

# 基于体感技术的手势追踪与识别<sup>①</sup>

朱明茗, 景红

(西南交通大学 信息科学与技术学院, 成都 610031)

**摘要:** 体感技术使人们更直接的通过自己的肢体动作与电脑设备产生互动, 减小了鼠标、键盘等传统输入设备带来的束缚. 使用 Kinect 体感设备, 对体感关键技术及 Kalman 滤波器算法进行了深入的研究, 并在此基础上提出了一种基于骨骼坐标的手势识别算法. 最后详细阐明了基于 Kinect 的手势追踪与识别系统的具体设计与实现方法.

**关键词:** 体感技术; Kinect; 手势追踪; 手势识别; Kalman

## Gesture Tracking and Recognition Based on Motion Sensing Technology

ZHU Ming-Ming, JING Hong

(School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China)

**Abstract:** Motion Sensing Technology makes people interact with computer more directly and reduces the constraints of mouse and keyboard. This paper studies the key technology of Motion Sensing Technology and Kalman filtering algorithm deeply. At the same time it presents a gesture recognition algorithm based on skeleton coordinates. At last this paper expounds the design and implementation of Gesture Tracking and Recognition System based on Kinect.

**Key words:** motion sensing technology; Kinect; gesture tracking; gesture recognition; Kalman

体感技术也称为动作感应控制技术, 它使人们可以通过更为自然、更具直觉性的肢体动作对机器发出指令. 根据体感方式与原理的不同, 体感技术可分为惯性感测、光学感测以及惯性和光学联合感测<sup>[1]</sup>. 其中, 惯性感测主要以惯性传感器为主, 如用重力传感器、磁传感器等传感设备探测人体动作的物理参数, 并根据这些参数计算出人体动作; 光学感测则主要是通过光学传感器获取人体影像, 根据获取的人体影像分析人体动作, 进而达到操纵设备的效果; 而惯性和光学联合感测就是将不同的感测原理结合起来, 如 WII 就是使用重力传感器和红外线传感器实现动作的侦测<sup>[2]</sup>.

微软的 Kinect 是一种利用光学感测原理识别人体动作的 3D 体感摄影机, 它能够准确的追踪人体骨骼、识别人体的动作, 并实时处理相应操作<sup>[3]</sup>. 通过将 Kinect 体感设备与计算机结合起来, 用户无需使用任何输入设备, 即可通过自身的肢体动作向计算机发布指令, 使之对计算机的控制具有更大的灵活性, 进而增加了用户与计算机的交互性和用户体验.

随着 Kinect 的推出, 它已被应用于游戏、电子商务、医疗、教育等多个领域. 例如: 模拟 3D 试衣间, 客户可以快速试穿衣服; 协助医生做外科手术, 降低手术的时间和风险; 模拟化学实验室, 降低了学生实验的危险性.

本文通过使用 Kinect 体感设备, 基于 OpenNI/NITE 开发平台, 采用 Kalman 滤波器, 提出了基于坐标的手势识别算法, 达到对手势的追踪与识别的目的.

## 1 Kinect关键技术介绍

### 1.1 深度提取

Kinect 与传统摄像头最大的区别在于它不仅可以获得彩色信息, 同时, 在可视范围内, 能够采集物体的深度信息<sup>[4]</sup>. 它的基本原理是: 激光照射粗糙物体会产生衍射斑点, 这些衍射斑点具有高度的随机性, 并且随着距离的变化, 散斑也随着改变, 在同一空间中不同的散斑图案都不相同. 因此, 只要使用散斑对

<sup>①</sup> 收稿时间:2013-12-09;收到修改稿时间:2014-01-07

空间编码, 当有物体进入空间时, 即可定位。

Kinect 首先利用 2 个 3D 摄像头对测量空间进行散斑编码, 并读取编码的光线; 接下来晶片对读取的编码计算, 进而产生深度图像<sup>[3]</sup>。从 Kinect 产生的深度图像如图 1 所示。



图 1 Kinect 深度图像

### 1.2 Kinect 的骨骼追踪

Kinect 在实现深度信息提取的基础上, 能够实现对人体骨骼的追踪。它通过对深度信息的处理, 计算人体各个部分的位置信息, 通过(x,y,z)坐标表示<sup>[5,6]</sup>。使用不同的开发平台, 定位的关节点也有区别, 例如, 在 OpenNI 中定义了 24 个关节点, 目前 15 个可以使用, 如图 2 所示。图 3 为使用 OpenNI 提取的骨骼图像。

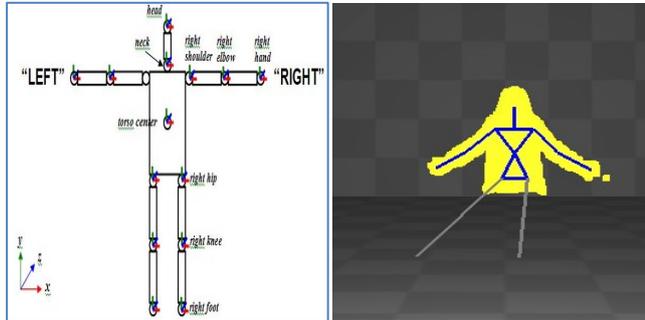


图 2 关节点分布图

图 3 Kinect 骨骼图像

## 2 Kalman 滤波跟踪

目标跟踪是指确定视频序列中每幅图像运动目标的位置, 监视目标从进入到离开摄像头可视范围的全过程。

Kalman 是一种高效的递归滤波器。它可以根据系统的历史测量值, 建立最大化后验概率的系统状态模型, 对下一状态做最优估计, 有效地解决了状态空间模型的估计与预测。其基本思想是以系统的状态空间模型为基础, 利用前一时刻的状态估计值和现在时刻的状态测量值来更新对状态变量的估计, 求解出现在时刻的状态估计值。

k 时刻系统状态和观测方程如式(1)~(2)所示:

$$x_k = Fx_{k-1} + Bu_k + w_k \tag{1}$$

$$z_k = H_k x_k + v_k \tag{2}$$

其中的  $x_k$  是 k 时刻描述系统的状态,  $F$  是  $n \times n$  的传递矩阵, 表示系统的参数。  $B$  是联系输入控制和状态改变的  $n \times c$  矩阵。 变量  $w_k$  为过程噪声;

$z_k$  表示 k 时刻测量值。  $H_k$  是  $m \times n$  矩阵, 描述测量系统的相关参数。  $v_k$  是测量噪声, 被假设为服从高斯分布  $N(0, R_k)$ 。

由此, 便可以计算时刻 t 状态的先验估计  $x_k^-$  及  $x_k^-$  的协方差  $P_k^-$ , 表达式如(3)~(4)所示。

$$x_k^- = Fx_{k-1} + Bu_{k-1} + w_k \tag{3}$$

$$P_k^- = FP_{k-1}F^T + Q_{k-s} \tag{4}$$

式(3)~(4)为 Kalman 的五个核心方程式的前两个, 他们构成了 Kalman 滤波器预测部分。

接下来便可以计算 Kalman 更新率  $K_k$ ,  $x_k$  和  $P_k$  最优的更新值, 表达式如(5)~(7)所示:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{5}$$

$$x_k = x_k^- + K_k (z_k - H_k x_k^-) \tag{6}$$

$$P_k = (I - K_k H_k) P_k^- \tag{7}$$

式(5)~(7)就构成了 Kalman 滤波器的修正部分。

Kalman 滤波器的预测阶段, 用从过去得到的信息进一步修正模型以取得下一状态将出现的信息; 校正阶段对基于前一次的模型进行调整。这两个阶段如图 4 所示<sup>[7-10]</sup>。

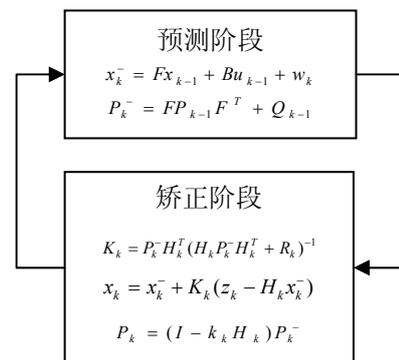


图 4 Kalman 滤波器算法示意图

## 3 基于 Kinect 的手势追踪与识别系统

本研究主要使用 Kinect 体感设备, 使用 OpenNI/NITE 平台, 实现人体手势识别, 进而对计算

机等设备达到控制的效果. 系统框架流程图如图 5 所示.

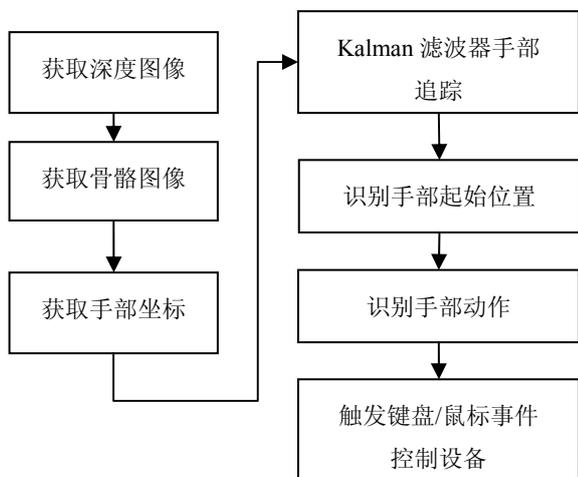


图 5 系统实现流程图

### 3.1 OpenNI/NITE 平台下的骨骼提取

OpenNI 骨架由关节点(joint)构成, 每一个关节点都包含位置信息.

① 首先获取深度图像, 提取人物信息, 因此需要使用 OpenNI 中 DepthGenerator 和 UserGenerator 生产节点, 用来产生深度图像和人物信息.

② 根据获取的人物信息生成骨骼信息. 为了正确关节点位置, 使用 NITE 中预先定义的矫正用姿势“Psi”, 如图 6 所示.

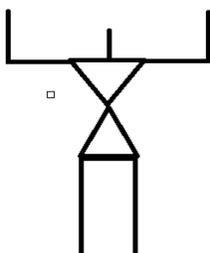


图 6 “Psi”姿势图

③ 当人物摆出的“Psi”姿势且 NITE 正确识别后, 调用 Skeleton 进行人体骨骼的校正和分析.

④ 当骨骼校正成功后, 便可对骨骼关节点追踪, 通过 GetSkeleton 函数获取骨骼点信息; 如果骨骼校正失败, 则重新校正. 图 7 为 OpenNI/NITE 识别骨骼信息流程图.

当骨骼追踪成功后, 可通过 OpenNI 中的 GetSkeletonJoint()来读取骨骼的信息. 函数具体声明如下: XnStatus xn:: Skeleton Capability :: GetSkeleton

Joint (XnUserID user,XnSkeletonJoint eJoint, XnSkeleton JointTransformation & joint);其中, user 为追踪的用户的 ID; eJoint 指定系统要追踪哪个关节, 这里我们追踪的是用户的右手信息, 因此可设为 XN\_SKEL\_RIGHT\_HAND; joint 则用来存储要追踪的关节点信息.

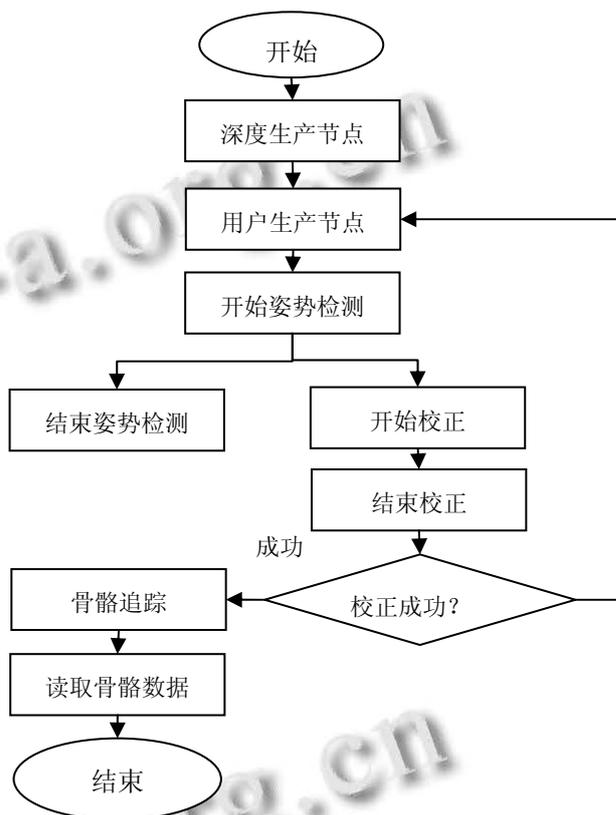


图 7 OpenNI/NITE 识别骨骼信息流程图

### 3.2 Kalman 滤波器的手部追踪

由于 Kalman 滤波器具有轨迹预测功能, 能够根据目标历史位置预测目标下一状态的位置. Kinect 通过提取深度信息实现对骨骼的追踪, 对光照和颜色具有鲁棒性, 更便于追踪.

由于 Kinect 采集图像的速率是每秒 30 帧, 相邻两帧之间相隔时间 $\Delta t$ 很短, 因此可以认为相邻两帧之间的手部运动是匀速的<sup>[9]</sup>. 我们可以将手的状态用位置  $x, y$  和速度  $v_x, v_y$  表示. 这四个状态变量组成的状态向

$$\text{量 } x_k = \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}, \text{ 可以得出传递矩阵 } F = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

将  $x_k$  和  $F$  代入上式(3)中, 得出  $k$  时刻手部信息的先验估计  $x_k^-$ ; 接下来将  $k-1$  时刻的协方  $P_{k-1}$  和  $F$  代入上式(4)中, 对进行预测; 将预测的值传入方程(5)中, 得出模型的更新比率  $K_k$ ; 将  $K_k$  值代入修正方程(6), (7)来计算  $k$  时刻的最优更新值. 更新目标位置, 并重复上述过程实现手部实时跟踪, 直到人走出 Kinect 的可视范围<sup>[9]</sup>.

### 3.3 手部动作识别

本研究主要系统设计了上、下、左、右挥手四个动作.

如图 8 所示, 若将手起始位置设为原点, 手的起始位置与终止位置之间的夹角在  $-\frac{\pi}{4} \sim \frac{\pi}{4}$ , 即终止位置在图中的 A 部分, 表示在执行“向右挥手”这个动作. 同理可得位置落在 B, C, D 部分表示执行“向左挥手”, “向上挥手”, “向下挥手”动作.

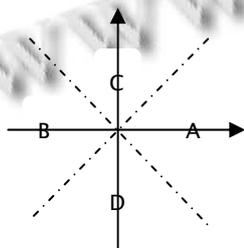


图 8 手势控制区域图

当人在执行“挥手”这个动作, 往往在开始和结束时手停留的时间稍长, 在挥手过程中停留时间较短. 因此, 可以设当手部在同一位置停留时间超过一定时间时, 该位置为手的起始位置. 针对终点位置确定, 与起始位置略有不同. 这是由于当人在做“挥手”动作时, 若手部收回速度过快, 以手部的实际位置为终点位置会使识别结果发生错误. 由于人“挥手”动作可以想象成手部向某一方做加速运动, 因此通过 Kalman 滤波器可以很好的预测手部终点的位置. 可以从确定起点位置时刻算起, 经过一段时间后的 Kalman 预测位置设为终点位置. 若手势的起始位置为  $P_1(x_1, y_1)$ , 结束位置为  $P_2(x_2, y_2)$ , 他们之间的夹角为  $\theta$ , 则  $\tan\theta = \frac{y_2 - y_1}{x_2 - x_1}$  [4,11].

由图 8 易得  $\theta$  与手势动作对应关系如表 1 所示.

表 1

$\tan(\theta + \frac{\pi}{4}) \geq 0$	$x_2 - x_1 \geq 0$	向右挥手
	$x_2 - x_1 < 0$	向左挥手

$\tan(\theta + \frac{\pi}{4}) < 0$	$y_2 - y_1 \geq 0$	向上挥手
	$y_2 - y_1 < 0$	向下挥手

因此, 通过对  $\tan(\theta + \frac{\pi}{4})$  与  $P_1, P_2$  值的判断, 识别出手势所表达的意思. 这样, 系统根据识别出手部的动作, 触发键盘 `keybd_event()` 或鼠标事件 `mouse_event()`, 进而实现对设备的控制.

系统通过使用 Kinect 提取深度信息, 避免了识别过程中光照、颜色的差异带来的影响. 同时使用 Kalman 滤波器对手势实时追踪, 使系统更准确的追踪定位手部位置. 通过对“挥手”动作的分析, 最后利用手部位置信息判断挥手角度, 进而识别出手部动作.

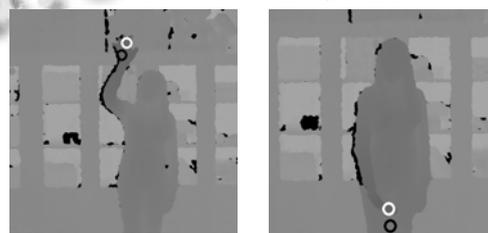
### 4 仿真实验结果与分析

对于 Kalman 滤波器模型, 由以上分析可以得出其状态向量维数  $dynam\_params=4$ , 测量向量维数  $measure\_params=2$ ; 由于 Kinect 每两帧之间间隔 0.03s, 因此设传递矩阵中的  $dt=0.03$ .

因为 Kinect 对手部的位置检测较敏感, 当实验者手部有轻微不可避免的位移, 系统还是对手部动作进行识别, 这并不是我们期望的. 因此实验中设置一个阈值, 若手部的运动在 0.5cm 之内, 认定手部位置并没有发生变化.

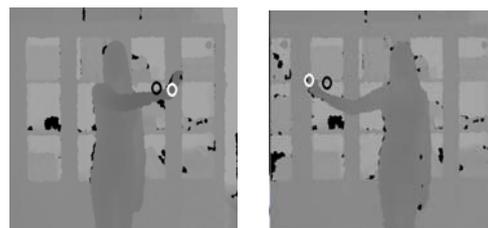
实验中将手部在同一位置超过 0.3s 时的实际位置设为起始位置; 从确定手部起点时刻算起, 将 0.5s 之后的 Kalman 滤波预测结果设置为手部终点位置.

Kalman 滤波器的手部追踪结果如图 9 所示. 图中, 白色圆环表示手部的实际位置, 黑色圆环表示预测位置. 为了使实验结果明显, 在实验过程中挥手



(a) 向上追踪

(b) 向下追踪



(c) 向左追踪

(d) 向右追踪

图 9 手部追踪----实验演示效果

手部的速度较快, 如果以正常速度挥手, 发现白色与黑色圆环基本重合, 跟踪效果较好. 动作识别效果如图 10 所示.

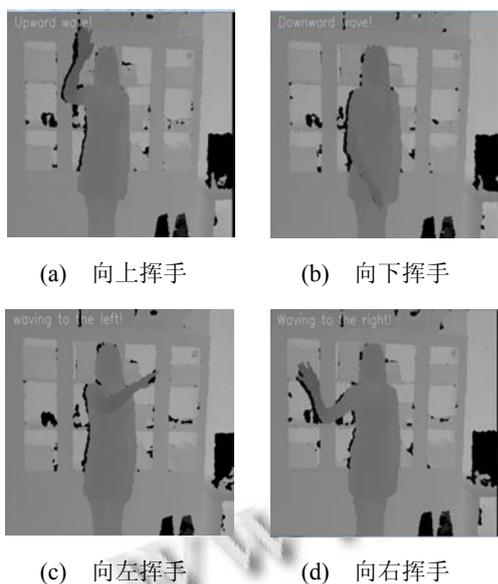


图 10 动作识别----实验演示效果

实验中对四种动作各执行 100 次, 识别结果如表 2 所示:

表 2 识别结果

方向	试验次数	识别个数	识别率
向上	100	96	97%
向下	100	95	96%
向左	100	97	98%
向右	100	96	97%

从表 2 看出, 当手部向上与向下, 向左与向右切换时的识别率能达到 98%. 试验结果出差错的地方主要出现在手部向直角方向转换. 这是由于手部在收回过快且向直角方向转换时, Kalman 滤波器还是会将会将终点动作预测为手部收回的方向, 因此造成误差.

### 5 结语

本文通过 Kalman 滤波器实现了手部追踪, 并准确定位出当前手部位置. 同时基于手部坐标分析运动方

向, 提出了手势识别算法, 能够正确识别出手部动作. 基于这种算法的手势识别可以在多种应用中使用, 如 PPT 的控制、多媒体的播放等, 同时, 也为今后对手语学习等的复杂手势识别的进一步研究奠定了很好的基础.

### 参考文献

- 1 体感技术. <http://baike.soso.com/lemma/CreateV1.e?sp=2&sp=l59933778>.
- 2 马建荣,章苏静,李凤.基于体感技术的亲子互动游戏设计与实现.中国电化教育,2012,308:85-86.
- 3 余涛.Kinect 应用开发实战:用最自然的方式与机器对话.北京:机械工业出版社,2013.
- 4 张毅,张烁,罗元,徐晓东.基于 Kinect 深度图像信息的手势轨迹识别及应用.计算机应用研究,2012,29(9):3547-3550.
- 5 Wang BL, Chen ZY, Chen J. Gesture recognition by using kinect skeleton tracking system. 2013 Fifth International Conference on Intelligent Human-Machine Systems and Cybernetics(IHMSC). 2013,10. 418-422.
- 6 Tang C, Ou YS, Jiang GL, Xie QQ, Xu YS. Hand Tracking and Pose Reconition via Depth and Color Information. 2012 IEEE International Conference on Robotics and Biomimetics, 2012,12. 1104-1109.
- 7 覃文军,杨金柱,王力,等.基于 Kalman 滤波器与肤色模型的手势跟踪方法.东北大学学报(自然科学版),2013,(34):474 - 477.
- 8 Asaari MSM, Suandi SA. Hand gesture tracking system using Adaptive Kalman Filter. Proc. of the 2010 10th International Conference on Intelligent Systems Design and Applications, 2010.10. 166-171.
- 9 于仕琪,刘瑞祯.学习 OpenCV.北京:清华大学出版社, 2009.
- 10 朱志玲,阮秋琦.结合尺度不变特征变换和 Kalman 滤波的 Mean Shift 视频运动目标跟踪.计算机应用,2013,33(11): 3179-3182,3243.
- 11 李平,李允俊.基于手势识别算法的鼠标终端.计算机系统应用,2013,22(8):83-87,71.