

基于 WebRTC 的残疾人鼠标研究与实现

陈 晨, 吴建国

(安徽大学 计算机科学与技术学院, 安徽 合肥 230039)

摘 要:文中基于手势识别和 WebRTC 框架提出了残疾人鼠标的设想,对整个系统模块进行了研究和详细设计。残疾人鼠标系统基于 B/S 架构,将图像获取、命令执行和图像匹配分离,利用 Firefox 进行图像获取和鼠标操作的模拟,利用 JavaCV 和 Jetty 实现服务器端的图像匹配和命令反馈。文中使用形状比例和转角函数等进行特征抽取,使用背景差分 and 欧式距离进行图像匹配,在快速响应的同时保证匹配的正确率。B/S 架构使得系统可以构建于云环境并具有良好的扩展性、伸缩性和可定制性。实验表明,系统已经达到初期的设计目的,并具有可接受的延迟,符合鼠标操作对于实时性的要求。

关键词:图像匹配;手势识别;残疾人鼠标;WebRTC;Firefox

中图分类号:TP317.4

文献标识码:A

文章编号:1673-629X(2013)09-0032-04

doi:10.3969/j.issn.1673-629X.2013.09.008

Research and Implementation of Disabled Mouse Based on WebRTC

CHEN Chen, WU Jian-guo

(College of Computer Science & Technology, Anhui University, Hefei 230039, China)

Abstract: It has proposed the idea of disabled mouse based on gesture recognition and WebRTC framework, and has done study and detailed design of the entire system modules. Disabled mouse system is based on B/S structure, separates image acquisition, command execution and image matching, using Firefox for image acquisition and mouse operation simulation, using JavaCV and Jetty to implement server-side image matching and command feedback. Shape proportion and corner function is used for feature extraction, and image matching is conducted by background difference and Euclidean distance to ensure the correct rate of the match with rapid response at the same time. B/S architecture allows the system to be built on the cloud environment and has good scalability, flexibility and customizability. The experiment shows that the system has achieved initial design purpose and acceptable delay, meeting the requirements the mouse operation for the real-time requirements.

Key words: image matching; gesture recognition; disabled mouse; WebRTC; Firefox

0 引 言

残疾人因为手部缺陷而无法操纵鼠标,近年来已有很多适用于残疾人的人机交互方式提出,例如:脚控鼠标、视线控制鼠标、语音输入鼠标等。这些方法技术复杂、价格昂贵,手势利用手的各部位组合成特定的形状或运动轨迹来表达某种特定含义,有自然直观、不需要中间媒体的特点,十分适合残疾人人机交互^[1]。手势识别可以基于数据手套^[2],也可以基于计算机视觉^[3]。基于视觉的手势识别系统一般分为视频采集和手势图像提取、手势图像预处理、手势分割、手势分析、手势建模、手势识别数个步骤。手势识别按识别对象

划分可分为识别手的姿势和形状的静态手势识别^[4-5]和识别一组连续的手型变化或者手势运动轨迹的动态手势识别^[6]。

文中提出了一种基于静态手势识别的残疾人鼠标解决方案:将可控物体指代为残疾人可以控制的健全的身体部位,包括脚、手臂等能够移动或形态可发生变化的。鼠标操作类型不多,残疾人完全可以利用可控物体模拟鼠标操作计算机。已有的算法或系统一般基于本地环境。文中将构建 B/S 架构的系统,即将手势分割、手势分析、手势建模、手势识别等运算密集的任务放到服务器端进行,客户端的浏览器仅负责图像采集、

收稿日期:2012-11-06

修回日期:2013-02-23

网络出版时间:2013-04-22

基金项目:安徽省教育科技重点项目(KJ2009A57)

作者简介:陈 晨(1988-),女,硕士研究生,研究方向为数字图像处理、手势识别;吴建国,教授,研究方向为数字系统、中文信息处理、智能识别。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20130422.1729.060.html>

提交和模拟鼠标事件。工作过程:浏览器向服务器提交图像数据,服务器调用其上图像匹配模块进行手势识别,并将结果反馈给浏览器,浏览器根据反馈操纵鼠标。

B/S 架构有以下优势:

- (1)符合当前的云计算趋势;
- (2)客户端只负责图像传输和处理结果的接受执行,图像处理在服务器端进行,今后软硬件提升后的升级、重构,硬件、算法、逻辑的改变都只在服务器端进行,利于组建具有伸缩性的系统;
- (3)出于个性化服务的需要,为不同用户保留各自模板库,甚至可以在不同应用环境下动态切换算法;
- (4)移动应用的需求,人们可能使用不同种类的终端设备,一个公共场合的终端设备也可能被很多用户使用,如果模板库只保留在某个固定的本地机器上,将会非常不便。

1 总体模块设计

残疾人智能鼠标由简单、廉价的图像采集装置和手势识别系统组成,其中图像采集装置包括摄像头、照明灯,摄像头垂直设置于残疾人可控物体正上方,采集残疾人可控物体在平面上投影。

系统基于 WebRTC 技术,分为浏览器、WebSocket/HTTP 服务器、后端图像处理 3 个模块,如图 1。浏览器使用 Firefox,WebSocket/HTTP 服务器使用 Jetty。由于 Jetty 由 Java 编写,后端图像处理采用 JavaCV,它是 OpenCV 的 Java 包装。

整个过程分为两个阶段,如图 1 所示。

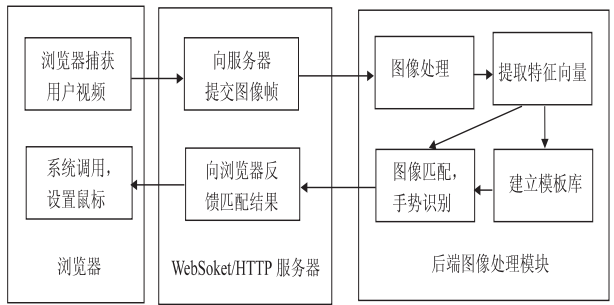


图 1 总体模块设计

前一阶段是定制。服务器得到浏览器提交的可控物体图像后,利用图像分割得到目标物,再提取有用特征,设定特征向量同鼠标动作的映射关系后,保存到形状模板库中;

后一阶段是匹配。服务器再次将浏览器端提交图像转换到特征空间后,将当前图像的特征向量与形状模板库进行匹配,找到最接近类型,完成识别和分类,反馈匹配结果给浏览器端。浏览器端根据反馈结果调用系统函数,模拟鼠标事件。

2 相关技术解释

系统将使用以下技术:

- 1) WebRTC: WebRTC 既是一系列基于 Web 浏览器的实时音频视频通信技术的简称^[7],同时也是 Google 的一项开源框架的名称,它使浏览器具有实时捕获、传输媒体数据的能力;
- 2) WebSocket: 提供浏览器与服务器间建立持久连接,进行全双工通讯的能力^[8]。它改变了传统 B/S 架构请求反馈的信息交互模式,使得浏览器和服务器间可以实时交换图像数据和鼠标指令;
- 3) Firefox 插件: 操纵鼠标是一种系统调用,浏览器内置的 JavaScript 引擎出于安全考虑,未提供此类接口。编写了含有 Win32 鼠标系统函数的 DLL 动态链接库,运用 Firefox 插件开发中的 XPCOM 和 xpconnet 技术^[9],使 JavaScript 代码可通过 Firefox 操纵鼠标;
- 4) JavaCV: JavaCV 是对 OpenCV 等常用计算机视觉库的封装,后端图像处理模块即由其实现并放置在服务器端。

Firefox 的 Alder 分支提供了对 WebRTC 和 WebSocket 客户端的 JavaScript API 支持,Jetty 则提供了 WebSocket 服务器端的支持。Jetty 搭建 HTTP/WebSocket 服务器,客户端的浏览器加载服务器的特定网页,打开摄像头实时采集残疾人可控物体图像并通过 WebSocket 循环提交图像帧到服务器,同时监听服务器反馈的匹配结果通过浏览器插件操纵鼠标。具体地说:浏览器会定时地获取视频流某一时刻的快照,将其绘制到页面的 canvas 元素上,再将其包装成 JavaScript 中的 DataURL。DataURL 是一种 Base64 编码字符串,可通过 WebSocket 以文本形式提交到服务器,服务器再解码成二进制数据。经处理后将匹配结果通过 WebSocket 发送回浏览器端,若是鼠标点击等操作,则发送一个命令字符串,若是鼠标移动定位,则发送可控物体在图像帧中的相对位置,浏览器会将其转换到实际屏幕分辨率对应的像素点。反馈触发 Firefox 插件中操纵鼠标函数,从而达到残疾人利用可控物体模拟鼠标的初衷。

3 图像处理模块详细说明

3.1 图像预处理:降噪、分割、归一化

Base64 编码字符串解码后即得到图像的二进制数据,服务器会对这些图像帧进行图像分割以得到残疾人可控物体投影。图像分割常用方法包括光流法、帧间差分法、背景差分法,其中背景差分建模简单、对背景变化适应快,适合此处实时性要求。背景差分法是由输入图像与背景进行减运算得到输入图像与背景不同的区域,所得区域即为需要检测的可控物体区

域^[10]。

图像预处理步骤为:先将当前图像帧转换为灰度图,设定阈值,构造出图像帧的二值图像,再使用形态学先开后闭操作处理图像^[11],去除噪点,获得可控物体的二值图像。由于物体与摄像头距离可能造成同一物体的多次二值投影大小不等,需对其进行归一化:取得可控物体的外围正方形,并将此正方形归一化到统一边长。

3.2 特征提取

提取特征以同多种鼠标操作形成映射关系。这些特征应具有平移和尺度不变性的原则,从每幅图像帧中提取以下特征:

1) 外围矩形宽高比和形状比例:外围矩形就是由图像中目标区域最外围四条水平或竖直切线(上、下、左、右四个方向)所组成的矩形区域。用像素点个数表示目标物外围矩形的高 rect_height 和宽 rect_width, 外围矩形宽高比计算公式为:

$$\text{Whr} = \frac{\text{rect_width}}{\text{rect_height}}$$

2) 形状比例:形状比例计算周长和面积的关系,在一定程度上描述了区域的紧凑性,几何上相似的两个形状将会有相近的比例值。令目标物的周长(perimeter)为目标轮廓图像在目标矩形区域内包含的像素总数,目标物的面积(area)为目标物二值图像在目标矩形区域内包含的像素总数,形状比例计算公式为:

$$\text{Sr} = \frac{\text{perimeter}^2}{4\pi \text{area}}$$

3) 转角函数:转角函数是指图形关于边长的一个函数,标识了从图形边界上的一个初始点开始到当前点经过一点边长所累积的转角^[12]。当前点从初始点开始沿逆时针方向遍历整个边界,转角由初始方向与当前点的切线方向夹角确定,切线方向取逆时针方向。为简化问题,加入以下限制:初始点的转角函数值为零;线段上不存在勾状;两条线段的拐角处不形成圆圈;转角累加不超过 2π ,超过则取模。

这样修正后的转角函数如下:

$$V_i = \left(\Theta\left(\frac{i}{d}\right) - \Theta\left(\frac{1}{d}\right) \right) \bmod 2\pi, 0 \leq i \leq d$$

其中 d 为需要抽取特征的维度,即将图形的边界 d 等分,在每个等分点上计算从初始点到达此位置时累积的转角。一个例子见图 2。

4) 其他特征取值:还采用像素值为 1 的个数和数条切线上像素值由 0 变 1 的个数作为剩余的特征向量。整个特征向量如

{外围矩形宽高比 Whr, 形状比例 Sr, 转角函数值 $V_1 \cdots V_m$, 像素值为 1 的个数 C , 数条切线上像素值由 0

变 1 的个数 $P_1 \cdots P_N$ }

若是定制阶段,则到此结束,识别则继续进行下面的步骤。

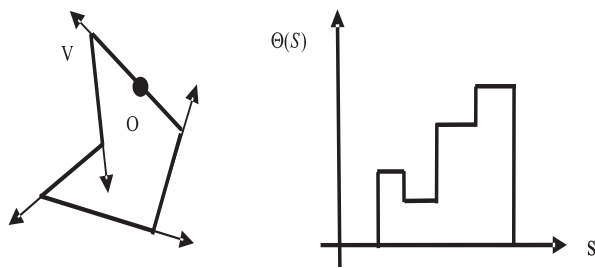


图 2 转角函数图形,横轴表示占边长比例

3.3 分类识别

模板匹配法核心思想是通过测量原始数据与预先存储的模板之间的相似度来进行匹配,此处采用欧氏距离进行匹配分类:

$$D = \sum_{i=0}^n \beta_i (\varphi_i - F_i)^2$$

其中 D 为欧氏距离, φ_i 为待识别的目标物特征向量中的特征分量, F_i 为模板库中相应的特征分量, β_i 为分量调节系数。运用调节系数可以将各个特征分量调节到一个数量级上。

此处需要考虑 2 个问题:

1) 假设指定 A 图像匹配鼠标定位移动, B 图像匹配鼠标点击,浏览器发送图像帧的间隔很短,会多次发送 B 图像,如果直接根据 B 图像匹配到鼠标点击的话,可能造成鼠标在某处的多次点击,而事实上只需要一次。采取的方法是:记录前几帧图像的匹配结果,如果此帧是 B,查看前几帧匹配的结果中是否存在 A 不存在 B,是则说明此帧转换到了 B,向浏览器反馈鼠标点击的命令;否则前几帧中存在 B,说明是同一图像采集的多次发生,不进行任何操作。对于 A,则无需关心前几帧的匹配结果;

2) 设定阈值,当欧式距离最小的匹配结果依然大于此阈值时,认为此图像无法匹配到任何模板,无法分辨,拒识。

4 服务器端手势识别算法

服务器端手势识别算法循环执行以下流程:

输入:某一时刻浏览器提交的图像帧 f , 即一幅图像;

输出:反馈给浏览器的鼠标操作命令。

STEP 1: 原始彩色图像转换为灰度图像,设定阈值,灰度图像转化为二值图像;

STEP 2: 背景差分法分割出可控物体形状;

STEP 3: 形态学先开后闭处理,去除噪声;

STEP 4: 计算可控物体投影质心位置并记录;

STEP 5:归一化,取得可控物体的外围正方形,并将此正方形归一化到统一边长,方法如下:

1)取可控物体横纵方向上切线的最长距离,设为 length;

2)将可控物体放置在 length * length 的正方形中,则投影在某个方向上长度达到 length,在另一个方向上的中间位置可控物体到正方形边的距离相等,此正方形即为外围正方形;

3)缩放此正方形至边长为统一的 global_length;

4)将外围正方形放回原图像大小的黑色背景中,外围正方形质心同 STEP 4 中记录的可控物体质心重合。

STEP 6:特征提取,将二值图像映射到特征向量;

STEP 7:匹配,计算同各模板的欧式距离,取欧式距离最小者;若此值大于设定阈值,拒识,转下次循环,否则作为匹配结果并记录;

STEP 8:
if 匹配结果是鼠标移动定位操作:

根据 STEP 4 中记录的可控物体质心位置计算可控物体在图像帧中的相对位置,将此相对位置发送回浏览器;

else if 匹配结果是鼠标点击等操作:

比较前几帧的匹配结果,若前几帧中存在鼠标定位不存在鼠标点击,则将命令字符串发送回浏览器;若前几帧中存在鼠标点击,则不做任何操作。

5 实验

浏览器使用 Firefox 的 Alder 分支,在 Windows 环境下采用 VS 2010 和 C++编写 DLL 动态链接库;使用 Jetty 搭建 HTTP/WebSocket 服务器,JavaCV 实现后端图像处理模块。使用两台连接在互联网上的个人电脑进行试验,浏览器端运行在 Windows 7 系统、2G 内存的环境中,服务器端运行在 Windows 7 系统、4G 内存的环境中。

设定 4 种手势分别对应移动、单击、双击和右击,每个手势均重复 100 次,不记录其中可能夹杂的其他手势。记录浏览器端响应的鼠标动作,同意图的鼠标动作比较以计算正确率。实验结果如表 1。

在程序中记录向服务器提交一幅图像帧到接收到服务器返回匹配命令的时间间隔,计算得到 400 次不同操作的平均延迟为 1.142 s。

实验表明,系统具有了初步的可用性,可以为残疾人提供具有一定实时性的简单的鼠标操作模拟。同时由于双击、右击的可控物体投影形态较为接近,且特征

维数较低、匹配算法较简单,说明准确率较易受到模板之间差异大小的影响,出现少量的拒识、误判的现象。

| 表 1 不同鼠标操作的识别准确率 | | | | | |
|------------------|------|------|------|------|-------|
| 鼠标操作 | 测试次数 | 正确识别 | 错误识别 | 拒绝识别 | 识别率/% |
| 移动 | 100 | 100 | 0 | 0 | 100 |
| 单击 | 100 | 95 | 1 | 4 | 95 |
| 双击 | 100 | 86 | 8 | 5 | 86 |
| 右击 | 100 | 85 | 7 | 8 | 85 |

6 结束语

欧式距离的简单匹配算法使得服务器在运算能力较差的情况下依然可以获得较好的实时性能。当服务器运算能力提高时,可以考虑更准确复杂的手势特征提取和图像匹配算法以提高手势识别的正确率。这些改变对浏览器端是透明的,体现了文中架构设计的合理之处,也将是进一步的研究目标。

参考文献:

[1] Du Youtian, Chen Feng, Xu Wenli, et al. A survey on the vision-based human motion recognition[J]. Acta Electronica Sinica, 2007, 35(1): 84-90.

[2] 张睿, 张锡恩. 虚拟场景中基于数据手套的交互方法[J]. 计算机工程, 2005, 31(12): 223-225.

[3] 任海兵, 祝远新, 徐光佑, 等. 基于视觉手势识别的研究-综述[J]. 电子学报, 2000, 28(2): 118-121.

[4] 章丰明, 任彧. 基于 MEB-SVM 的静态手势识别研究[J]. 机电工程, 2010, 27(6): 120-123.

[5] 谭昶, 肖南峰. 基于改进 RCE 和 RBF 神经网络的静态手势识别[J]. 计算机工程与应用, 2011, 47(7): 172-176.

[6] 孙丽娟, 张立材, 郭彩龙. 基于视觉的手势识别技术[J]. 计算机技术与发展, 2008, 18(10): 214-216.

[7] Web Real-time Communications Working Group. WebRTC 1.0: Real-time Communication Between Browsers[DB/OL]. 2012. <http://dev.w3.org/2011/webrtc/editor/webrtc.html>.

[8] HyBi Working Group. The WebSocket API[DB/OL]. 2012. <http://dev.w3.org/html5/websockets/>.

[9] Mozilla developer network. Creating custom Firefox extensions with the Mozilla build system[DB/OL]. 2012. <https://developer.mozilla.org/en-US/docs/XPCOM>.

[10] 张毅刚, 曹阳, 项学志. 静态背景差分运动目标检测研究[J]. 电子测量与仪器学报, 2010, 24(5): 494-499.

[11] 马义德, 张红娟. PCNN 与灰度形态学相结合的图像去噪方法[J]. 北京邮电大学学报, 2008, 31(2): 108-112.

[12] 孙正兴, 彭彬彬, 丛兰兰, 等. 在线草图识别中的用户适应性研究[J]. 计算机辅助设计与图形学学报, 2004, 16(9): 1207-1215.

作者：[陈晨](#)，[吴建国](#)，[CHEN Chen](#)，[WU Jian-guo](#)
作者单位：[安徽大学 计算机科学与技术学院, 安徽 合肥, 230039](#)
刊名：[计算机技术与发展](#)

ISTIC

英文刊名：[Computer Technology and Development](#)

年，卷(期)：2013(9)

参考文献(12条)

1. [Du Youtian, Chen Feng, Xu Wenli](#) [A survey on the vi-sion-based human motion recognition](#)[期刊论文]-[Acta Electronica Sinica](#) 2007(01)
2. [张睿, 张锡恩](#) [虚拟场景中基于数据手套的交互方法](#)[期刊论文]-[计算机工程](#) 2005(12)
3. [任海兵, 祝远新, 徐光佑](#) [基于视觉手势识别的研究-综述](#)[期刊论文]-[电子学报](#) 2000(02)
4. [章丰明, 任彧](#) [基于 MEB-SVM 的静态手势识别研究](#) 2010(06)
5. [谭祺, 肖南峰](#) [基于改进RCE和RBF神经网络的静态手势识别](#)[期刊论文]-[计算机工程与应用](#) 2011(07)
6. [孙丽娟, 张立材, 郭彩龙](#) [基于视觉的手势识别技术](#)[期刊论文]-[计算机技术与发展](#) 2008(10)
7. [Web Real-Time Communications Working Group](#) [WebRTC 1.0:Real-time Communication Between Browsers](#) 2012
8. [HyBi Working Group](#) [The WebSocket API](#) 2012
9. [Mozilla developer network](#) [Creating custom Firefox extensions with the Mozilla build system](#) 2012
10. [张毅刚, 曹阳, 项学志](#) [静态背景差分运动目标检测研究](#)[期刊论文]-[电子测量与仪器学报](#) 2010(05)
11. [马义德, 张红娟](#) [PCNN与灰度形态学相结合的图像去噪方法](#)[期刊论文]-[北京邮电大学学报](#) 2008(02)
12. [孙正兴, 彭彬彬, 丛兰兰](#) [在线草图识别中的用户适应性研究](#)[期刊论文]-[计算机辅助设计与图形学学报](#) 2004(09)

本文链接：http://d.wanfangdata.com.cn/Periodical_wjfz201309008.aspx