

基于 Kinect 深度图像信息的手势轨迹识别及应用^{*}

张毅^{1a}, 张烁^{1a}, 罗元^{1b}, 徐晓东²

(1. 重庆邮电大学 a. 自动化学院; b. 光电工程学院, 重庆 400065; 2. 电子科技大学 自动化学院, 成都 611731)

摘要: 为实现基于 Kinect 深度图像信息的手势轨迹识别, 提出了一种基于隐马尔可夫模型(HMM)的手势轨迹识别的方法。首先采用新型 Kinect 传感器获取图像深度信息; 然后通过 OpenNI 的手部分析模块获得手心的位置, 提取轨迹特征; 最后利用隐马尔可夫模型训练有效的轨迹样本并实现轨迹的识别。实验结果证明, 该方法能有效地识别手势轨迹, 并可用于控制智能轮椅的运动。

关键词: 手势轨迹识别; Kinect 传感器; OpenNI; 隐马尔可夫模型

中图分类号: TP311.52 **文献标志码:** A **文章编号:** 1001-3695(2012)09-3547-04

doi: 10.3969/j.issn.1001-3695.2012.09.092

Gesture track recognition based on Kinect depth image information and its applications

ZHANG Yi^{1a}, ZHANG Shuo^{1a}, LUO Yuan^{1b}, XU Xiao-dong²

(1. a. Institute of Automation, b. Institute of Optoelectronics Engineering, Chongqing University of Posts & Telecommunications, Chongqing 400065, China; 2. School of Automation, University of Electronic Science & Technology of China, Chengdu 611731, China)

Abstract: In order to realize the gesture track recognition based on the depth image information got by the Kinect sensor, this paper presented a new method. First, it used Kinect sensor to obtain depth image information. Then it obtained the position of palm through OpenNI hand analysis module, and extracted trajectory feature. Finally, it adopted Hidden Markov model (HMM) trained effective track sample and achieved track recognition. Experiments results show that the presented method can effectively identify the gesture track, and apply to control intelligent wheelchair.

Key words: gesture track recognition; Kinect sensor; OpenNI; HMM

近年来, 鉴于手势具有直观性、自然性的特点, 其已成为人机交互的一种重要手段^[1,2]。手势轨迹则是按照某种规则识别出手势表达的含义, 手势轨迹信息的应用有很强的可移植和可扩展性。传统的基于视觉手势轨迹识别研究包含了很多算法, 例如: Dinh 等人提出了采用 Harr 特征和 AdaBoost 算法识别 24 个基本英文手语^[3]; Zhang 等人^[4]提出了使用 HMM 算法来进行中文手语视频检索; Silanon 等人^[5]提出了根据手势运动分析识别泰文字母的方法。然而, 以上基于视觉的手势轨迹识别的前期分割方法易受光照、背景、摄像头特性等因素影响, 识别率不高。本文采用微软的 Kinect 传感器, 获取深度数据, 对深度图像进行手势分割和识别, 不受光照、背景等因素的影响, 提高了手势轨迹识别的稳定性和鲁棒性。

本文从 Kinect 传感器中获取深度数据, 通过 OpenNI 的手部分析模块获得手心位置, 对手心点进行去抖动, 标志开始点和结束点, 提取手势轨迹样本, 采用 HMM 模型对样本进行训练, 然后识别。本文设计了左转、右转、前进、后退、停止五个手势, 实现了与智能轮椅直观、自然的人机交互。

1 Kinect 传感器获取手部信息

1.1 Kinect 简介

Kinect 传感器可以同时获取 RGB 和深度图像数据, 支持实时的全身和骨骼跟踪, 并可以识别一系列的动作^[6,7]。图 1 为 Kinect 外观图。如图 1 所示, 左边镜头为红外线发射器, 中间的镜头是一般常见的 RGB 彩色摄像头, 右边镜头是红外线 CMOS 摄像头所构成的 3D 深度传感器。Kinect 具有追焦功能, 底座马达可驱动 Kinect 左右旋转 27°; 还有数组式麦克风。Kinect 可同时捕获彩色影像、3D 深度影像, 以及声音讯号。

Kinect 不同于普通摄像头的是, 它有感知世界的 CMOS 红外传感器。该传感器通过黑白光谱的方式来感知环境: 纯黑代表无穷远, 纯白代表无穷近, 黑白间的灰色地带对应物体到传感器的物理距离。它收集视野范围内的每一点, 并形成一幅代表周围环境的景深图像。传感器以每秒 30 帧的速度生成深度图像流, 实时 3D 地再现周围环境。

1.2 获取手势轨迹信息

Kinect 采用的分割策略是将人手从背景环境中区分出来。

收稿日期: 2012-01-12; **修回日期:** 2012-04-10 **基金项目:** 国家科技部国际合作项目(2010DFA12160); 重庆市科技攻关资助项目(STC, 2010AA2055); 国家自然科学基金资助项目(51075420, 60905066); 重庆市教委科学技术研究项目(KJ100516)

作者简介: 张毅(1966-), 男, 重庆人, 教授, 博导, 主要研究方向为机器人及应用、数据融合、信息无障碍技术(zhangyi99@263.net); 张烁(1986-), 女, 河南商丘人, 硕士研究生, 主要研究方向为图像传感与处理; 罗元(1972-), 女, 湖北人, 教授, 博士, 主要研究方向为机器视觉、智能信号处理、数字图像处理; 徐晓东(1981-), 男, 山东潍坊人, 博士研究生, 主要研究方向为机器人视觉、人—机器人交互研究。

OpenNI 提供了 API 接口,即提供一个多语言、跨平台的框架,该框架具有与外接硬件沟通的能力。此外,OpenNI 还包括中间层空间,用来做手势辨识或是追踪之类的处理。目前 OpenNI 支持的硬件包含 3D Sensor、RGB Camera、IR Camera、Audio Device 这四类。OpenNI 中间层部分定义了下面三种中间件:a)全身分析,由感应器获取图像信息,产生身体的相关资讯,如关节、相对位置与角度、质心等;b)手部分析,追踪手的位置;c)场景分析,分析场景内的资讯,如分离前景和背景、地板的座标轴、辨识场景内的不同物体。图2为通过 OpenNI 手部分析与视觉开发包(OpenCV)相结合的方法检测并标示出的手心位置。

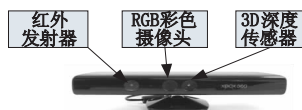


图1 Kinect外观



图2 深度图像和彩色图像对比

手势检测要通过辨识预先定义好的手势(如挥手),实验中,保持手在身体的前方,展开手掌,指尖朝上,并且正对 Kinect 传感器,手与传感器之间的距离在 20 cm ~ 1 m。挥手的速度不能过快或过慢且有一定的运动幅度,直到检测到手。此时,产生一个布尔信号给程序,程序开始追踪手位置。本文对检测到的手心点进行了去抖动处理,保存前一帧判断所得手心的圆圈(center, radius)数据,再和当前帧的圆圈数据比较。本文通过实验测试设定阈值为 2.4 mm,如果偏差在阈值范围内,那么就继续显示前一帧的圆圈,偏差在阈值范围以外,重新绘制手心圆圈的位置,这样就可以避免在小幅度内的抖动。图3为“向上”手势轨迹分析图。设置图像左下角为起点,在图像上画轨迹的过程中,会出现一些无意义的手势帧序列或者无效帧,尤其是在手势刚开始输入及手势结束时,中间帧基本都是有效帧^[8]。无效帧对于手势特征的提取是无益的,还增加了多余的运算量,降低了系统的识别率。本文提出了新的轨迹起始点和结束点的判断。通过实验发现,当有手势流输入时,系统检测到手心后,开始跟踪手心点,当检测到在很小范围内波动的连续 5 帧,则认为是手心处在静止点,可以作为手势轨迹输入的开始,第二次手心静止点则认为手势轨迹输入结束。

2 基于隐马尔可夫模型(HMM)的手势轨迹识别

2.1 HMM 模型定义

隐马尔可夫模型是一种很成熟的匹配时变数据的技术,它是随机状态机器。HMM 的使用涉及到训练和分类两个阶段。在手势识别系统中,观察矢量序列和编码符号序列,统称观察序列,记为 O ,它是一种随机序列。一个有 N 个状态(记为 $S_1, S_2, \dots, S_N$)的 HMM 是由三元组参数 $\lambda = \{\pi, A, B\}$ 表示的,其中:

a) $\pi = [\pi_1, \pi_2, \dots, \pi_N]$ 为初始分布用于描述观察序列 O 在 T_i 时刻所处状态的概率分布, $\pi_i = P(q_i = S_i), i = 1, 2, \dots, N$, 它满足 $\sum_{i=1}^N \pi_i = 1$ 。

b) $A = \{a_{ij} | i, j = 1, 2, \dots, N\}$ 为状态转移概率矩阵,即 $a_{ij} = P(q_i = S_j | q_{i-1} = S_i, q_{i-2} = S_k, \dots) = p(q_i = S_j | q_{i-1} = S_i)$, 它满足 $\sum_{j=1}^N a_{ij} = 1$ 。

c) B 为状态输出概率矩阵,它是随机变量或随机矢量在各

状态的观察概率空间中的分布, $B = \{b_j(k), j = 1, 2, \dots, N, k = 1, 2, \dots, M\}$, 它满足 $\sum_{k=1}^M b_j(k) = 1$ 。

2.2 动态手势轨迹特征选择和提取

动态手势不同于静态手势,因为静态手势仅仅是空间位置和形态上的变化,而动态手势同时涉及到时间及空间上的变化。实验表明,人的运动可以只通过运动信息来识别^[9],因此手势可以转换为空间的运动轨迹来识别。

手势轨迹的基本特征是位置、速度和角度^[10]。对于速度而言,同一个手势差别也很大,因此不实用。手势轨迹曲线上点的坐标位置可以作为手势轨迹识别的显著特征,但如前所述,即使是相同的手势轨迹也会有出入,坐标点自然也会有变化。考虑使用手势轨迹不同时刻的切线角度变化作为轨迹特征。由 Kinect 传感器可获取 t 时刻和 $t-1$ 时刻的手心位置 $(x_c^t, y_c^t)(x_c^{t-1}, y_c^{t-1})$, 通过式(1)可得出手势轨迹的角度 φ^t , 如图 4(a) 所示。

$$\varphi^t = \tan^{-1} \left(\frac{y_c^t - y_c^{t-1}}{x_c^t - x_c^{t-1}} \right) \quad (1)$$

根据所得角度,本文采用 12 方向链码进行离散化,如图 4(b) 所示。



图3 “向上”手势轨迹点分析

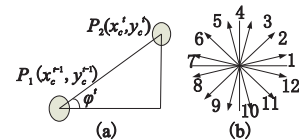


图4 12方向链码提取特征向量

2.3 HMM 训练与识别

用于解决训练问题的基本方法是 Baum-Welch 算法(也称前、后向算法),它是一种基于期望调节(expectation modification)概念的算法。

定义一个变量 $\xi_t(i, j)$, 对应于观察序列 $O_1, O_2, \dots, O_T$, 假设在 t 时刻的状态是 $S_i, t+1$ 时刻的状态是 S_j 的概率。

$$\begin{aligned} \xi_t(i, j) &= P(q_t = S_i, q_{t+1} = S_j | o_1 o_2 \dots o_T) = \\ &= \frac{P(q_t = S_i, q_{t+1} = S_j, o_1 o_2 \dots o_T)}{P(o_1 o_2 \dots o_T)} = \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_i \sum_j \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)} \end{aligned} \quad (2)$$

定义一个变量 $\gamma_t(i)$, 对应于观察序列 $O_1, O_2, \dots, O_T$, 假设在 t 时刻的状态是 S_i 的概率。

$$\begin{aligned} \gamma_t(i) &= P(q_t = S_i | o_1 o_2 \dots o_T) = \\ &= \frac{P(q_t = S_i, o_1 o_2 \dots o_T)}{P(o_1 o_2 \dots o_T)} = \\ &= \frac{\alpha_t(i) \beta_t(i)}{\sum_i \alpha_t(i) \beta_t(i)} \end{aligned} \quad (3)$$

由式(2)(3)可以给出重估参数为

$$\begin{aligned} \bar{a}_{ij} &= P(S_j | S_i) = \frac{\sum_t \xi_t(i, j)}{\sum_t \gamma_t(i)} \\ \bar{b}_i(k) &= P(v_k | S_i) = \frac{\sum_{t, o_t = v_k} \gamma_t(i)}{\sum_t \gamma_t(i)} \\ \bar{\pi}_i &= P(S_i) = \gamma_1(i) \end{aligned} \quad (4)$$

其中: $\bar{a}_{ij}, \bar{b}_i(k), \bar{\pi}_i$ 分别表示经过参数重估后的 HMM 参数, 即 π, A, B 三个参数。这就是著名的 Baum-Welch 算法。可以证明, 经过这样的重估算法后, 有结果 $P(O | \bar{\lambda}) > P(O | \lambda^2)$ 产生。

重复迭代,前一次输出的概率与本次输出的概率之差的绝对值相对于这两次输出概率绝对值的平均值比例小于 5×10^{-4} 时停止迭代,从而获得最大的输出概率,完成了训练的目的。

基于 HMM 模型的识别过程一般通过比较已经训练好的参数模型集和测试序列的似然度来完成,因此这涉及到了最大概率问题的计算。由于 Viterbi 算法能够在已经给定的模型下最佳地解释观察序列,而且计算量比前后向概率计算算法要少得多,因此本文选用 Viterbi 算法计算似然度进行识别。Viterbi 算法描述如下:

a) 初始化。

$$\sigma_1(i) = \max P(q_1 = s_i, | o_1) = \pi_i b_i(o_1) \quad 1 \leq i \leq N, W_i = 0 \quad (5)$$

b) 前向递归。

$$\begin{aligned} \sigma_t(j) &= \max P(q_1 \cdots q_{t-1}, q_t = s_j, o_1 o_2 \cdots o_t) = \\ &= \max_i [a_{ij} b_j(o_t) \max P(q_1 \cdots q_{t-1} = s_i, o_1 o_2 \cdots o_{t-1})] = \\ &= \max_i [a_{ij} b_j(o_t) \sigma_{t-1}(i)] \quad 1 \leq i \leq N, 2 \leq t \leq T, 1 \leq j \leq N \\ W_t(j) &= \arg \max [\sigma_t(i) a_{ij}] \quad 1 \leq i \leq N, 2 \leq t \leq T, 1 \leq j \leq N \end{aligned} \quad (6)$$

此处是对每个 i 计算每个 t , 记录该 $\sigma_t(j)$ 的最大值,也是前一步的 q_{t-1} 来源,以便最后回溯。

c) 结果。

$$\begin{aligned} P^* &= \max_{i=1-N} [\sigma_T(i)] \\ q_T^* &= \arg \max_{i=1-N} [\sigma_T(i)] \end{aligned} \quad (7)$$

由 q_{t+1} 反查记录 W , 得到 $q(t=T-1, T-2, \cdots, 1)$, 条件概率最大的轨迹模式类作为该轨迹序列的类别,即为识别结果,此时得到的路径便是最优路径。手势轨迹识别流程如图 5 所示。

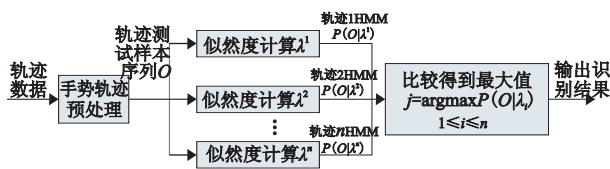


图5 手势识别的系统流程

3 实验结果分析

本文定义了五种不同的手势轨迹来完成智能轮椅的基本控制。如图 6 所示,手向左运动代表左转,向右运动代表右转,向上运动代表前进,向下运动代表后退,画圈代表停止。



图6 设计五种手势控制轮椅

3.1 基于 HMM 手势识别实验

本文采用从左到右的 HMM(状态是有序的序列,并左右转移,或连续几帧停留在一个状态,如图 7 所示)作为模型依据。在 HMM 建模阶段,每条有效轨迹对应一个 HMM。训练中

模型状态数的选取,理论上是越多越好。因为随着状态数的增加,训练的模型越能真实、准确地描述轨迹,但这也会使计算量大大增加,影响实时性的要求。状态数的选择与轨迹的复杂程度有关,根据轨迹的复杂程度选择状态数为 4~6 个不等。图 8 为不同状态数的识别率。从图 8 中可以看出,状态数为 4 时识别率较高,满足实时性要求。

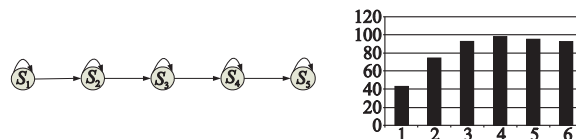


图7 从左到右无跳转HMM模型

图8 不同状态数的识别率

利用本文方法,分别采集 10 个人手势轨迹,每个人每种手势做 2 次实验,将其中 50 个手势轨迹作为训练样本,另外 50 个作为测试样本。表 1 给出了五种手势轨迹的识别率,可以看出,本文方法具有很高的识别率,满足应用的需求。

表1 五种手势的识别率

手势轨迹示意	向左←	向右→	向上↑	向下↓	圆圈○
准确次数	49	48	50	49	48
识别率/%	98	96	100	98	96

3.2 鲁棒性验证

为验证本系统在不同光照环境下的鲁棒性,在不同光照环境下每个指令分别进行了 50 次实验,表 2 为在分别为日光灯光照环境和黑暗环境下每个指令分别做 50 次实验中正确识别的次数及其准确率。图 9 是在不同光照及复杂背景条件下的对比。实验证明,本文设计的手势识别系统对光照变化及复杂背景干扰具有较强的鲁棒性。

表2 不同光照环境下每个指令的识别率

指令	日光灯环境(a)		黑暗环境(b)	
	正确次数/次	准确率/%	正确次数/次	准确率/%
前进	49	98	48	96
后退	49	98	48	96
左转	48	96	47	94
右转	49	98	47	94
停止	48	96	48	96



(a) 日光灯环境下的手势识别效果



(b) 黑暗环境下的手势识别效果



(c) 复杂背景下的手势识别效果

图9 不同光照及复杂背景条件下的对比

3.3 基于手势轨迹识别的智能轮椅控制系统设计

系统的主要硬件包括 Kinect 传感器一个、电脑一台、无线网卡一个、智能轮椅本体一台。系统采用的软件编程环境是 VS2008, 首先安装 OpenNI, 配置 OpenCV, 可以使 Kinect 传感器在 OpenNI 架构下被 PC 机识别。

在实验过程中,通过 Ad hoc 网络,采用无线网卡在智能轮椅和电脑之间进行无线通信;将 Kinect 传感器作为摄像头采集

图像,在 VS2008 编程环境下处理传感器得到的图像数据,根据定义好的手势命令可以得到相应的控制命令;通过 Socket 通信,上位机将控制命令发送给 ARM 控制器,上位机系统通过 RS232 串行总线与基于 DSP 的驱动模块和传感器模块通信;智能轮椅的驱动由 ARM9 上位机、底层驱动模块和传感器模块完成。基于手势轨迹的智能轮椅控制系统的实现过程如图 10 所示。图 11 为操作界面。用户将手放在正对 Kinect 传感器的位置时, Kinect 实时采集图像,获取深度图像和彩色图像,并通过 OpenNI 算法检测手心,得到手势轨迹,通过 HMM 中 Viterbi 算法得到的识别结果判断手势是否正确,如果正确就控制轮椅运动,如不正确继续执行上一个运动控制指令,以此实现对智能轮椅运动的控制。

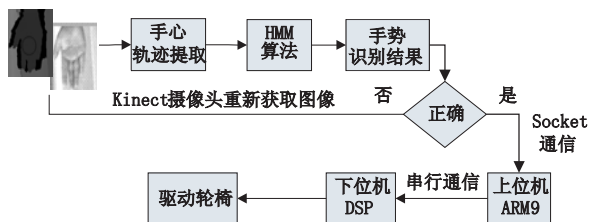


图10 系统流程



图11 操作界面

在控制轮椅实验中,本文设计了一条轮椅运动路线,具体路线如图 12(a)所示,整个实验环境长 6 m、宽 5 m,轮椅从 A 点运动到 B 点,两个长方形代表障碍物,中间的箭头代表理想的轮椅运动路线和方向;(b)采集了五条实验轨迹,五条轨迹基本一致,实验证明,本方法能够有效地控制轮椅的运动。

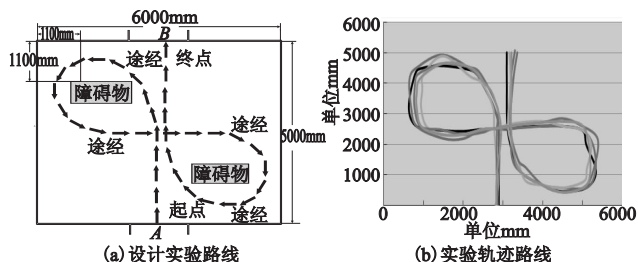


图12 实验路线

4 结束语

本文通过深度信息有效地从实时的视频流中分割手势并检测手心点,可以有效地避免光照和复杂背景的干扰,提取准确的轨迹特征,引入 HMM 模型,对手势的运动轨迹进行训练和识别,并通过无线网络实时地控制智能轮椅。实验结果表明,该方法具有较高的识别率。为完善本系统,今后的工作将加入手形的变化,改进手势起始点的确定方法,并且进一步提高系统的鲁棒性,增强系统的推广性。

参考文献:

- [1] ZHANG Jian-jie, LIN Hao, ZHAO Ming-guo. A fast algorithm for hand gesture recognition using relief[C]//Proc of the 6th International Conference on Fuzzy Systems and Knowledge Discovery. 2009: 8-12.
- [2] KAO M C, LI T H S. Design and implementation of interaction system between humanoid robot and human hand gesture[C]//Proc of SICE Annual Conference. [S. l.]: IEEE Press, 2010: 1616-1621.
- [3] CHEN Qing, GEORGANAS N D, PETRIU E M. Real-time vision-based hand gesture recognition using Haar-like features[C]//Proc of IEEE Instrumentation and Measurement Technology Conference. [S. l.]: IEEE Press, 2007: 1-6.
- [4] ZHANG Shi-lin, ZHANG Bo. Using HMM to sign language video retrieval[C]//Proc of the 2nd International Conference on Computational Intelligence and Natural Computing. 2010: 55-59.
- [5] SILANON K, SUVONVORN N. Hand motion analysis for Thai alphabet recognition using HMM[J]. International Journal of Information and Electronics Engineering, 2011, 1(1): 65-71.
- [6] IKEMURA S, FUJIYOSHI H. Real-time human detection using relational depth similarity features[C]//Proc of the 10th Asian Conference on Computer Vision. 2011: 25-38.
- [7] URTASUN R, FUA P. 3D human body tracking using deterministic temporal motion models[C]//Proc of the 10th Asian Conference on Computer Vision. 2004: 92-106.
- [8] 杨青. 手势识别就是研究[D]. 大连: 大连理工大学, 2010.
- [9] KAHOL K, TRIPATHI P, PANCHANATHAN S, et al. Gesture segmentation in complex motion sequences[C]//Proc of IEEE International Conference on Image Processing. 2003: 105-108.
- [10] YOON H S, SOH J, BAE Y J, et al. Hand gesture recognition using combined features of location, angle and velocity[J]. Pattern Recognition, 2011, 34(7): 1491-1501.