

融合人脸多特征信息的表情识别系统

魏冉¹⁾ 姜莉²⁾ 陶霖密³⁾

(清华大学计算机科学与技术系, 北京 100084)

摘要 基于对处于不同表情中人脸特征差异的分析,发现用同种方法提取面部各部分特征无法达到信息利用度的最大化,会产生有用信息丢失或者冗余计算,降低了算法的识别准确率和运行速度。针对面部表情改变时,变化最大的3个部分——嘴、额头和眉毛在形状、纹理和距离上的差异,提出用模板匹配法提取嘴部特征,用边缘检测法提取额头特征,用外轮廓检测法提取眉毛特征,并综合这三者的输出得到最终面部表情识别结果的多特征提取识别系统。实验结果验证了该方法的稳定性与有效性,该算法无论在识别准确率还是在整体运行速度上都达到了较高的水平。

关键词 人脸表情识别 多特征提取 模板匹配 边缘检测

中图法分类号: TP391 文献标识码: A 文章编号: 1006-8961(2009)05-792-09

Facial Expression Recognition System Based on Multiple Feature Integration

WEI Ran¹⁾, JIANG Li²⁾, TAO Lin-mi³⁾

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

Abstract Based on the analysis of the characteristics of different expressions in the human face, it was found that using the same method to extract the facial features cannot maximize the availability of information. The lost of useful information and the redundant computing may reduce the accuracy and the performance the algorithm. Aiming at the three specific parts on the face—mouth, forehead and eyebrows—which makes the largest changes in shape, texture and distance when facial expression changes take place, a new multi-feature extraction method is developed. To extract the features, it applies template matching on the mouth, edge detection on the forehead and contour detection on the eyebrows. The integration of the three feature extraction outputs becomes the final result of the system. Experiments validated the algorithm with stability, effectiveness, a high recognition accuracy and fast running speed.

Keywords facial expression recognition, multi-feature extraction, template matching, edge detection

1 引言

在人们的日常交流中,面部表情起着十分重要的作用。它最直接地体现了人们内心的情感状态,为彼此之间的交流奠定了基调。随着计算机视觉及人工智能技术的发展,作为人机交互重要技术之一的人脸情感识别已成为各相关领域十分热门的话题。

一般来说,一个人脸表情识别系统通常包括3

个部分:人脸检测、表情特征提取和表情分类,如图1所示。建立一个人脸表情识别系统,首先必须对人脸进行检测或跟踪,确定人脸的位置,接着再提取与表情相关的特征,最后针对提取出的特征进行分



图1 人脸表情识别系统框架

Fig. 1 Structure of human facial expression recognition system

类和识别。

其中“表情特征提取”部分可谓是整个流程的核心所在,也是整个系统识别率高低的键。近十年来,有关特征提取方面的优秀算法层出不穷。比如基于几何特征的方法,它利用了人脸主要器官的位置和形状会在肌肉牵引下发生一定的几何变化这一特性。这种方法通常是基于基准点来进行分析的^[1-2],如图 2 所示。

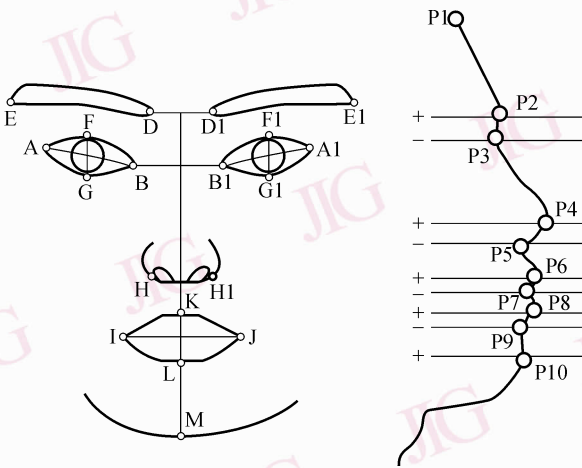


图 2 人脸部件和轮廓特征点

Fig. 2 Example of fiducial points

该方法对基准点提取的精度要求很高,往往在低质量或低对比度图像上无法做到很好的表现。同时它也忽略了皮肤纹理的信息,在细微表情变化的测量上不够敏感。

另一种比较常见的方法是基于外观特征的方法。外观特征一般通过对人脸图像或局部图像按像素进行滤波得到,反映了人脸图像底层的信息。这种方法还分为基于灰度法和基于局部特征法。基于灰度法主要是使用灰度图像信息结合 PCA^[3]和 DCT^[4-5]变换进行特征提取;而基于局部特征法则是利用局部像素之间的关系(梯度、相关性、纹理等)来定义特征,并对每一像素点进行计算,所以它得到的其实是每个局部点附近图像的特征。该方法包括小波变换法(比较著名的有 Gabor 小波^[6]),高阶局部自相关(HLAC)^[7],局部二元模式^[8]等等。

然而,对于所有上面所提到的方法,它们都有一个共性,也就是用同一种方法,或者同一类模型去处理人脸不同部位的特征。这种求统一的思想诚然在算法的适用性及可扩展性上有很大的优势,但是它不可避免地存在一个问题:不可能是局部最优的。

这里所说的“局部”指的是面部的不同部分(比如不同的器官)。举个例子,人在扬眉毛的时候,眉毛的位移是最大的变化。此时如果用光流法应该能够测出比较明显的变化,相反,若用肌肉模型法则很难仅仅将“眉毛”作为计算对象,会引入冗余信息。而对于嘴巴来说,由于其构造比较复杂,可能特征点法会更大程度上简化计算量,提高识别准确度。此外,若想要进一步将上述的算法运用到实时的视频中,则会遇到运行速度上的瓶颈,因为上述的算法都需要大量的图像处理 and 计算,普通的个人电脑的运算速度无法保证算法运行的实时性。

针对面部在表情变化时最明显的三大特征——嘴形变化、额头纹理变化以及眉毛位置变化设计了 3 种不同的特征提取算法,而各个算法也分别都是测量对应特征最平衡的方法,即兼顾了准确性与速度。之后将 3 个算法的输出做一个整合,作为最后的表情识别结果。算法在整体性能和识别准确度上都有很好的表现。

2 算法原理

算法的总体流程如图 3 所示。

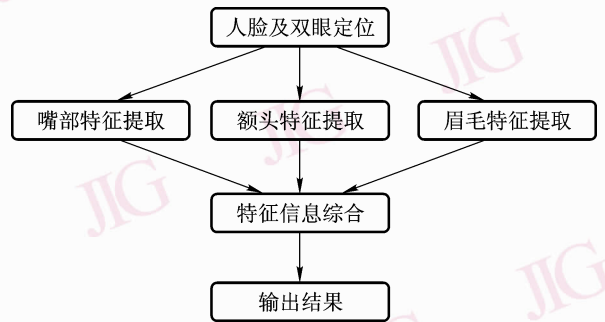


图 3 算法整体流程

Fig. 3 Flow chart of the algorithm

对于无大幅度晃动的正面人脸图像,基于 Haar 特征及 Adaboost 的人脸检测及脸部器官定位算法已经可以得到很好效果。根据人脸及眼睛的准确定位,可以得到嘴巴、额头和眉毛的位置。在此基础上,通过运用 3 种不同的针对性算法,设计了一个多特征提取及综合系统。该系统能够最大程度上提取各部分的特征信息,无论在准确率还是在时间效率上都达到较高的水平。

2.1 人脸及双眼定位

基于灰度图像中弱特征检测的方法是建立在

Haar 特征检测方法^[9-10]基础之上的,并采用了 Bradley 所训练的分类器文件。

检测结果如图 4 所示。

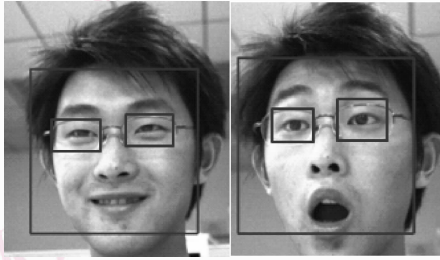


图 4 人脸及双眼的检测结果

Fig. 4 Detection result of human face and eyes

对于正面人脸,无论当前处于什么表情,其检测结果的正确率都是很高的。

人脸及双眼定位是接下去 3 个特征提取算法的基础,主要用于面部器官定位。

2.2 基于模板匹配的嘴部特征提取

通过对大量处于不同表情人脸的观察,归纳出了如图 5 所示的几种典型嘴型。



图 5 不同表情中的典型嘴形

Fig. 5 Common shapes of mouths in different expressions

不难发现,从灰度图上观察,人的嘴部的几个重要部分的颜色存在着很明显的差异:嘴唇的灰度适中,牙齿偏白,口腔偏黑。于是嘴部的灰度图在人脸表情变化时会有较大形状和颜色变化(“悲伤”与“抿嘴笑”时嘴角的下弯上翘,“露齿笑”时嘴巴中间的大块白色,以及“惊讶”时嘴巴中间的大块黑色)。针对这个特点,设计了一种基于模板匹配的认识算法。

算法分为以下几个步骤:

- (1)人脸及双眼定位;
- (2)嘴部匹配范围定位;
- (3)选取匹配模板;
- (4)等比例缩放模板大小;

(5)枚举匹配位置;

(6)匹配;

(7)取最大相似度。

首先通过已经得到的人脸及双眼的位置,确定嘴部所可能存在的位置范围。然后从表情库中取出不同表情所对应的嘴部模板,对其进行一定的缩放处理之后,在人脸图上每个可能的位置进行匹配,取一个最大相似度值作为该模板的相似度。最后在所有的模板中找出相似度最大的一个,将其所对应的表情作为嘴部识别结果。

2.2.1 匹配模板的选取

匹配模板的选取需遵循数量少、覆盖特征大的原则。匹配模板的数量和匹配算法需要的时间成正比,故太多的模板势必会导致算法效率的下降。然而模板数量又不能过少,否则会导致某类嘴形没有被选入模板而最终匹配失败。所以需要在算法运行的速度和准确率之间寻求一个平衡。

经过对 Yale 人脸表情库^[11]和 FGNET 表情库^[12]的研究和筛选,最终选取了如表 1 所示分布的嘴部模板库。

表 1 嘴部模板库的组成

Tab. 1 Sources of the mouth template library

百分比 (%)	来源	特点
60	耶鲁人脸数据库	灰度图,高对比度,数量少
30	FGNET 人脸表情库	彩色图,低对比度,数量多
10	自己收集	彩色图,高对比度,用于弥补一些罕见的嘴部特征

其中,Yale 人脸表情库都是灰度图,而且对比度较高,十分适合于作为嘴部匹配的模板。FGNET 人脸表情库的数量比较多,但它是彩色图像且对比度和分辨率均较低,所以仅仅将其作为对一些特有表情的补充。另外还从网上收集了一些特殊脸型、肤色的表情图片,以进一步完善模板库。

最终的模板库包括 4 种基本嘴部状态——中性、露齿上翘、张开、下弯,每种状态 10 个灰度图模板。

2.2.2 匹配范围定位

在匹配算法中,匹配的范围是影响整体效率的重要因素之一。范围过大,会导致过多的无用匹配尝试,降低速度;范围过小,则很可能会因为没有包括最优值点而丧失最优解。

通过对大量人脸图片的观察及分析,可以得出

如图 6 所示的普遍规律。



图 6 嘴巴在人脸中的位置规律

Fig. 6 Example of locating mouth

对于一般人脸来说,嘴巴完全处于人脸的下半部分,也就是第 2 列图中垂直方向分割线的下方。而在水平方向上,两个嘴角一般不会超出眼角的位置(也就是图 6 中第 3 列人脸的两条竖线),即便是人将嘴巴张大时也满足该规律(图 6 中第 2 行图片即为高兴时咧嘴笑的情况,也是嘴巴左右跨度最大的情况)。此时,通过利用人脸及双眼检测的结果,即可将嘴巴可能存在的范围缩小到整个人脸的四分之一。

2.2.3 匹配模板处理

在进行匹配之前,必须对匹配模板做一定的处理。其中最重要的是对其进行大小上的调整。由于完全不知道图像中嘴巴的大小,只有不断改变模板的大小来“尝试”。

具体方法是将模板在某个大小范围内按照固定的比例进行缩放。这个范围的长宽上界即为先前得到的嘴巴所在范围大小。而下界可将高度设为嘴部所在范围高度的 1/10,宽度根据模板大小等比例缩放即可。由此可以在原模板库的基础上得到一系列新的模板,如图 7 所示。

2.2.4 匹配过程

将缩放后的嘴部模板在原图所有可以摆放的位置上进行匹配。设原图像大小为 $W \times H$,模板大小为 $w \times h$,则一共有 $(W - w + 1) \times (H - h + 1)$ 个可以摆放的位置。根据图 8 所示的方法进行逐个位置的匹配。

图中,灰色轮廓为原图,虚线轮廓为模板,黑色箭头为匹配轨迹。

针对每个可能的摆放位置,使用下式计算结果矩阵 R :

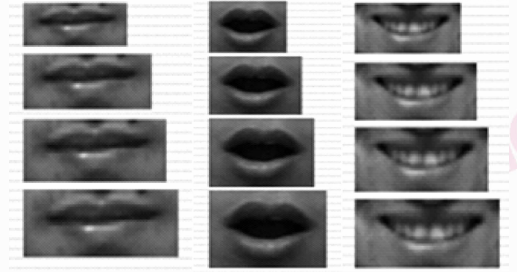


图 7 缩放后的嘴部模板

Fig. 7 The mouth templates generated by zooming

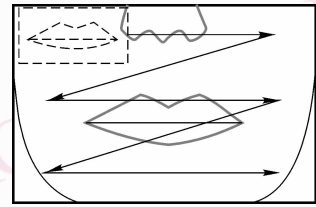


图 8 嘴部匹配过程示意图

Fig. 8 Template matching process of mouth

$$T'(x', y') = T(x', y') - \frac{1}{w \cdot h} \cdot \sum_{x'', y''} T(x'', y'') \quad (1)$$

$$I'(x + x', y + y') = I(x + x', y + y') - \frac{1}{w \cdot h} \cdot \sum_{x'', y''} I(x + x'', y + y'') \quad (2)$$

$$R(x, y) = \sum_{x', y'} \frac{T'(x', y') \cdot I'(x + x', y + y')}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}} \quad (3)$$

其中, I 为原图像矩阵, T 为模板矩阵, R 为最终结果矩阵,大小为 $(W - w + 1) \times (H - h + 1)$ 。所有的求和操作均是在模板和原图的重叠区域内求和,即 x', y', x'', y'' 的取值范围为 $x' = 0, \dots, w - 1, y' = 0, \dots, h - 1, x'' = 0, \dots, w - 1, y'' = 0, \dots, h - 1$ 。

取矩阵 R 中的最大值作为该模板的相似度。

最后,从所有缩放过的模板中,取出相似度最高的模板,并将其对应的表情作为嘴部表情识别的结果。

2.3 基于 Canny 边缘检测的额头特征提取

人在处于苦恼或者厌恶的情感时,额头部分肌肉会内缩,带动皮肤表面出现褶皱,也就是平时所说的“皱眉”。衡量“皱眉”程度主要通过提取额头部分的纹理信息,采用 Canny 边缘检测法。Canny 于

1986 年提出 Canny 算子^[13],它与 Marr(LoG)边缘检测方法类似,也属于是先平滑后求导数的方法。

对于一幅人脸图像,在得到了其中人脸及双眼的位置信息之后,就可以大致计算出额头的位置。具体方法如图 9 所示。

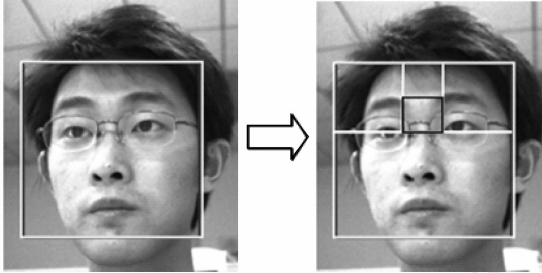
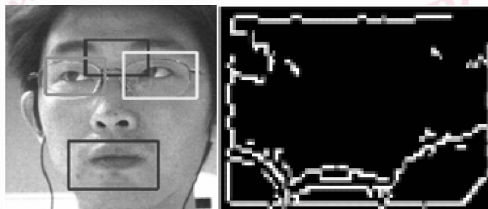


图 9 根据人脸及两眼位置计算额头位置

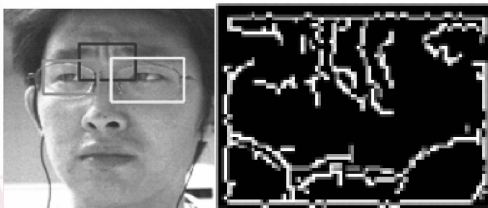
Fig. 9 Example of locating forehead

额头区域的范围是一个矩形。它的左右边界即为左右眼内眼角的位置,下边界为眼角所在水平线,上边界为人脸上界与眼角连线之间的垂直方向平分线所在位置(如图 9 中黑色矩形的边界位置)。

在确定了额头的区域之后,利用 Canny 边缘检测对其进行处理,将结果作为黑白二值图像输出后如图 10 所示。



(a) 中性脸



(b) 厌恶脸

图 10 对额头进行 Canny 边缘检测的结果

Fig. 10 Canny-edge-detection result of forehead

可见,当人脸处于皱眉和非皱眉状态时,额头的 Canny 边缘检测结果在边缘密集度(也就是图中右侧图像的黑白像素比)上有着很大的差别。通过比较当前图像与事先提取的中性脸图像的额头信息,

就能够准确得出当前的皱眉程度。

2.4 基于轮廓检测的眉毛特征提取

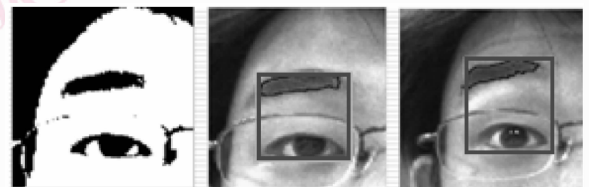
人在惊讶时会不自觉地睁大眼睛,故眼球露出的大小是表情识别中值得提取的特征之一。然而考虑到眼球大小检测的难度比较大,且很容易受到光照等外部条件的影响,于是不得不寻找与其等价的其他特征。进一步观察发现,人在感到惊讶的时候,不仅眼睛会睁大,眉毛会同时被肌肉带动向上扬。根据这个特性,设计了基于轮廓检测的特征提取方法。

2.4.1 眉毛-眼睛距离提取

首先,根据已经得到的双眼位置,对其分别进行向上延伸(延伸 2 到 3 倍眼睛高度的距离),即可得到两幅同时包括眼睛和眉毛的图像。

由于眼镜和眉毛的灰度值都很低,低于整幅图的亮度平均值。故为了更精确地得出眼睛和眉毛的轮廓,先对图像进行二值化,阈值设为平均亮度的 0.6 倍(该值是经过实验得到的较优值),得到的结果如图 11(a)。

对二值化后的灰度图用轮廓检测算法检测所有外轮廓。将面积最大的两个轮廓提取出来,其中必定一个是眉毛外轮廓,一个是眼睛外轮廓。



(a) 二值化后

(b) 中性表情轮廓

(c) 扬眉时的轮廓

图 11 眉毛-眼睛距离检测示意图

Fig. 11 Example of calculating the distance between eyebrow and eye

计算所有轮廓线上点纵坐标的最大最小值,即可得到眉毛眼睛之间纵向距离的绝对值。从图 11(b)、图 11(c)中可以看出,中性与扬眉状态的眉毛眼睛距离(图中矩形的高度)有明显的差距。

然而,单纯计算绝对的距离差值是不行的。当人脸大小变化时,其眉毛-眼睛的距离也会随之改变。因此必须找到一个与眉毛-眼睛距离随人脸大小做等比例变化的相对不变量。而两眼之间的瞳距是很好的选择。

2.4.2 瞳距计算

在已经得到的双眼位置范围内,用边缘算子卷积法计算出眼球的左右轮廓(分别对应卷积后的极

大极小值点)。

所用的边缘算子如下:

$$h_{\text{mask}} = \begin{bmatrix} 0.5 & 0.5 & 2 & 0.5 & 0.5 \\ 0.5 & 0.5 & 2 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 \\ -0.5 & -0.5 & -2 & -0.5 & -0.5 \\ -0.5 & -0.5 & -2 & -0.5 & -0.5 \end{bmatrix}$$

$$v_{\text{mask}} = \begin{bmatrix} 0.5 & 0.5 & 0 & -0.5 & -0.5 \\ 0.5 & 0.5 & 0 & -0.5 & -0.5 \\ 2 & 2 & 0 & -2 & -2 \\ 0.5 & 0.5 & 0 & -0.5 & -0.5 \\ 0.5 & 0.5 & 0 & -0.5 & -0.5 \end{bmatrix}$$

若眼睛图像矩阵为 p , 则设

$$h = h_{\text{mask}} \otimes p, v = v_{\text{mask}} \otimes p$$

$$C_{i,j} = \begin{cases} \sqrt{h_{i,j}^2 + v_{i,j}^2} & v_{i,j} \geq 0 \\ -\sqrt{h_{i,j}^2 + v_{i,j}^2} & v_{i,j} \leq 0 \end{cases} \quad (4)$$

对于矩阵 C , 根据如下规则找到图像每一行中的一对极大、极小值点, 亦为真正的眼球边缘点:

- (1) 极大、极小值点之间的距离大于一定的宽度;
- (2) 极大、极小值点之间的像素灰度值小于一个特定的阈值;
- (3) 极大、极小值之间的“高光点”个数小于一个特定的阈值。

处理后得到的边缘点即为图 12 中的“+”和“×”形点。

接下来, 对边缘点进行最小二乘意义上的最佳椭圆拟合。通过计算椭圆的圆心, 即可得到双眼瞳孔的大致所在位置, 如图 12 中所示。

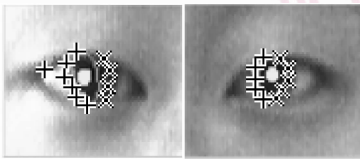


图 12 边缘检测及椭圆拟合结果

Fig. 12 Results of pupil locating

计算双眼瞳孔的距离, 即可得到准确的瞳距。最后, 通过计算眉毛-眼睛距离与瞳距的比值, 就可得到一个不受人脸大小影响的衡量扬眉程度的特征值。再比较当前人脸与事先提取的中性脸的扬眉程度, 即可分析正在检测的人脸是否处于扬眉状态。

2.5 特征信息综合

在成功提取了嘴部、额头和眉毛的特征状态之后, 通过这些信息做一个排列组合, 可以得到最终的人脸表情识别结果, 如表 2 所示。

表 2 3 种特征排列组合表

Tab. 2 Combination of extracted features

嘴部	额/眉			
	正常	皱眉	扬眉	皱眉 + 扬眉
中性	平静	厌恶/愤怒	吃惊	吃惊
露齿上翘	高兴	苦笑	惊喜	惊喜
张开	吃惊	吃惊	惊讶	惊讶
下弯	悲伤	悲伤	吃惊	吃惊

正常人脸一般不会出现同时皱眉和扬眉的情况。识别出该结果很可能是由于扬眉所造成的皱纹引起的皱眉无法识别。所以将“皱眉 + 扬眉”和“扬眉”同等对待;

本算法在经典的 7 种人脸表情(平静、高兴、悲伤、惊奇、恐惧、厌恶、愤怒)的基础上, 合并了难以分辨的(比如厌恶与愤怒)表情, 增加了可以识别的新表情(比如苦笑、惊喜), 从而最终形成了一套全新的 8 种表情分类: 平静、高兴、吃惊(程度浅)、惊讶(程度深)、厌恶(愤怒)、悲伤、苦笑、惊喜。

3 实验及结果分析

3.1 “情感清华”视频人脸表情库

为了测试本算法的识别率, 特地拍摄制作了一个视频人脸表情库, 并将其命名为“情感清华”。该库的具体参数如表 3 所示。

表 3 “情感清华”视频人脸表情库参数表

Tab. 3 Parameters of EmoTsinghua

参数名称	参数值
视频拍摄对象人数	7
视频序列个数	14
视频表情种类	平静、高兴、悲伤、惊喜、厌恶(愤怒)、吃惊、苦笑
视频段数	98
视频格式	BMP 图像序列、AVI 视频
分辨率	640 × 480

该库中还有一些比较特殊的视频序列, 具体如

表 4。

表 4 特殊视频序列表

Tab. 4 Special cases in EmoTsinghua

视频序列名	特点
Zed_glasses	光线较暗
Mach_no_glasses	人脸有明显倾斜
pigger_glasses_low_light	光线很暗
stan_glasses	眼镜镜框在眼睛中间
orlando_glasses	黑色宽边眼镜

该库的拍摄对象全部为亚洲人。所有视频序列分为戴眼镜和不戴眼镜两种。部分内容截图如图 13 所示。

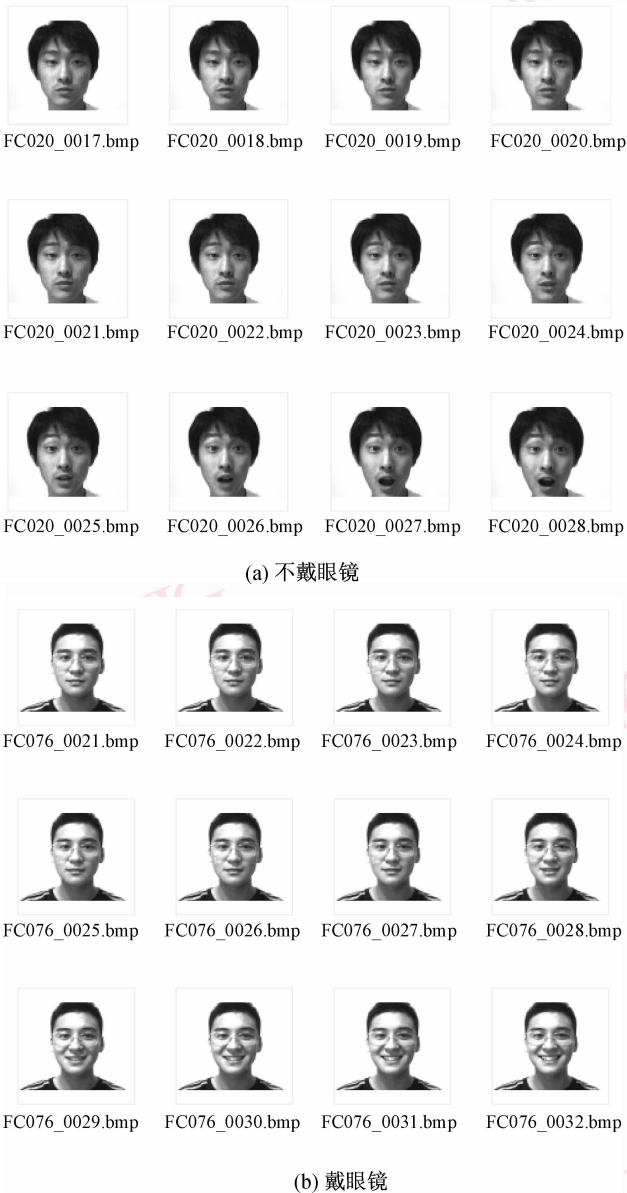


图 13 “情感清华”视频人脸表情库的部分数据截图

Fig. 13 Examples of data in EmoTsinghua

3.2 各模块单独测试结果

针对不同的表情视频,就嘴部模板匹配模块、皱眉检测模块及扬眉检测模块做了单独测试。其结果如表 5 所示。

表 5 各模块单独测试结果表

Tab. 5 The number of successful recognition points in each feature extraction algorithm

部位	成功识别测试点数(个)						
	平静	高兴	悲伤	厌恶	吃惊	苦笑	惊喜
嘴部	14	13	8	11	14	13	11
皱眉	14	14	14	14	14	13	14
扬眉	14	14	14	14	14	13	11

注:每种表情的测试点总数为 14 个,皱眉检测及扬眉检测在测试之前都要经过针对一段中性脸视频的初始化过程。

从表中可以发现,在对平静、高兴和吃惊这 3 种表情嘴部的测试中,模板匹配算法有十分优异的表现。而在悲伤、愤怒这些嘴部变化并无显著特征的表情,该方法的测试结果并不优异,这也是为何要借助后面两种检测方法来得到最终结果的原因之一。

基于 Canny 边缘检测的皱眉检测算法无论是在正确性还是在抗干扰性方面都是十分优秀的。

扬眉检测算法对测试环境以及测试者眼睛周围的状况依赖比较大。过暗的光线可能会导致眼球边缘检测失败,从而丢失瞳距信息;而宽边眼镜则会导致眉毛与眼睛的无法分离。扬眉检测在测试过程中主要的错误发生在如下两种情况:

- (1) 眉毛上扬过度,与眼睛的距离缩小甚至连成一片,导致无法用求轮廓的方法区分眉毛与眼睛;
- (2) 宽边眼镜严重干扰了眉毛与眼睛的区分。

除了以上两种情况,扬眉检测识别正确率接近 100%。

3.3 完整系统识别结果

将 3 种检测算法整合到一起,通过排列组合的方式输出最终的结果,其测试结果如图 14 所示。

从图 14 中可见,对于平静、高兴、厌恶(愤怒)和吃惊这 4 种从外观上很容易区分的表情,该系统的识别率是很高的,前三种表情达到了 100% 的正确率,吃惊表情的正确率为 92.9%;而对于苦笑和惊喜这两种“复合表情”(苦笑由悲伤与高兴复合,惊喜则由吃惊与高兴复合),由于检测的算法只是单纯对面部不同部分的变化进行特征提取,没有充分考虑到“复合表情”所带来的新的面部肌肉或纹

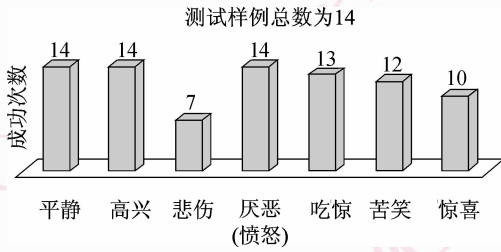


图 14 完整系统测试结果

Fig. 14 The test result of the whole system

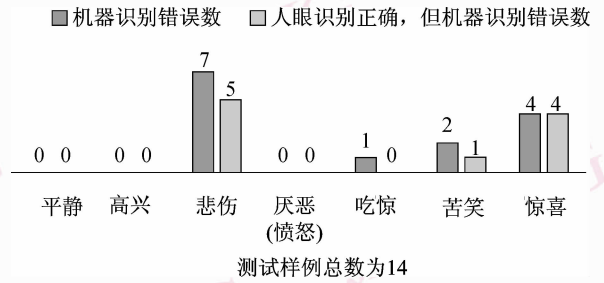


图 15 人眼、机器识别结果对比图

Fig. 15 Recognition result contrast between human eyes and our system

理变化(比如苦笑时由于皱眉和嘴角的上扬,会导致眼睛的高度变小,以至于无法识别),所以识别率要稍微低一些,正确率平均为 83.3%。“悲伤”这个表情比较特殊,因为本身单从形状和纹理的角度,它和“平静”的差别并不大,相对明显的一点差异是嘴角略微下弯。可是有些测试者天生嘴角就有下弯,正是这种不确定性导致最终的识别结果不理想,只有 50% 的正确率。

会造成识别结果错误的还有另一个很重要的原因,那就是库中表情的归类本身就存在歧义。如果连人眼都分辨不出的表情,那让机器来识别就毫无意义了。为此,特地找了 10 个测试者对每段视频进行了人眼辨别,表 6 所示为辨别结果。

表 6 人眼识别测试结果

Tab. 6 The number of test sets successfully recognized by human eyes

	成功识别的测试点数(个)						
	平静	高兴	悲伤	厌恶	吃惊	苦笑	惊喜
成功识别数	140	140	107	132	137	97	127

注:总共 $10 * 14 = 140$ 个测试点。

从表 6 中可以看出,对于“悲伤”和“苦笑”这两类表情视频,人眼识别正确率就不是很高,这是导致本文算法识别失败的直接原因之一。

为了更直观地看出该表情识别系统真正的识别效果,将人眼测试成功识别(人眼识别成功数超过 5 人),但该系统无法识别的表情视频序列数做了归纳,结果图 15 所示。

从图 15 中可知,若不考虑人眼识别因素,整个系统的平均识别正确率为 85.7%;若排除人眼识别失败的测试点,系统平均正确率为 89.8%,均达到较好水平。

4 结 论

针对人脸嘴部、额头和眉毛在表情变化时的形状及纹理改变规律,分别设计了 3 种有很高效率的特征提取算法——模板匹配法、边缘检测法、轮廓检测配合瞳距基准法。然后通过排列组合将这 3 种算法提取出来的 3 种特征信息进行整合,形成了一个高效准确的表情识别系统。多特征提取的一大好处是能够增强鲁棒性,也就是说当一种表情的某个特征提取失败时,可以通过另一个特征的正确提取来弥补。实验结果表明,该方法在具有较快响应速度的同时也具有较高的识别率。

参考文献 (References)

- 1 Campadelli P, Lanzarotti R, Savazzi C. A feature-based face recognition system[A]. In: Proceedings of International Conference on Image Analysis and Processing[C], Mantova, Italy, 2003: 68-73.
- 2 Pantic M, Rothkrantz L J M. Facial action recognition for facial expression analysis from static face images[J]. IEEE Transactions on Systems, Man, and Cybernetics (Part B), 2004, 34(3): 1449-1461.
- 3 Liu Chengjun. Gabor-based kernel PCA with fractional power polynomial models for face recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 25(5): 572-581.
- 4 Akakin H C, Sankur B. DCT Based Facial Feature Extraction[A]. In: Proceedings of the 14th National Signal Processing and Applications Conference[C], Antalya, Turkey, 2006:1-4.
- 5 Jiang Bin, Yang Guosheng, Zhang Huanlong. Comparative study of dimension reduction and recognition algorithms of DCT and 2DPCA [A]. In: Proceedings of Machine Learning and Cybernetics [C], Kunming, China, 2008, 1:407-410.
- 6 Ye Jingfu, Zhan Yongzhao, Song Shunlin. Facial expression features extraction based on Gabor wavelet transformation[A]. In: Proceeding of the Conference on Systems, Man and Cybernetics [C], Hague, Netherlands, 2004, 3:2215-2219.

- 7 Shinohara Y, Otsuf N. Facial expression recognition using Fisher weight maps[A]. In: Proceedings of the 6th International Conference on Automatic Face and Gesture Recognition[C], Seoul, Korea 2004; 499-504.
- 8 Shan Caifeng, Gong Shaogang, McOwan P W. Robust facial expression recognition using local binary patterns[A]. In: Proceedings of the IEEE International Conference on Image Processing [C], Genova, Italy, 2005, 2: II-370-3.
- 9 Paul Viola, Michael Jones. Robust real-time face detection[A]. The 8th International Conference on Computer Vision [C], Vancouver, Canada, 2001, 2: 747.
- 10 Paul Viola, Michael Jones. Rapid object detection using a boosted cascade of simple features [A]. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition[C], Kauai Island, Hawaii, USA, 2001, 1: 511-518.
- 11 Yale University. The Yale Face Database[DB/OL]. <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>, 1997.
- 12 Technical University of Munich. FEEDTUM Database[DB/OL]. <http://www.mmk.ei.tum.de/~waf/fgnet/feedtum.html>, 2005.
- 13 John Francis Canny. Finding Edges and Lines in Images[R]. AITR-720, Cambridge, MA, USA: Massachusetts Institute of Technology, 1983.