

基于手势识别的智能输入^①

张文清, 李芬兰, 欧海燕

(汕头大学 工学院电子系, 汕头 515063)

摘要: 针对在人机交互中键盘远远不能满足人们的要求, 提出了一种手势识别算法, 并结合键盘的按键实现智能输入. 该算法采用肤色分割提取出手部区域, 根据手形检测出大拇指和伸出的手指数目及位置, 把手势转化为计算机所能接受的二进制语言. 根据二进制数, 结合键盘的按键, 把不同手势定义为不同的按键, 实现对计算机的智能输入. 实验结果表明该算法具有一定的鲁棒性, 并能达到较高的成功率.

关键词: 手势识别; 智能输入; 肤色分割

Intelligent Input with Gesture Recognition

ZHANG Wen-Qing, LI Fen-Lan, OU Hai-Yan

(Department of Electronic Engineering, Shantou University, Shantou 515063, China)

Abstract: In the human-computer interaction, the keyboard far cannot satisfy the requirement of people. In this paper, a method based on gesture recognition algorithm is proposed, and combine with keyboard keys to realize intelligent input. In the algorithm, the hand region is extracted by using skin color segmentation. According to the hand shape to detect the thumb and the number of outstretched fingers, and their position, then gestures are converted into binary language computer can accept. Combined binary number with the keyboard keys, gestures are defined as the different buttons to achieve compute intelligent input. The experimental results show that the algorithm has certain robustness, and it can achieve a high success rate.

Key words: gesture recognition; intelligent input; skin color segmentation

随着人与计算机的交互活动的越来越频繁, 当今的计算机外部输入设备: 键盘, 鼠标, 遥控, 电子笔等越来越满足不了人们的要求. 手势由于具有自然、方便的人际交流特性, 能够轻松的根据不同的形态代表不同的含义, 可以通过图像的形式把用户的手势含义转化为计算机所能接受的二进制语言, 所以基于手势识别的手势控制越来越成为研究的热点. 这种手势控制的优点是可以先根据文化的差异或用户的习惯来定义手势, 使用户真正的融入到虚拟环境中, 真正的实现人机交互.

国内外对手势识别做了很多的研究, 任彧, 顾成成^[1]提出了一种用梯度方向直方图提取手势的 Hu 不变矩等特征, 并用支持向量机作为机器学习工具来进行识别的方法. 该方法较大程度上消除了手势旋转对

识别结果的影响, 但是需要一定的训练样本才能进行分类识别, 计算复杂度较高. 文献[2-5]采用手形识别技术, 根据手指的长度和宽度、手掌的长度和宽度, 以及手掌的表面积等特征来区分不同的手势. 该方法能达到较高的识别率, 但是需要获取手势图片时的距离不变. 本文考虑一些平台上没有键盘或使用键盘不方便的情况下, 提出了一种快速简单的手势识别方法, 并结合键盘的按键分布, 来替代键盘终端的输入功能.

1 基于手形特征的手势识别的智能输入

算法的基本步骤如图 1 所示: 先用摄像头捕获图片, 进行手势分割和预处理, 然后对手势图片进行分类, 再进行特征提取: 伸出手指的数目和大拇指. 最后根据提取到的特征对手势图片进行识别.

^① 收稿时间: 2014-02-28; 收到修改稿时间: 2014-03-27

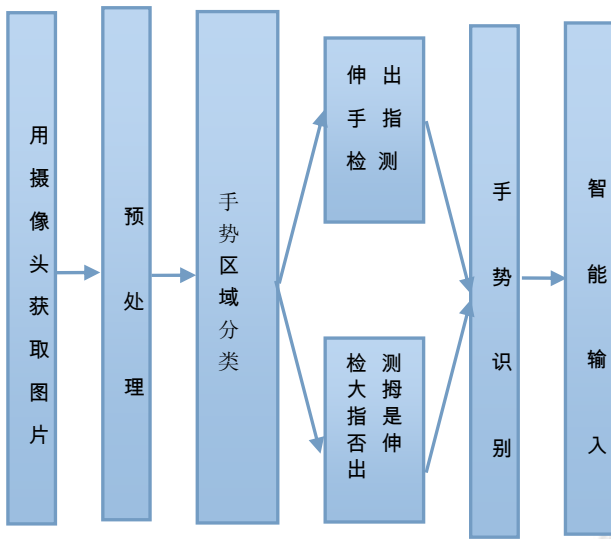


图 1 手势识别的算法流程图

1.1 预处理

为了更好地识别,我们需要在图片中分割出手势区域,而手最大的特征:它是属于人体的一部分,外表是人体的肤色.所以采用在颜色空间中建立肤色模型的方法来对手部区域进行分割.在传统的 RGB 空间,三基色(r、g、b)不仅代表颜色,还表示了亮度.由于周围环境光照的改变,使得皮肤的分割过程是不可靠的.所以常用的 RGB 表示方法不适合于肤色模型,为利用肤色在色度空间的聚类性,需要把颜色表达式中的色度信息与亮度信息分开,所以需要把颜色空间转换为色度与亮度分开的色彩表达空间.颜色空间的转换常用的颜色模型主要有:YCrCb、HSV、YIQ 等.在本文的实验中选用 YCrCb 空间作为肤色分布统计的映射空间,该空间的优点是受亮度变化的影响较小,而且是两维独立分布,能较好地限制肤色分布区域.

根据统计表明,人的肤色区域的像素点是符合椭圆模型的分布的,在分割的第一步是采用椭圆模型分割,Hsu R L, Abdel-Mottaleb M, Jain A K^[6]提出了一个符合肤色区域像素点的分布椭圆模型如下:

$$\frac{(x-e_x)^2}{a^2} + \frac{(y-e_y)^2}{b^2} = 1, \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} cb-c_x \\ cr-c_y \end{bmatrix} \quad (1)$$

其中 $c_x, c_y, \theta, e_x, e_y, a, b$ 为椭圆模型中相应的系数,我们经过多次试验,取以下值:

$$c_x = 109.38, c_y = 152.02, \theta = 2.53, e_x = 1.60, e_y = 2.41, a = 25.39, b = 14.03 \quad (2)$$

通过上面的模型,把图像转换为二值化图像(符合

上述模型分布的赋值为 1,为白色像素点,不符合的赋值为 0,为黑色像素点),但是,此模型有个缺点就是如果有类肤色的物体放在手边上,分割会把手和这个物体连在一起,具有局部收敛性,这样对以后的识别就有很大的影响.由于手掌区域都是属于同一个人的皮肤,也就是说这区域的像素点的相关性是很大的,接下来先选取椭圆模型分割后得到区域的质心,再计算上面用椭圆模型分割后的白色像素点与这个基准点的马氏距离,距离大于一个阈值的话,则把这个像素点剔除掉(赋值为 0).这样解决了上面的局部收敛的问题.也得到了一个更精准的二值化图像.再对二值化图像进行开,闭运算,用腐蚀,膨胀的方法来去除一些小的干扰点.选出联通分量面积最大的区域,这样就把手势区域提取出来.

1.2 手势区域分类

为了更精确的识别出手势,必须对不同角度的手势进行识别,一开始先要把手图片分为水平和竖直两类.分别求出手势区域图像的长和宽,然后求出比例

$$Ratio = \frac{L}{W} \quad (3)$$

其中, L 为手势区域的长, W 为手势区域的宽.若 $Ratio > 1$ 的话,是竖直图像,反之就是水平图像.

1.3 特征提取

1.3.1 大拇指检测

大拇指在手势中是一个很重要的特征,有没有大拇指,手势的含义就不一样.接下来要判断出伸出的手指中是否含有大拇指.

观察得到:如果是一副竖直图像的话,大拇指伸出的区域在手的最左端或者最右端.因为不可能所有手势图片都是标准的竖直图片,为了增加检测的准确度和扩大检测的范围,先用角度变换的方法对图片进行角度平移.对于水平图像,在图 2(a)中,计算质心与最右边端点的连线与水平线之间的夹角.对于竖直图像,在图 2(b)中,计算质心与最上面端点的连线与水平线之间的夹角.角度在一个阈值范围内的话,就进行角度平移,这样可以把图片变为标准的竖直图片.

Panwar M^[7]提出了一种统计的方法来识别大拇指.在图 2(c)中,从最左边的点,最右边的点分别向右和向左画一个矩形,矩形的长和宽满足以下条件:

$$height = s_7 - s_1, width = \frac{(s_6 - s_4)}{6} \quad (4)$$

经过统计表明：如果伸出大拇指，那么大拇指所在的矩形区域内的白色像素点的个数会明显减少，我们可以根据矩形区域内的白色像素点和总的白色像素点的个数之比来选定一个阈值判断大拇指是否伸出。分别计算出两个矩形区域内的白色像素点的个数和总的白色像素点的个数之比 $lrate$, $rrate$ 。根据 Panwar M^[7] 和进行的多次试验表明：这个阈值选取为 0.069，所以当 $lrate$ 和 $rrate$ 中有一个小于 0.069 时，就表明有大拇指伸出。如果两个都小于 0.069 的话，那就代表无大拇指伸出，因为人一个手中不可能有两个大拇指。

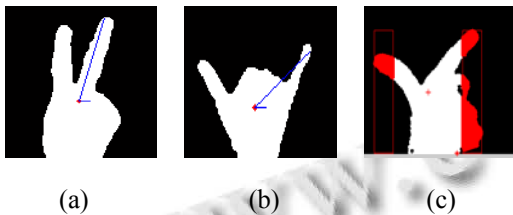


图 2 大拇指检测

1.3.2 伸出手指的检测

一个手指只有一个指尖，可以通过检测出指尖来得到伸出手指的数目。在这里，认为指尖相当于手指的波峰。通过对二值化图像的手指区域的轮廓进行遍历，发现水平图像和垂直图像不同。对垂直图像来说，只要考虑白色像素点的纵坐标的变化。如果纵坐标是先减后增，那就说明到达波峰了。对水平图像来说，只要考虑白色像素点的横坐标的变化。如果横坐标是先增后减，那说明也到达波峰了。如图 3(a)所示，把检测出的点作为指尖候选点时，弯曲手指的顶端也满足上述条件，这样就会对检测产生干扰，所以接下来需要增加判别条件。

通过观察发现，弯曲手指的顶端虽然满足上述条件，但是这些点更加靠近质心，离质心的距离比指尖到质心的距离更小。所以接下来通过欧式距离来筛选指尖。计算公式如下：

$$d_{(a,b)} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \quad (5)$$

其中 a 指的是前面检测得出的指尖候选点， b 指的是二值化图像的质心。通过多次试验表明：选取 $d_{(a,b)}$ 的最大值的 0.64 倍作为阈值，就可以排除掉排除掉弯曲手指的顶端，这样就得到了指尖的个数，也得到了伸出手指的数目。

通过轮廓遍历，不但得出了伸出手指的数目，还可以确定具体是哪个手指伸出。为了达到人机交互，我们把检测结果转化为二进制语言。在图 3(b)中，以 $d_{(a,b)}$ 最大值的 0.64 倍作为阈值，画一条水平线，在这条线以上的就认为是伸出手指的指尖，对应的位置取 1，这条线以下的点则为弯曲手指的顶端，对应位置取 0。算法的伪代码如图 3 所示。

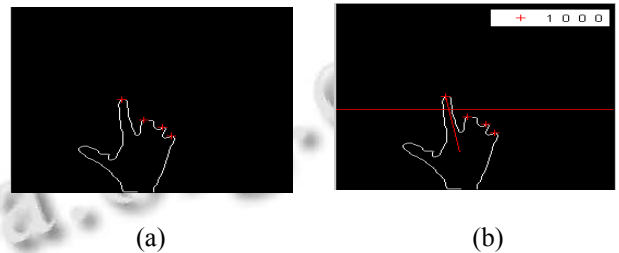


图 3 伸出手指检测

```

1: 因为在竖直图像中，指尖都在质心以上。按轮廓遍历得到质心以上的白色像素点的坐标：
   a(k)=i,b(k)=j;
2: 找出指尖候选点，即 a(k)<=a(k-1)&&a(k)<=a(k+1)的点：
   if(a(k)<=a(k-1)&&a(k)<=a(k+1))
       c(L)=a(k);
       d(L)=b(k);
   end
3: 通过欧氏距离排除弯曲手指的影响，
   threshold=0.64*sqrt((s1-q1)^2+(s2-q2)^2);%s2 为最高点
   count=1;
   for L=1:size(c)
       juli=sqrt((c(L)-q1)^2+(d(L)-q2)^2);
       if(juli<threshold)
           c1(count)=c(L);
           d1(count)=d(L);
       end
       count++;
   end

```

图 4 轮廓遍历的伪代码图

1.4 智能输入

通过上面的检测，可以用 5 位制的二进制数表示手势。如图 5 所示，1 代表该位置的手指伸出，0 代表没有伸出。其中首位表示的是大拇指是否伸出。后面四位依次代表的是各个位置上的手指是否伸出。

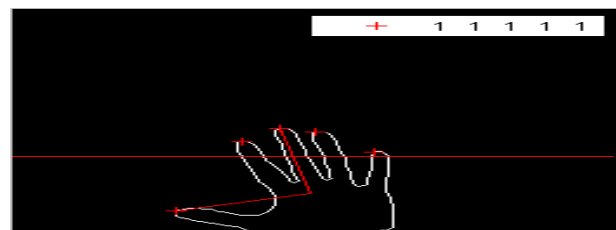


图 5 二进制表示识别结果

在表 1 中，用不同手势代表相应的键，这样就能把手势转化为二进制语言，从而代表不同的键，对计

计算机进行智能输入。

2 实验与分析

为了验证算法的有效性和实时性, 分别根据文献[8]和本文算法, 对每种手势输入 20 张图片进行测试, 测试结果如表 2 所示. 通过 matlab GUI 我们做成了一个实时处理的界面, 用摄像头获取手势进行输入, 实验结果如表 3 所示. 从以上的仿真结果中可以看出: 对于键'e','+'等所代表的手势图片, 这是因为人的手常常不习惯摆弄出这几种手势, 获取手势图片时会产生一定的干扰, 所以识别率偏低. 但是对于整体手势图片而言, 本文算法的平均识别率较高.

表 1 用手势定义的键

3 总结

本文提出了一种基于大拇指和伸出手指检测的手势识别算法, 并结合计算机键盘的按键分布来实现对计算机的智能输入. 该输入能够使用户根据自己的习惯来定义手势, 更好的实现人机交互. 但是, 该算法在图片中出现人脸等大面积的皮肤区域时的识别率较低. 因此未来的研究工作主要是使算法克服图片中出现人脸等大面积皮肤区域的影响以及使算法的实时性

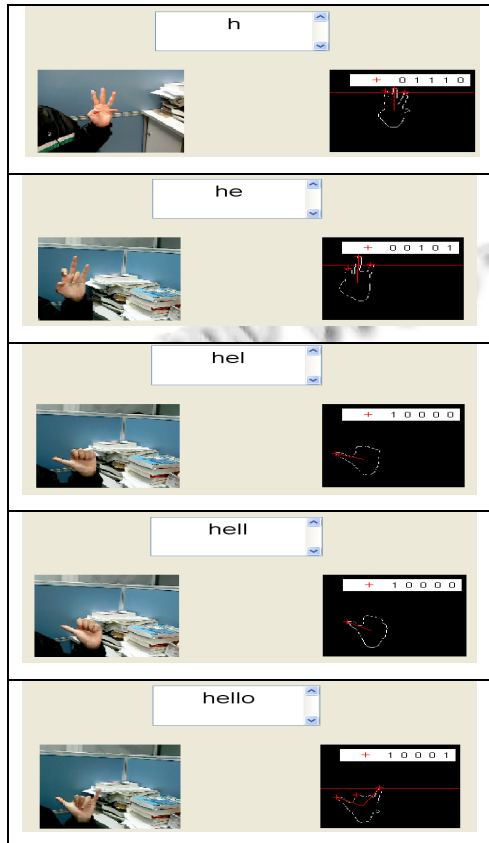
更好.

表 2 手势识别的测试结果

手势	图片数量	文献[8]	本文算法
a	20	1.00	1.00
b	20	1.00	1.00
c	20	1.00	1.00
d	20	1.00	1.00
e	20	0.95	0.95
f	20	1.00	0.95
g	20	1.00	1.00
h	20	1.00	1.00
i	20	1.00	1.00
j	20	1.00	1.00
k	20	0.90	1.00
l	20	0.90	1.00
m	20	1.00	1.00
n	20	0.95	1.00
o	20	0.95	0.95
p	20	1.00	1.00
q	20	0.90	0.90
r	20	1.00	1.00
s	20	1.00	0.95
t	20	1.00	0.95
u	20	1.00	1.00
v	20	0.95	0.95
w	20	0.90	0.95
x	20	0.95	1.00
y	20	1.00	1.00
z	20	1.00	1.00
0	20	1.00	1.00
1	20	1.00	1.00
2	20	1.00	1.00
3	20	1.00	1.00
4	20	1.00	1.00
5	20	1.00	1.00
6	20	1.00	1.00
7	20	1.00	1.00
8	20	0.95	0.95
9	20	1.00	0.95
enter	20	0.95	1.00
CapsLock	20	1.00	1.00
Esc	20	0.95	0.95
space	20	1.00	0.95
Backspace	20	0.95	1.00
shift	20	1.00	0.95
+	20	1.00	1.00

-	20	0.95	1.00
*	20	0.95	1.00
/	20	0.95	1.00
=	20	1.00	1.00
.	20	1.00	1.00
,	20	0.95	0.95
;	20	1.00	1.00

表 3 智能输入的测试结果



参考文献

- 1 任彧,顾成成.基于 HOG 特征和 SVM 的手势识别.科技通报,2011,27(2).
- 2 Sanchez-Reillo R, Gonzalez-Marcos A. Access control system with hand geometry verification and smart cards. Aerospace and Electronic Systems Magazine, IEEE, 2000, 15(2): 45–48.
- 3 Sanchez-Reillo R, Sanchez-Avila C, Gonzalez-Marcos A. Biometric identification through hand geometry measurements. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000, 22(10): 1168–1171.
- 4 Kumar A, Wong DCM, Shen HC, et al. Personal verification using palmprint and hand geometry biometric. Audio-and Video-Based Biometric Person Authentication. Springer Berlin Heidelberg, 2003: 668–678.
- 5 Han CC. A hand-based personal authentication using a coarse-to-fine strategy. Image and Vision Computing, 2004, 22(11): 909–918.
- 6 Hsu RL, Abdel-Mottaleb M, Jain AK. Face detection in color images. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2002, 24(5): 696–706.
- 7 Panwar M. Hand gesture recognition based on shape parameters. 2012 International Conference on Computing, Communication and Applications (ICCCA). IEEE. 2012. 1–6.
- 8 Panwar M. Hand gesture based interface for aiding visually impaired. 2012 International Conference on Recent Advances in Computing and Software Systems (RACSS). IEEE. 2012. 80–85.