

特约稿件

基于深度学习的单目摄像头动态手势识别与交互

孙博文, 于 峰

(哈尔滨理工大学 计算机科学与技术学院, 哈尔滨 150080)

摘 要: 现有手势识别的应用多为基于特定设备实现的, 如 Kinect、Leap motion 等, 因其价格和实用性的原因, 无法得到大规模的普及。而单目摄像头则是电脑或移动设备最为常见的配件, 结合相应手势识别与交互算法, 便可以进行手势交互。因此文中研究了基于单目摄像头的手势识别与交互算法, 通过单目摄像机捕捉图像, 利用肤色进行图像分割, 并根据轮廓面积去除噪点, 得到手势图像的二值图, 然后将二值图的矩阵作为参数传入神经网络进行深度学习模式识别, 通过卷积加全连接网络结构和优化参数选择, 以防止过拟合的产生, 再结合指尖检测和排除算法, 以提高识别精度。在手势交互方面则通过手势重心的改变来判断手势图像平面上的运动, 并进行检测位移, 防止识别到干扰物造成的手势图像位置突变。在手势的前后运动上, 则通过手势图像大小, 对不同手势进行面积补偿调整, 减少因为手势不同带来的面积变化而造成的干扰。

关键词: 深度学习; 卷积神经网络; 手势识别算法; 手势交互

DOI: 10.15938/j.jhust.2021.01.005

中图分类号: TP183 **文献标志码:** A **文章编号:** 1007-2683(2021)01-0030-09

Dynamic Gesture Recognition and Interaction of Monocular Camera Based on Deep Learning

SUN Bo-wen, YU Feng

(School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract: Most of the existing gesture recognition applications are based on specific devices, such as Kinect, Leap Motion, etc. Because of its price and practicability, it can not be widely used. Monocular camera is the most common accessory of computer or mobile devices. With the corresponding gesture recognition and interaction algorithm, gesture interaction can be carried out. In this paper, the algorithm of gesture recognition and interaction based on monocular camera is studied. The monocular camera captures the image, uses the skin color to segment the image, and removes the noise points according to the contour area to get the binary image of the gesture image. Then, the matrix of binary graph is introduced into the neural network as a parameter for deep learning pattern recognition. Through convolution and full-connected network structure, as well as optimization of parameter selection and prevention of over-fitting, combined with fingertip detection and elimination algorithm, the recognition

收稿日期: 2020-04-26

基金项目: 国家自然科学基金(61702140).

作者简介: 于 峰(1972—), 男, 硕士研究生.

通信作者: 孙博文(1963—), 男, 副教授, 硕士研究生导师, E-mail: fractals@163.com.

accuracy is improved. In the aspect of gesture interaction, the motion on the plane of gesture image is judged by the change of gesture center of gravity, and the displacement is detected to prevent the sudden change of gesture image position caused by the interference. In the front and back movement of gestures, the area of different gestures is compensated for and adjusted by the size of gesture images, so as to reduce the interference caused by the area changes caused by different gestures.

Keywords: deep learning; convolutional neural network; gesture recognition algorithm; gesture interaction

0 引言

人机交互一直是计算机领域的研究热点,从早期的穿孔纸带到鼠标、键盘操作,从现在的触摸屏技术到语音识别交互,人与计算机交流方式越来越自然和人性化。而近年来的虚拟现实、增强现实技术的兴起,也带动了手势识别交互技术的发展。

尽管如此,手势交互仍然没有达到广泛应用的程度,其主要原因是用户需求和交互硬件的代价得不到匹配。目前人们使用鼠标、键盘进行人机交互,代价低廉且体验较好,而市面上的手势交互都是靠特定硬件辅助进行的,需要用户进一步资金投入,并且存在应用场景有限、准确率不够高等问题,所以使得手势交互得不到大规模推广。

单目摄像头下的手势识别相比较 Kinect、Leap motion 下的手势识别等较为困难,因为单目摄像头拍摄的影像没有深度信息,也没有红外检测温度信息来帮助识别人手,但一旦研发成功将是最低廉最方便的人机交互方式。本文通过对拍摄到的图片进行处理、计算,可以做到对手势的识别与交互,并且速度更快,占用资源更少。

1 国内外研究发展现状

1983 年,在美国 at&t 公司任职的 Grimes 设计了世界上第一款数据手套,用户佩戴该手套可以与计算机进行交互。1987 年,VPL 公司使用了光纤传感器等技术,对课本上的 260 个单词进行手势识别。1991 年,Vital Technologies 公司设计出了可应用于实际的手势交互设备 ChverGlove^[1]。1997 年,Chan-su-lee 设计一种韩语手势识别算法^[2],对 31 个韩文手势和 131 个韩文符号进行了识别,识别率达到了 96.7%。1998 年,Marcell Assan 和 Kirsti Grobel 提

出了利用隐马尔科夫模型进行手势识别算法^[3],识别效果也有很大提高。2000 年,Chan wah 将图像序列中进行了模块拆分^[4],对不同的任务设置了对应的模块,提取图像中二维二值特征向量,输入神经网络,利用一组递归神经网络(RNN)和隐马尔科夫模型(HMM)链的组合输入,识别时间序列中手势姿态,其识别率达到了 86.8%。Oyedotun 和 Khashman 在手势识别上使用了卷积神经网络和堆叠去噪自编码^[5],识别准确率达到了 91.33%。

随着国外手势识别技术发展,国内在计算机视觉方面也出现了很多优秀的成果。2003 年,邹伟等测量手势的三维空间位置信息^[6],将模糊理论与 HMM 结合,提出一种手势识别算法,平均识别率达到 94.6%。2017 年,郑州大学的范文兵等提出了一种基于肤色特征的手势检测识别方法^[7],识别率最大可以达到 98%。2017 年,易生等提出了 3D 的卷积神经网络代替传统 2D 卷积神经网络的手势识别方法^[8],手势识别识别率高达 95.09%。2018 年,曹洁等提出了基于 RGB_D 的动态手势识别方法^[9],该方法采用聚类算法将手势进行分割,然后将手势的深度信息引入到卡尔曼滤波器中作为跟踪参数之一,在跟踪的过程中对手势进行加窗处理,最后结合快速动态时间规划算法进行动态手势识别,识别率达到 96.8%。

2 手势识别与交互算法

2.1 传统手势识别算法

阈值分割^[10]:根据灰度信息,进行阈值分割,其中包括直方图双峰法与自适应阈值法等。固定的阈值分割法速度快,但是准确率低;自适应系列的多阈值分割计算量大,且结果对阈值较为敏感。边缘图像分割:检测灰度级或者结构具有突变的地方称为

边缘。这种不连续性可通过求导来检测,其位置对应一阶导数的极值点,对应二阶导数的过零点(零交叉点),因此常用微分算子进行边缘检测,其中包括一阶微分算子有 Roberts 算子、Prewitt 算子和 Sobel 算子,二阶微分算子有 Laplace 算子和 Kirsh 算子等。基于区域的分割:区域生长和分裂合并法是两种典型的串行区域技术,其分割过程后续步骤的处理要根据前面步骤的结果进行判断而确定,对每个需要分割的区域找一个种子像素作为生长的起点,然后将种子像素周围邻域中与种子像素有相同或相似性质的像素合并到种子像素所在的区域中,最后从整个图像出发,不断分裂得到各个子区域,将相近性质的子区域合并,实现目标提取。

Haar-like 特征的检测方法^[11]:通过提取图像的边缘特征、线性特征、中心特征和对角线特征,将其组合成特征模板。使用 AdaBoost 级联分类器中进行分类。HOG 特征的检测方法^[12]:HOG 特征是一种在计算机视觉和图像处理中用来进行物体检测的特征描述子,它不是基于颜色值而是基于梯度来计算直方图的,通过计算和统计图像局部区域的梯度直方图来构成特征,然后利用 SVM 分类器进行检测。模板匹配^[13]:在一幅图像中寻找一个特定目标的方法,这种方法的原理非常简单,遍历图像中的每一个可能的位置,比较各个位置与模板是否“相似”,当相似度足够高时,就认为找到了目标,利用凸包算法对手掌轮廓进行多边形逼近,判断其集合形状,对其进行分析得出的分类结果。

2.2 基于 YCBCR 的肤色分割

通过摄像头拍摄的图片默认是以 RGB 编码存储的,将其转换为 YC_bC_r 编码^[14]:

$$\left. \begin{aligned} Y &= 0.257R + 0.564G + 0.098B + 16 \\ C_b &= -0.148R - 0.291G + 0.439B + 128 \\ C_r &= 0.439R - 0.368G - 0.071B + 128 \end{aligned} \right\} \quad (1)$$

式中: Y 为灰度分量; C_b 为亮度与蓝色色度分量差值; C_r 为亮度与红色色度分量差值, RGB 分别代表红绿蓝 3 个颜色分量,式中的参数设定来源于文献^[14]。通过 RGB 3 个颜色分量特定部分叠加可得到亮度 Y ,在比较 Y 与蓝色与红色分量的差值,得出 C_b 与 C_r 的值。

转换完成后,去除不需要的 Y 通道,根据人手的肤色来筛选 C_bC_r 通道^[15],即

$$dst(x, y) = \begin{cases} 255 & 133 \leq C_r \leq 173, 77 \leq C_b \leq 127 \\ 0 & \text{其他} \end{cases} \quad (2)$$

用上述条件进行阈值化处理,将肤色颜色部分提取出来。将会得到多个连通区域,计算连用区域的面积,将过小面积的连通区域作为噪点去除。

本文通过上述利用肤色进行区域提取后,会得到脸部和手势的连通区域,比较脸部和手势部位的区别,会发现脸部的深颜色区域比较多,通过大量测试结果认定计算灰度值在 30 以下的像素所占比例进行区分手势和脸部的连通区域效果较好。

$$H = \frac{N}{M} \quad (3)$$

式中: N 为灰度值小于 30 的像素点个数; M 为所有像素点个数。

比较 H 的大小,如果小于 0.003,那么就可以认为是手势的连通区域,从而将脸部区域剔除。然后将手势区域调整为 25×25 的 0,1 特征矩阵,用于之后的神经网络的识别。

2.3 深度学习之卷积神经网络的搭建

输入层:传入为 25×25 的 0,1 矩阵。

卷积层:输入为 25×25 矩阵,卷积核为 6×6 矩阵,偏置值为 1×1 矩阵,步长为 1,输出结果为 20×20 矩阵。

在卷积层之中,存在卷积核与偏置值,卷积核的规模一般都小于输入规模,它负责对输入进行局部特征提取。设置一个合适的步长,让卷积核对于全部输入参数进行分区域的计算,这样一来,可以降低参数规模,又能尽可能的保留特征点不丢失。卷积计算公式如下:

$$S = \sum_{k=1}^n (X_k * W) + b \quad (4)$$

式中: S 为对应区域矩阵与卷积核矩阵 W 和偏置值 b 计算的结果; n 为根据步长得出的需要计算的区域个数; X_k 是第 k 个计算区域矩阵; $*$ 为矩阵间的对应位置元素相乘。

池化层:输入为 20×20 矩阵,池化范围 2×2 矩阵,输出结果为 10×10 矩阵。并且选用计算最大值作为池化层输出结果。

池化层与卷基层相搭配,其作用是降低参数规模,对于传入的矩阵参数进行压缩,一般有两种压缩方式,可以选择计算其池化范围内平均值或者选择

最大的数作为计算结果,本文选用了最大值池化。

计算最大值池化公式如下:

$$S = \max \left\{ \sum_{i=1}^n \sum_{j=1}^n X_{i,j} \right\} \quad (5)$$

式中: S 为当前区域池化结果; n 为池化规模; $X^{i,j}$ 为当前池化区域参数。

全连接层: 输入为 10×10 矩阵, 权重为 6×100 矩阵, 偏置为 6×1 矩阵, 输出为 6×1 矩阵。

全连接神经网络层上的每一个神经元都对上一层的所有输出进行运算, 通过反向传播, 不断的更改权重和偏置以达到预期目的。相比较于卷积神经网络, 全连接神经网络的运算量更加庞大, 所需要的资源更多。

卷积层激活函数选用 RELU 函数^[6], 全连接层激活函数选用 softmax 激活函数^[7], 代价函数为对数似然函数。

RELU 激活函数:

$$\sigma(z) = \max(0, z) \quad (6)$$

式中: z 为卷积计算后的输出。

Softmax 激活函数:

$$a_i^l = \frac{e^{z_i^l}}{\sum_{j=1}^n e^{z_j^l}} \quad (7)$$

式中: z_i^l 为第 l 层第 i 个神经元的输出; z_j^l 为 l 层所有神经元的输出。

代价函数:

$$C = -\ln a_i^l \quad (8)$$

式中 a_i^l 为最后输出层 (l 层) 第 i 个神经元的输出。

卷积模型搭建如图 1 所示:

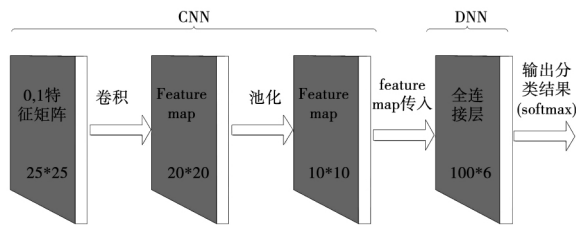


图 1 CNN 模型图

Fig. 1 CNN model diagram

输入 25×25 矩阵参数以 6×6 大下的卷积核进行步长为 1 的卷积计算, 得出到 20×20 的输出矩阵, 然后池化将参数缩为 10×10 矩阵, 用其进行全连接层计算, 用 softmax 激活函数进行最终结果的输出。

2.4 对于卷积神经网络的优化

2.4.1 参数选择

在神经网络的训练中, 参数范围是影响训练速度的因素之一, 相比于图片的灰度值 (参数范围是 0 到 255), 二值图的参数范围 (0 或 1) 小得多。且将二值图与归一化的灰度图对比可知, 二值图 (0, 1 特征矩阵) 的离散程度更高。当选取其作为参数时, 可以更快地降低代价函数。所以, 将手势二值图作为参数来进行神经网络的训练, 可以更好的训练神经网络。

2.4.2 通过规范化减少过拟合的产生

本文是通过调整代价函数的方式来调整权重参数的。一般的, 随着代价函数变小, 准确率不断提高的, 当代价函数很小时, 训练集的准确率可以达到很高, 但如果继续训练, 却会出现测试集的准确率随着代价函数的减小反而降低的现象, 这就是因为训练产生了过拟合的问题。本文通过 L2 规范化^[8]来减轻过拟合的影响, 代价函数改变如下:

$$C' = C + \frac{\lambda}{2n} \sum w^2 \quad (9)$$

其中: C' 为 L2 规范化后的代价函数; C 为原代价函数, 这样一来, 当反向传播进行时, 权重 w 的调整就会从 $w = w - \delta$ 变为 $w = (1 - \frac{\eta\lambda}{n}) w - \delta$, δ 为每次权重的变化量, η 为每次步长; n 为每次梯度下降的训练数量; λ 为设置规范化大小参数。比较权重的变化就会发现, 每次反向传播时 w 的改变都会将 w 的基础上先减小一部分然后再减去改变量。这是因为小的权重在某种程度上, 意味着更低的复杂性, 也就对数据给出了一种更简单却更强大解释, 因此应该优先选择。简单的来说, “减轻”了神经网络对于输入数据的依赖, 降低了训练样本局部特性的识别。

2.5 指尖检测及排除算法

利用卷积神经网络, 可以有一个很好的识别效果, 但是由于性能的限制, 在保障流畅性的前提下, 卷积神经网络的规模不能太大, 与此同时其层数也应该有所限制, 再加上训练样本有限, 这样势必会影响到识别准确率。

发现在神经网络识别过程中, 置信度低于 0.6 时, 其分类结果超过 50% 是错误的, 其中一部分是误分类, 将手势类别弄错, 另一部是非标准手势的干扰。

所以, 为了平衡性能与精度, 本文设计了一个补偿算法, 指尖检测与排除算法。当卷积神经网络得

出的结果小于 0.6 时,根据 softmax 激活函数的定义,说明此时识别的结果很有可能是有误差的,这时就可以进行二次识别,然后综合考虑得出结果。

本文对 0,1 特征矩阵进行了分析判断,该特征矩阵与手势一一对应,根据手掌与手指比例,选取第 8 行(25×25 特征矩阵)作为起点开始分析手指数目(不包括大拇指),横向遍历后,根据 0,1 相对位置得到各个手指起点,然后向上遍历特征矩阵,记录指尖位置,在从第二列进行检测,检查其是否漏检,由此可以得到各个指尖位置坐标。再利用重心设计,寻找手掌重心点,分别计算指尖与重心点连线和竖直方向的夹角,用以判断是否为非法手势。

图 2 为指尖检测算法流程图,具体为:

- 1) 首先根据上文中的图像处理得到手势图像特征 25×25 矩阵。
- 2) 在第 8 行位置寻找第一个 1 以及序列 01 中 1 的位置,将它们存储到链表之中。
- 3) 查找链表。寻找列数最小位置,判断是否为边界处,若是则链表不变,否则在列数为 2 处由下至上竖向寻找 1 以及序列 10 中 1 的位置,计算两处 1 的距离,若大于 4 则链表不变,否则将第二次找到的 1 的位置加入到链表之中。
- 4) 遍历存储位置的链表。每一个都做向上寻找轮廓顶点操作,并将对应链表位置更改为顶点位置坐标。

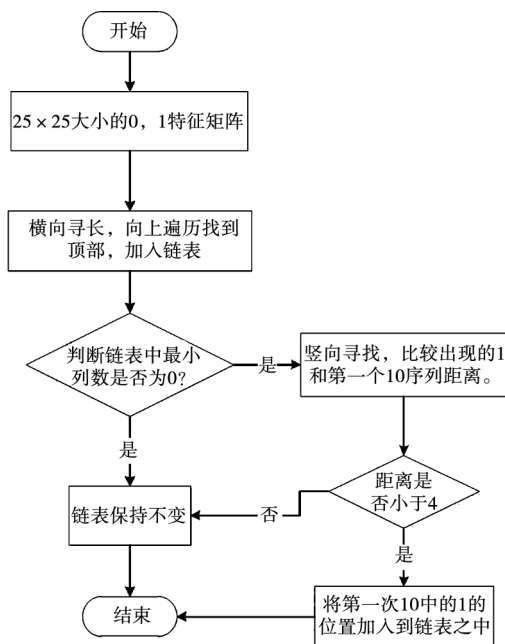


图2 指尖检测算法流程图

Fig.2 Flow chart of fingertip detection algorithm

图 3 为排除算法流程图,具体为:

- 1) 通过指尖算法得到指尖位置信息。
- 2) 将矩阵上的 1 的位置信息求平均值,计算其手势重心。
- 3) 根据指尖与手势重心得出向量 a ,在与手势图像竖直方向向量 $b(1,0)$ 做点乘运算,计算其夹角余弦值。
- 4) 对于每一个指尖得到余弦值进行分析,设定角度,当角度大于 60° (不包括大拇指) 时视为非法手势,否则就根据指尖个数确定手势姿态类别。

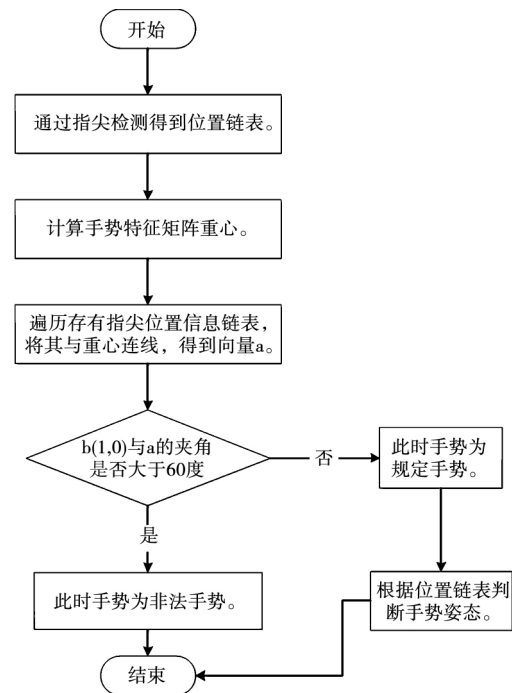


图3 排除算法流程图

Fig.3 Flow chart of exclusion algorithm

2.6 三维空间的手势交互

单目摄像头下,只能得到二维图像信息,因此通过计算手部外框面积来获得远近方向的判断,并进行面积补偿来减少因为手势变化带来的面积改变。

横纵位置改变: 手势姿势的变化会带来手势图像的的面积的变化和重心的变化,所以用重心来作为手势位置信息时不够准确,但是手腕部位却不会受到手部变化的影响,因此选取手势图片最下方特征点作为手势横纵位置变化的基准。

远近位置改变: 在手势图片识别后,根据手势大小画出对应大小矩形框边界,以伸出 5 跟手指的手势姿态为基准,面积补偿公式如下:

$$\left. \begin{aligned} S_0 &= S_0 \times 2.28 \\ S_1 &= S_1 \times 1.28 \\ S_2 &= S_2 \times 1.25 \\ S_3 &= S_3 \times 1.25 \\ S_4 &= S_4 \times 1.2 \\ S_5 &= S_5 \times 1.0 \end{aligned} \right\} \quad (10)$$

其中: S_0 是拳头姿态时的面积; S_1 是伸出 1 跟手指时的面积; S_2 是伸出 2 跟手指时的面积; S_3 是伸出 3 跟手指时的面积; S_4 是伸出 4 跟手指时的面积; S_5 是伸出 5 跟手指时的面积。

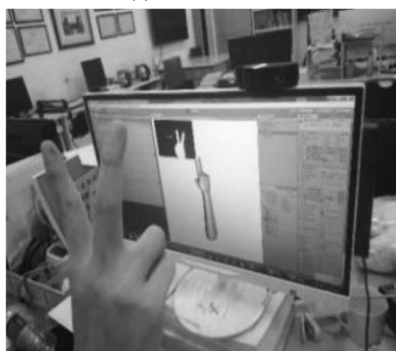
至此, 6 种手势姿态的面积以伸出 5 跟手指的姿态面积为基准, 其他手势姿态面积按照如上公式进行放大补偿, 就可用来作为判断位置远近的信息。

将对应手势映射到 3D 模型上, 让模型根据手势做出不同的动作, 来进行虚拟空间的操作。

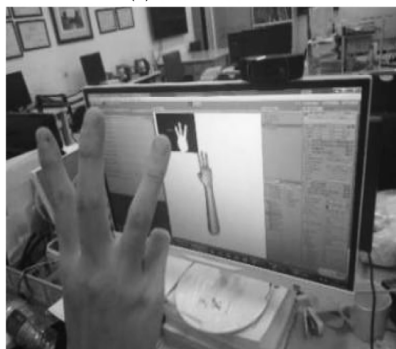
通过手势操控 3D 手势模型如图 4 所示。



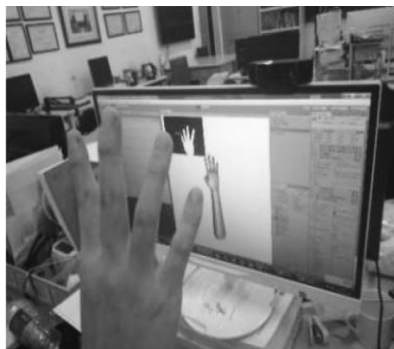
(a) 手势1的识别



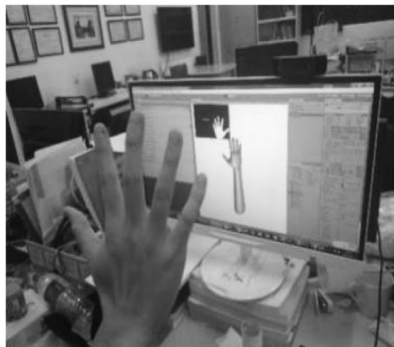
(b) 手势2的识别



(c) 手势3的识别



(d) 手势4的识别



(e) 手势5的识别



(f) 手势6的识别

图 4 手势识别

Fig. 4 Recognition of gesture

3 实 验

3.1 实验环境

本文利用 C# 语言编写卷积神经网络框架以及指尖检测与排除算法。其中神经网络训练样本为 6 000 张图片, 6 种手势各 1 000 张。主要硬件配置: Intel(R) Core(TM) i7-4790K, 8 GB 内存。操作系统: Windows 7 64 位操作系统。

手势识别具体姿态如图 5 所示。

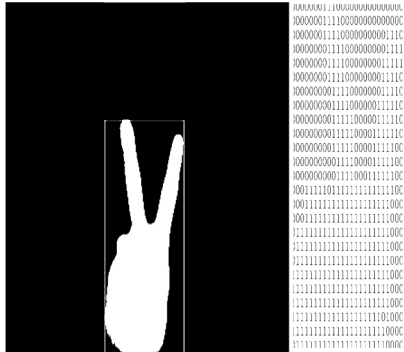
3.2 实验数据

本文采用自制数据集, 通过摄像头采集视频图像, 实时转换为 0, 1 特征矩阵。将其作为神经网络训练集。

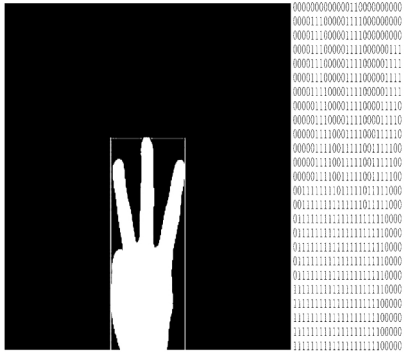
每种手势各采集2 000个特征矩阵,共12 000个。其中6 000个作为训练集,6 000个作为测试集,结果如图6所示。



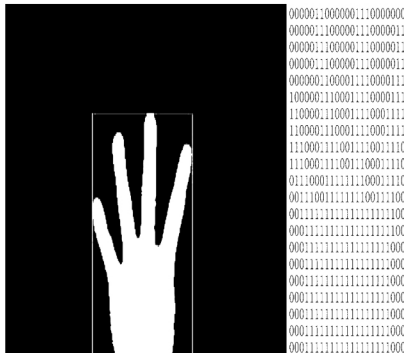
(a) 手势1及其对应手势特征矩阵



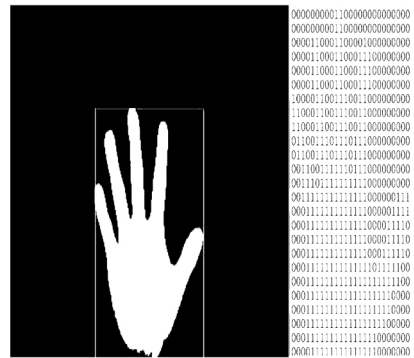
(b) 手势2及其对应手势特征矩阵



(c) 手势3及其对应手势特征矩阵



(d) 手势4及其对应手势特征矩阵



(e) 手势5其对应手势特征矩阵



(f) 手势6其对应手势特征矩阵

图5 手势及其对应手势特征矩阵

Fig. 5 Gesture and its corresponding gesture feature matrix

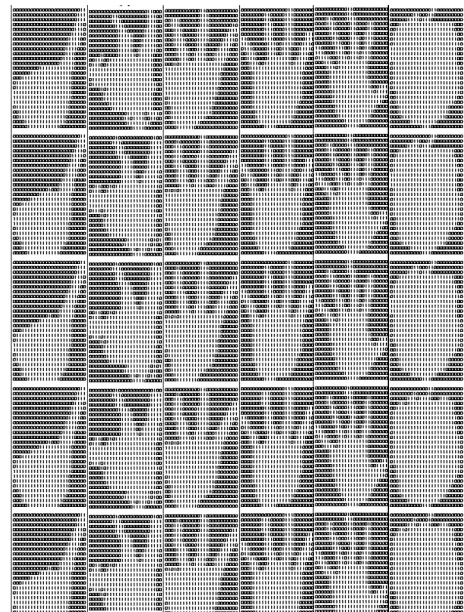


图6 手势特征矩阵训练集

Fig. 6 Training set of gesture feature matrix

3.3 实验结果

用训练集和测试集测试模型准确率,在训练集中,6 种手势的识别率都达到了 100%,测试集结果如表 1 所示。

表 1 测试集识别准确率测试结果

Tab.1 Test results of test set recognition accuracy

手势类别	测试样本数目	识别错误张数	改进后错误数	改进前准确率/%	改进后准确率/%
1	1 000	48	48	95.20	95.20
2	1 000	4	4	99.60	99.60
3	1 000	186	154	81.40	84.60
4	1 000	101	49	89.90	95.10
5	1 000	226	142	77.40	85.80
6	1 000	5	5	99.50	99.50
合计	6 000	570	402	90.50	93.30

表 1 中手势 3 和手势 5 的识别率较低。其原因是当手势 3 具有一定的旋转角度或者手指之间距离较为相近时,容易误判为手势 2;而手势 5 则是因为识别区域不完整,得到的手势二值图与手势 4 较为相似,很多测试样例被误分类为了手势 4 造成的。

手势识别准确率对比如表 2 所示。

表 2 识别准确率对比

Tab.2 Comparison of recognition rate

识别算法	识别准确率/%	识别时间
模板匹配 ^[4]	88.87	>1 s
肤色+SVM ^[2]	93.70	22.81 ms
本文算法	93.30	11 ms

可以看出,本文算法识别速度快,识别效果较好,在保证了准确率的前提下,内存占用小,运行速度快,完全可以满足动态手势识别以及后续的交互使用。

算法模型实际运行时,帧率接近于 60,每帧算法运行平均耗时约为 7 ms,最低 4 ms,最高 11 ms,完全可以达到流畅运行无延迟,程序真实运行速度如图 7 所示。

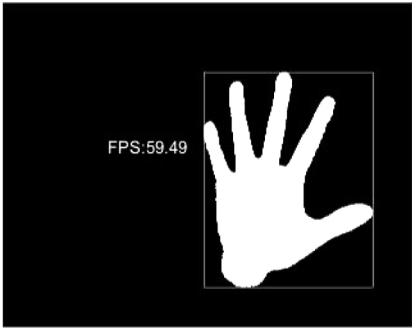


图 7 程序运行真实帧数

Fig.7 Number of Real Frames in Program Running

4 结 语

本文在卷积神经网络的基础加入了改进算法,指尖检测和排除算法,并将神经网络进行了优化,使神经网络能够更好的进行训练。该手势识别算法在样本上表现良好,虽然在测试样本中,手与手之间的形状、角度、对应姿态差异较大时,还是会产生一些误差,但通过改进算法可以很好的提升识别精度,而且也不会占用太多资源。综上,本文提出算法可以做到高精度的实时手势识别与交互。

参 考 文 献:

[1] KESSLER G D, HODGES L F, WALKER N. Evaluation of the Cyber Glove as a Whole-hand Input Device [J]. ACM Transactions on Computer-Human Interaction (TOCHI), 1995, 2 (4): 263.

[2] LEE C S, BIEN Z, PARK G T, et al. Real-time Recognition System of Korean Sign Language Based on Elementary Components [C] // Proceedings of 6th International Fuzzy Systems Conference, Barcelona, 4864, 1997, (3): 1463.

[3] ASSAN M, GROBEL K. Video-based Sign Language Recognition Using Hidden Markov Models [J]. Lecture Notes in Computer Science, 1998: 97.

[4] CHAN Wah, SURENDRA Ranganath. Gesture Recognition Via Pose Classification [C] // International Conference on Pattern Recognition, 2000: 3703.

[5] OYEDOTUN O K, KHASHMAN A. Deep Learning in Vision-based Static Hand Gesture Recognition [J]. Neural Computing and Applications, 2017, 28(12): 3941.

[6] 邹伟, 原魁, 杜清秀, 等. 一种基于 FSMM 的手势识别方法 [J]. 计算机工程, 2003, 29(15): 134.

ZOU Wei, YUAN Kui, DU Qingxiu, et al. A Gesture Recognition Method Based on FSMM [J]. Computer Engineering, 2003, 29

- (15): 134.
- [7] 范文兵, 朱连杰. 一种基于肤色特征提取的手势检测识别方法 [J]. 现代电子技术, 2017, 40(18): 85.
FAN Wenbing, ZHU Lianjie. A Gesture Detection and Recognition Method Based on Skin Color Feature Extraction [J]. Modern Electronic Technology, 2017, 40(18): 85.
- [8] 易生, 梁华刚, 茹锋. 基于多列深度 3D 卷积神经网络的手势识别 [J]. 计算机工程, 2017, 43(8): 243.
YI Sheng, LIANG Huagang, RU Feng. Gesture Recognition Based on Multi Column Depth 3D Convolutional Neural Network [J]. Computer Engineering, 2017, 43(8): 243.
- [9] 曹洁, 赵修龙, 王进花. 基于 RGB-D 信息的动态手势识别方法 [J]. 计算机应用研究, 2018, 35(7): 314.
CAO Jie, ZHAO xiulong, WANG Jinhua. Dynamic Gesture Recognition Method Based on rgb-d Information [J]. Computer Application Research, 2018, 35(7): 314.
- [10] 何俊, 葛红, 王玉峰. 图像分割算法研究综述 [J]. 计算机工程与科学, 2009, 31(12): 58.
HE Jun, GE Hong, WANG Yufeng. Review of Image Segmentation Algorithms [J]. Computer Engineering and Science, 2009, 31(12): 58.
- [11] 王文秀. 基于单目视觉的静态手势识别技术研究 [D]. 浙江工业大学, 2019.
- [12] 任彧, 顾成成. 基于 HOG 特征和 SVM 的手势识别 [J]. 科技通报, 2011, 27(2): 211.
REN Yu, GU Chengcheng. Gesture Recognition Based on Hog Feature and SVM [J]. Science and Technology Bulletin, 2011, 27(2): 211.
- [13] LIU Yun, ZHANG Lifeng, ZHANG Shujun. A Hand Gesture Recognition Method Based on Multi-Feature Fusion and Template Matching [J]. Procedia Engineering, 2012, 29: 1678.
- [14] MICROCHIP G. gm6010/gm6015 Programming Guide [J]. California US: Genesis Microchip Company, 2002: 85.
- [15] 王岩全, 孙博文. 基于单目摄像头的 3D 动态手势交互 [J]. 计算机工程与科学, 2018, 40(9): 1662.
WANG Yanquan, SUN Bowen. 3D Dynamic Gesture Interaction Based on Monocular Camera [J]. Computer Engineering and Science, 2018, 40(9): 1662.
- [16] BOUTILIER C, PATRASCU R, POUPART P, et al. Constraint-based Optimization and Utility Elicitation Using the Minimax Decision Criterion [J]. Artificial Intelligence, 2006, 170(8): 686.
- [17] KRIZHEVSKY A, SUTSKEVER I, HINTON G E, et al. ImageNet Classification with Deep Convolutional Neural Networks [C] // Neural Information Processing Systems, 2012: 1097.
- [18] 谢璐阳, 夏兆君, 朱少华, 等. 基于卷积神经网络的图像识别过拟合问题分析与研究 [J]. 软件工程, 2019(10): 27.
XIE Luyang, XIA Zhaojun, ZHU Shaohua, et al. Analysis and Research on over Fitting of Image Recognition Based on Convolutional Neural Network [J]. Software Engineering, 2019(10): 27.
- [19] 何小磊. 基于深度信息和彩色图像的手势识别 [D]. 华中科技大学, 2012.
- [20] 白玉. 基于指尖定位的手势识别算法研究 [D]. 北京: 北京交通大学, 2016.
- (编辑: 温泽宇)