1 开发环境和验证环境

基于多传感器的移动物体轨迹捕捉的开发环境是在 Windows 平台下,使用软件 eclipse 和在 eclipse 中配置的插件 ADT(Android Development Toolkit),其中 eclipse 的版本是 3.6.2,ADT 版本是 20.0.3.v201208082019-427395。

Eclipse 是 IBM 的一款软件产品,Eclipse 是一个开放源代码的、基于 Java 的可扩展开发平台。就其本身而言,它只是一个框架和一组服务,用于通过插件组件构建开发环境。幸运的是,Eclipse 附带了一个标准的插件集,包括 Java 开发工具包 JDK(Java Development Kit)。ADT 是 Android 在 eclipse 中的开发工具包,提供必要的 API。

验证环境为 Android 平台,版本为 Android 2.3。Android 是一个崭新的手机操作系统平台,在以 Google 为首的 40 多家开放手机联盟(OHA)的支持下,使用开放源码策略,平台全部以开源的项目来构建,是完全开放的操作系统平台。Android 系统的快速发展使得移动终端的发展趋向多元化,其强大的网络功能整合了 Google 网络服务优势,使得 Android 在移动技术发展方面有着光明的前景。目前 Android 已由刚开始的 Android1.0 版本经过多个版本的改进与升级,发展至现在的 Android 3.1、Android 4.0、Android 4.1、Android 4.2 版本,平台应用方面更加成熟和完善。

4.2 系统实现

系统包含 Login、MainActivity、MainAccelerometer、MyOrientation 这 4 个类,MainActivity 类是登陆实现类,登陆后显示主界面,Login 类是主界面的显示。MainAccelerometer 实现 Runnable 接口,MainAccelerometer 类是系统的核心实现类,主要实现移动主体轨迹捕捉和显示的功能;该类引用了 MyOrientation 类中的数据,MyOrientation 类的功能是实时获取姿态传感器三轴的数据。

轨迹捕捉功能实现的核心代码如下所示。

```

//核心绘制轨迹的方法,传感器采集到的数据发生变化时响应该方法
public void onSensorChanged(SensorEvent event){
    MyOrientation mo=new MyOrientation();
    MyAccelerometer mr2=new MyAccelerometer();
    Thread t1=new Thread(mo);
    Thread t2=new Thread(mr2);
    t1.start();
    try{
        Thread.sleep(5);
    }
    catch(InterruptedException e){
        e.printStackTrace();
    }
    t2.start();
    float []values=event.values;
    x=values[0];
    y=values[1];
    z=values[2];
    Paint paint=new Paint();
    String familyName="宋体";
}

```

```

Typeface font=Typeface.create(familyName, Typeface.BOLD);
paint.setColor(Color.RED);
paint.setTypeface(font);
paint.setTextSize(22);
float t=0.005f;//设定获取数据时间间隔,每5ms获取一次数据
a1=x;
b1=y;
c1=z;
x1=x1+va0*t+0.5f*(a1-a0)*t*t;
y1=y1+vb0*t+0.5f*(b1-b0)*t*t;
z1=z1+vc0*t+0.5f*(c1-c0)*t*t;
paint.reset();
collection1.add(x0);
collection1.add(x1);
collection2.add(y0);
collection2.add(y1);
//通过循环,不断取得移动物体移动过程中的位置点,实时显示位置变化
for(int k=0;k<collection1.size()-1;k++){
paint.setColor(Color.RED);
paint.setAlpha(10);
paint.setTextSize(26);
canvasTemp.drawLine(160+(Float)(10*Data.collection1_1.get(k)),160+(Float)(10*Data.collection2_1.get(k)),160+(Float)(10*collection1_1.get(k+1)),160+(Float)(10*collection2_1.get(k+1)),paint);
paint.reset();
}
x0=x1;
y0=y1;
z0=z1;
va0=va0+(a1-a0)*t;
vb0=vb0+(b1-b0)*t;
vc0=vc0+(c1-c0)*t;
a0=a1;
b0=b1;
c0=c1;
}

```

系统主界面如图 7 所示。



图 7 系统主界面

图 8—图 10 展示了系统对移动物体近似直线,数字 9,字母 U 移动的实时轨迹捕捉。



图 8 近似直线运动轨迹捕捉



图 9 近似数字 9 轨迹捕捉

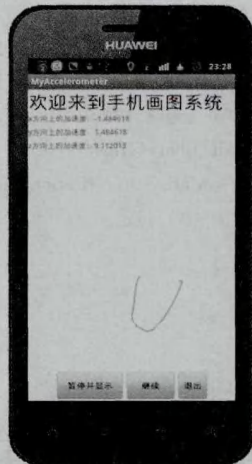


图 10 近似字母 U 轨迹捕捉

由于本方法是实时捕捉运动轨迹,程序开始运行后系统就会实时捕捉移动物体运动轨迹,移动物体的每次微小移动都会影响到轨迹的形成,因此模拟一个图形的轨迹时会得到一个相似的图。

4.3 性能分析

基于多传感器移动主体运行轨迹捕捉方法的优点体现在以下几个方面:

1) 时间复杂度

算法执行一次的时间复杂度是线性时间复杂度 $O(n)$,系统响应比较快,时间复杂度良好;且程序运行时仅占用内存 3.5M,不影响其他程序运行。

2) 实时性

算法每隔 5ms 捕获移动主体运行轨迹并可以实时地显示在移动终端上,可见算法具有较好的实时性。

3) 准确性

算法利用了物理学和计算机学的相关知识,实验中对物体近似直线、数字 9 和字母 U 移动的实时轨迹捕捉表明,即使是微小的变化也能在系统中体现出来,这表明系统可以准确获取移动物体的运行轨迹。

4) 对设备的要求

算法对设备的要求不高,随处可见的智能移动终端都可以作为本方法实现的载体。

5)耗电量

监测程序运行十分钟的耗电量为电池总电量的0.144%,其中检测手机电池容量为1930mAh,可见程序的耗电量非常低。

结束语 随着传感器的发展和普及,基于传感器的应用层出不穷,基于多传感器的移动物体轨迹识别是众多应用中很重要同时也是很有发展前景的一种方法。本文基于加速度传感器和姿态传感器提出了一种移动物体行动轨迹的实时捕捉方法,其在保证准确性的同时降低了对设备的要求,同时具有较好的实时性。但是该方法仍有很大的改进空间,如显示界面有待美化;功能比较单一,基础捕捉的轨迹可以开发一些有趣的应用;对于算法的识别准确性可以进一步的提高;由于移动终端内存有限,可以通过移动终端和PC通信,把处理的权限转移到功能更加强大的PC机上。

参考文献

- [1] Wang J S, Chuang F C. An Accelerometer-Based Digital Pen with a Trajectory Recognition Algorithm for Handwritten Digit and Gesture Recognition [J]. IEEE Transactions On Industrial Electronics, 2012, 59(8): 2998-3007
- [2] Wang L, Cheng J. Object Tracking and Trajectory Recognition Using Improved CAMSHIFT and Hidden Markov Model [J]. Automatic Control and Artificial Intelligence International Conference, IEEE, 2012; 1451-1454
- [3] Xu R, Zhou S L, Li W. MEMS Accelerometer Based Nonspecific User Hand Gesture Recognition [J]. IEEE Sensors Journal, 2012, 12(5): 1166-1173
- [4] Alk A, Chen F, ValaeS. A Novel Accelerometer-Based Gesture Recognition System [J]. IEEE Transactions On Signal Processing, 2011, 59(12): 6197-6205
- [5] Yun L, Zhang J X, Xing J G. Video object tracking based on position prediction guide CAMSHIFT[C]//ICACTE; Proceedings of the 2010 3rd International Conference on Advanced Computer Theory and Engineering. New York: ACM Press, 2010; 159-164
- [6] Yang J Y, Li Y F, Wang K Y. A New Descriptor for 3D Trajectory Recognition via Modified CDTW [R]. Hong Kong and Macau, 2010
- [7] 百度百科. 加速传感器 [EB/OL]. (2013-05-07) [2013-06-25]. <http://baike.baidu.com/view/3506030.htm>
- [8] 百度百科. 姿态传感器 [EB/OL]. (2012-04-23) [2013-06-25]. <http://baike.baidu.com/view/3843821.htm>
- [9] eoe 移动开发者社区. 姿态传感器 [EB/OL]. (2011-07-19) [2013-06-25]. <http://www.eoeandroid.com/thread-90317-1-1.html>
- [10] 赵圣占, 谷阳阳. 基于加权移动平均的遥测数据量化误差修正方法 [J]. 仪器仪表用户, 2011, 18(4): 94-96
- [11] 孟凡荣, 庄朋, 闫秋艳. 基于加权移动平均的数据流预测模型 [J]. 计算机应用研究, 2009, 26(10): 3680-3682, 3686
- [12] 张延龙, 王俊勇. 多传感器数据融合技术概述 [J]. 舰船电子工程, 2012, 33(2): 41-44
- [13] 程展传. 基于 Android 平台的传感器技术应用研究 [D]. 广州: 华南理工大学, 2011; 50-53
- [14] 吴亚峰, 于复兴. Android 应用开发完全自学手册——核心技术、传感器、2D/3D、多媒体与典型案例 [M]. 北京: 人民邮电出版社, 2012; 251-252
- [80] Azuma R T, Hoff B R, Neely III H E, et al. Making augmented reality work outdoors requires hybrid tracking [C]//Proceedings of the First International Workshop on Augmented Reality, 1998. New York, USA: IEEE, 1998; 219-224
- [81] 张钰, 陈靖, 王涌天, 等. 增强现实浏览器的密集热点定位与显示 [J]. 计算机应用, 2014, 34(5): 1435-1438, 1462
- [82] Hesch J A, Kottas D G, Bowman S L, et al. Consistency Analysis and Improvement of Vision-aided Inertial Navigation [J]. Robotics IEEE Transactions on, 2014, 30(1): 158-176
- [83] Hesch J A, Kottas D G, Bowman S L, et al. Camera-IMU-based localization; Observability analysis and consistency improvement [J]. International Journal of Robotics Research, 2014, 33(1): 182-201
- [84] Rekimoto J, Nagao K. The world through the computer; Computer augmented interaction with real world environments [C]//Proceedings of the 8th annual ACM symposium on User interface and software technology, 1995. New York, USA: ACM, 1995; 29-36
- [85] Lalonde J F, Efros A A, Narasimhan S G. Webcam clip art; Appearance and illuminant transfer from time-lapse sequences [J]. ACM Transactions on Graphics (TOG), 2009, 28(5): 131
- [86] Krivánek J, Ferwerda J A, Bala K. Effects of global illumination approximations on material appearance [C]//ACM Transactions on Graphics (TOG), 2010. NY, USA: ACM, 2010, 29(4): 112
- [87] Wloka M M, Anderson B G. Resolving occlusion in augmented reality [C]//Proceedings of the 1995 symposium on Interactive 3D graphics, 1995. NY, USA: ACM, 1995; 5-12
- [88] Fischer J, Regenbrecht H, Barattoff G. Detecting dynamic occlusion in front of static backgrounds for AR scenes [C]//Proceedings of the workshop on Virtual environments 2003. NY, USA: ACM, 2003; 153-161
- [89] Livingston M A, Swan II J E, Gabbard J L, et al. Resolving multiple occluded layers in augmented reality [C]//Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality, 2003. DC, USA: IEEE Computer Society, 2003; 56
- [90] Fortin P A, Hebert P. Handling occlusions in real-time augmented reality; dealing with movable real and virtual objects [C]//The 3rd Canadian Conference on Computer and Robot Vision, 2006. IEEE, 2006; 54-54
- [91] Assad M, Carmichael D J, Cutting D, et al. AR phone: Accessible augmented reality in the intelligent environment [C]//Proceedings of OZCHI2003, 2003. Brisbane, Australia: University of Queensland, 2003; 232-237
- [92] Hakkarainen M, Woodward C, Billinghurst M. Augmented assembly using a mobile phone [C]//Proceedings of the 7th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2008. New York, USA: IEEE, 2008; 167-168
- [93] Hile H, Borriello G. Information overlay for camera phones in indoor environments [C]//Proceedings of the Location and Context-Awareness, Third International Symposium, 2007. Heidelberg, Berlin, Germany: Springer, 2007; 68-84
- [94] Bruns E, Brombach B, Zeidler T, et al. Enabling mobile phones to support large-scale museum guidance [J]. IEEE Multimedia, 2007, 14(2): 16-25