

# 轻量级智能终端人脸识别系统研究与实现

马国峻<sup>1</sup>, 周海东<sup>2</sup>

(1. 西安文理学院 信息工程学院, 陕西 西安 710065; 2. 西安电子科技大学 通信工程学院, 陕西 西安 710071)

**摘要:** 针对传统人脸识别算法不能有效适用于智能移动端的问题, 提出一种基于经典 SIFT 算法的特征加权分簇匹配的轻量级改进方案, 该方案能自动学习、自适应添加可靠的测试样本到训练样本空间, 具有合理划分和科学权值分配特性, 使该方案在识别率和运行时间上都有提高。改进算法分别在 ORL 人脸库和 Yale 人脸库做了测试, 相对于经典 SIFT 算法识别率提升了 6.13% 和 14.11%, 运行效率提升了 9.1% 和 4.7%。同时按照 Zhou 的测试方法, 在 ORL 人脸库识别率达到 74.05%, 比 PCA、LBP 等经典算法都有明显的提升, 并在 Android 智能终端中对识别方案做了实现, 实验数据验证了改进算法在 Android 系统的可用性, 最后提出一种基于云架构的改进方案。

**关键词:** 人脸识别; 轻量级; 智能终端; SIFT; 检测区域划分

中图分类号: TP391

文献标识码: A

## Research and implementation of intelligent terminal lightweight face recognition system

MA Guo-jun<sup>1</sup>, ZHOU Hai-dong<sup>2</sup>

(1. School of Information Engineering, Xi'an University, Xi'an 710065, China;

2. School of Telecommunications Engineering, Xidian University, Xi'an 710071, China)

**Abstract:** In order to solve this problem that traditional face recognition scheme was not efficiently suitable to intelligent terminal scene. An improved lightweight scheme of feature weighted clustering matching based on SIFT was presented. The scheme can learn automatically and adaptively add test samples to the training sample space. Reasonable division and scientific distribution of weight make this scheme has improved at run time and recognition rate. The improved algorithm has been tested in the ORL face database and Yale face database, compared with the classical SIFT algorithm the recognition rate improved by 6.13% and 14.11%, the running efficiency increased 9.1% and 4.7%. At the same time, in accordance with the test method by Zhou, in the ORL face database, the recognition rate was up to 74.05%, significantly improved than PCA, LBP and other classical algorithm. The algorithm in Android terminal is implemented, and the improved algorithm is verified to be available in the Android system by experiments. Finally, an improved scheme was proposed based on the cloud architecture.

**Key words:** face recognition; lightweight; intelligent terminal; SIFT; detection zoning

### 1 引言

随着移动互联网的广泛应用, 移动应用的安全性越来越受到重视。身份鉴别是应用安全的核心技术之一, 目前主要采用的是用户口令、手机短信等措施。但由于这些措施与用户自身生物特征无关, 存在被盗用的风险。因此, 发展结合生物特征的移动智能身份鉴别技术非常迫切, 这是移动应用安全

的发展趋势。

人脸识别是目前广泛研究和应用的一种生物特征识别技术, 由于该技术主要基于用户的脸部照片或视频流, 不需要用户太多的配合, 从而拥有非接触、方便直接、友好、真实可靠等优点, 同时, 现在智能移动终端上一般都配备高清摄像头, 因此采用人脸识别技术作为智能移动终端的应用安全鉴别手段是一个很自然的想法。

收稿日期: 2015-11-02

然而, 由于移动智能终端系统, 例如 Android/iOS 手机等, 属于新兴的终端系统, 且自身计算能力和存储空间局限, 同时由于用户操作习惯, 使通常智能终端的训练集往往都是小样本集, 从而导致传统人脸识别算法直接应用于智能终端时效果不佳。因此, 要在智能移动终端上实现有效的人脸识别, 必须对传统的算法进行改进, 并进行相应的实现研究。基于这个背景, 本文提出一种改进的特征加权分块匹配算法, 并在移动端小样本的情形下, 采用自适应添加测试样本到训练样本集的方法进行优化, 同时提出一种基于 Android 系统的实现方案, 实现了一个轻量级人脸识别系统, 并对算法在 ORL 人脸库和 Yale 人脸库做了测试研究, 基于研究结果, 本文进一步提出一种基于云架构的改进方案。

## 2 相关工作

移动智能终端上的人脸识别研究工作很少, 而且平台大多都是较老, 如 BREW、Symbian 等系统。文献[1,2]中采用 Intel 的 PXA270 搭建 ARM 嵌入式系统, 文献[1]中使用肤色模型来检测人脸, 没有提到人脸识别的算法。文献[2]提出一种结合神经网络和 PCA 的算法, 在平台上测试有较高的识别率, 但由于加入神经网络的训练, 需要很多样本且计算代价大, 实用性较差, 同时文献[1,2]中都只是实验平台, 并没有真正地移植到手机或其他成熟的智能移动平台上。由于 Windows Phone 系统和 Windows 系统的程序有着良好的移植性, 因此, 也有些基于 Windows Phone 的研究, 文献[3]中使用多色彩空间的肤色模型, 结合人脸几何关系进一步验证人脸, 不仅检测速度快, 而且极大地降低漏检概率, 不足是仅实现了检测功能。文献[4]基于 Symbian 客户机, 使用 LBP 算法<sup>[5]</sup>来做人脸识别, 并提出一套完整的解决方案, 不足是采用平台较老。可以看到, 智能终端由于自身计算和存储局限的特点, 人脸识别在其上的研究仍需要走很长一段路。

## 3 尺度不变特征变换

尺度不变特征变换<sup>[6]</sup> (SIFT, scale invariant feature transform) 算法是 1999 年 David 教授提出的, 并在 2004 年将理论整理补充<sup>[7]</sup>。该算法旨在从一幅图像中找到一些尺度不变和旋转不变的关键点, 从

而可以在不同的视角变化下更好的匹配物体, 同时这些特征在仿射变换、三维扭曲、加性噪声、高照影响下仍具有较好的区分度。这些特征有很强的识别性, 文献[6]提到对于一个纹理简单的物体, 只要能同时匹配到 3 个不同的特征点, 那么可以认定此物体成功匹配。由于上述这些特点, 此算法常被用于目标识别中。

SIFT 算法工作流程如下。1) 检测极值点。在产生好尺度空间后, 找到局部最大和最小值。每个采样点要与当前层以及上下相邻的 26 个点比较, 得到最大的或者最小的点为候选关键点。2) 筛选关键点。对临近区域的位置、尺度和主曲率信息进行详细的比较, 以过滤掉低对比度和一些不好边缘响应的点。3) 关键点方向分配。为保证图像旋转特征的不变性, 此算法采用了给每个关键点分配一个主方向的方法。4) 生成特征描述子。

## 4 轻量级智能终端人脸识别系统

本节介绍 3 种人脸区域划分方式, 并比较了它们的优缺点, 然后提出一种改进的特征加权分区域匹配的 SIFT 算法, 并结合自动添加测试样本到训练样本机制, 使其适应于智能终端上, 最后给出了在 Android 系统中实现方案。

### 4.1 人脸区域划分

SIFT 算法缺点之一是特征向量的维数有 128 维, 在特征点的匹配过程中需要进行交叉匹配, 因此计算量较大, 然而, 这种计算量和产生的效果却不成正比。因为人脸的独特性基本都集中在一些特征区域, 如眼睛、鼻子等, 而且眼睛和鼻子区域特征向量的交叉匹配是没有意义的。如图 1 所示, 可看到有很多的误匹配。针对这个问题, 本文主要从匹配策略上, 对其进行改进。

常见的区域划分方式分以下几种方式。

#### 1) 平均划分

文献[5, 8]中提出一种平均划分区域并分配权值的方式, 这是一种最简单的划分方式, 即将人脸区域平均划分为  $n \times m$  块子区域。

#### 2) 基于人脸特征的划分

文献[9]将人脸的特征区域划分 6 个区域, 然后分别计算各部分的相似度。脸的基本构造都是相同, 即 2 个眼睛, 1 个鼻子, 1 个嘴巴等, 那么基于人脸固有特征的划分也是一种划分方式。

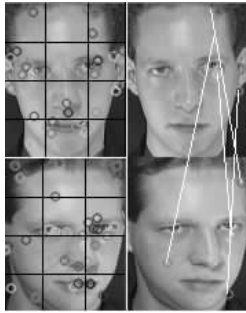


图 1 误匹配

### 3) *k*-means 聚类

文献[10]使用聚类算法对于生成的特征点，根据特征点的位置信息运行聚类算法，将其分到  $n$  类中。*k*-means 是常见的聚类算法，属于机器学习中的期望最大化算法，就是通过不停迭代学习一个模型，使得期望值最大化。

### 4.2 算法改进

SIFT 算法凭借其对尺度、旋转、光照、仿射变换、加性噪声等拥有良好的不变特性，在目标识别中应用很广泛。虽然在不同物体的检测和识别中，SIFT 特征已运用比较成功，但在人脸识别领域还是最近几年才开始使用此算法的。

在自动人脸识别系统中，为了提高识别率，常见的方法是需要一个很大的数据库和很多训练样本，但是遗憾的是，这样大的数据库的建立是非常昂贵的，同时也耗费很多时间。自动人脸识别系统通常是在 PC 端，或有大存储空间、高运算能力的硬件支持，而本文需要在移动端的 Android 系统中实现。那么如何在有很少的训练样本或更有甚者在单样本的条件下，能保持较高的识别度，这是本文关注的问题。SIFT 算法的出现解决了这个疑问，SIFT 算法可以在一张图像（500×500）中提取大约 2000 个稳定的特征点，这为单样本的识别提供了有效的数据支持。

在一般条件下，第 3 节中提到的 3 种分块方法中，*k*-means 无疑是最好的，无论是分类效果还是对脸部转动及遮挡问题都有很好的表现。由于本算法的实现场景是在智能移动终端下，那么运算的复杂度和空间复杂度是首先应该考虑，由于 *k*-means 和基于人脸特征的划分带来额外的运算量，所以应被排除掉。平均划分是实现最简单且计算量最小的，其在划分过程基本没有什么运算量，问题是划分参数该如何选取，同时平均划分在脸部没有对准的情况下，对于表情和遮挡条件下的识别效果下降

很快。

基于以上的分析，本文提出一种改进的识别方案，结合平均划分和基于人脸特征划分的方法，将人脸划分 7 块，在对每块分配权值，计算相似度，并用此作为识别的依据。

本文在划分时，垂直方向划为 3 份，分别代表的是额头到下眼睑、下眼睑到人中、人中到下巴。具体划分比例如图 2 所示，其中  $dd$  代表两眼间的距离。在识别之前，将容易被头发遮挡的额头部分去掉，对人脸做尺寸归一化，从左眼的左上  $\frac{dd}{2}$  处截取长为  $2dd$  的矩形。将双眼分在 1 号、2 号区域内；鼻子在 4 号区域；3 号和 5 号为左右脸颊的位置；6 号和 7 号作为一块区域包含嘴唇和部分下巴，但是实验过程中发现左右嘴角的也会有很多的误匹配，所以分为 2 块，分别包括左右嘴角部分。

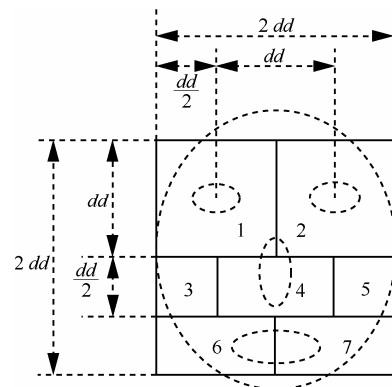


图 2 人脸分块示意

人脸的各个区域辨识度是不同的，假如一个测试样本的眼睛部位有个有效匹配，另一个左脸颊有个有效匹配，那么哪个更接近测试样本呢，直观上眼睛、鼻子、嘴巴及脸型是最好分辨的，所以为进一步增加系统识别率，给每个区域给定一个权值  $w$ 。

每个相应的子区域分别匹配，计算一个相似值。定义第  $i$  个子域相似值  $S_{si}$  为

$$S_{si} = \frac{1}{Means(D_{E-j})} \quad (1)$$

其中， $D_{E-j}$  代表第  $j$  组匹配特征向量间的欧氏距离，匹配规则按照 Lowe 教授的方法。式 (1) 是满足匹配条件的所有向量间距离均值的倒数，代表此区域的相似度，值越大代表越相似。

将所有子域相似值通过加权求平均的方式，得到局部相似度

$$Sim_L = \frac{\sum_i S_{si} w_i}{\sum_i w_i} \quad (2)$$

其中,  $S_{si}$  为第  $i$  块子域的相似值,  $w_i$  为第  $i$  块子区域 (取值是 1~7) 的权值, 具体的权值将在下节介绍。定义全局相似度  $Sim_G$

$$Sim_G = \frac{match(I_q, I_t)}{n_t} \quad (3)$$

其中,  $match(I_q, I_t)$  代表训练样本图片  $I_t$  与测试样本查询图片  $I_q$  之间总的匹配数目, 匹配方法和上文中所用的一致, 还是选择特征向量最近和此近距离的比值小于某一阈值的特征点, 此处阈值取 Lowe 教授推荐的实验参数值 0.6,  $n_t$  表示训练样本的特征点总数。

最后定义图片相似度  $Sim = Sim_G Sim_L$  来衡量训练样本和测试样本间的相似性, 这个值越大越相似。

在方案初始情况下, 每个分组只注册一个训练样本, 但当每次测试样本匹配分组时, 将匹配成功的样本添加到分组中, 最多只保留 2 个训练样本。成功匹配后, 当分组的训练样本不足 2 个时, 直接添加; 当分组的训练样本等于 2 个时, 比较图片相似度, 替换较小的样本。从下文实验中也将看到, 当有 2 个训练样本时已达到可观的识别率, 而且每个分组保存的样本存储代价也不大, 由于在注册阶段不用重复采集图像, 而是在以后运行中以学习的方式自动增加样本, 不用重复提取图像的特征点, 在运行时间上也占很大优势。

### 4.3 基于 Android 的实现方案

本文应用环境是移动智能终端, 由于 Android 系统在智能终端的统治地位, 将算法在 Android 系统上实现。

#### 4.3.1 软件工作流程

本文系统的主要流程如图 3 所示,

首先是移动智能终端通过前置摄像头采集人脸图像; 采集后进入人脸检测阶段, 这部分技术比较成熟, 在 Android 系统中也有相应的人脸检测的函数库, 由于是系统自带的 API, 调用方便且执行迅速; 检测到人脸后, 对面部区域进行截取, 设双眼距离为  $dd$ , 在水平方向, 以双眼中点为中心, 截取 2 倍的双眼距离即  $2dd$  的区域, 在垂直方向, 从双眼上方  $\frac{dd}{2}$  的区域开始截取大小为  $2dd$  的区域, 截

取后的人脸将只保留脸部区域 (不包含耳朵) 和少量的背景区域, 减少头发遮挡及背景的影响; 然后对截取的人脸进行 SIFT 特征提取, 再与数据库中的其他训练样本的图片比对, 通过计算图片相似度来区分属于哪组训练样本或非法用户; 最后将成功匹配的测试样本特征点信息存入数据库中, 并采用上文提到的机制保留图片相似度最大的 2 个样本。

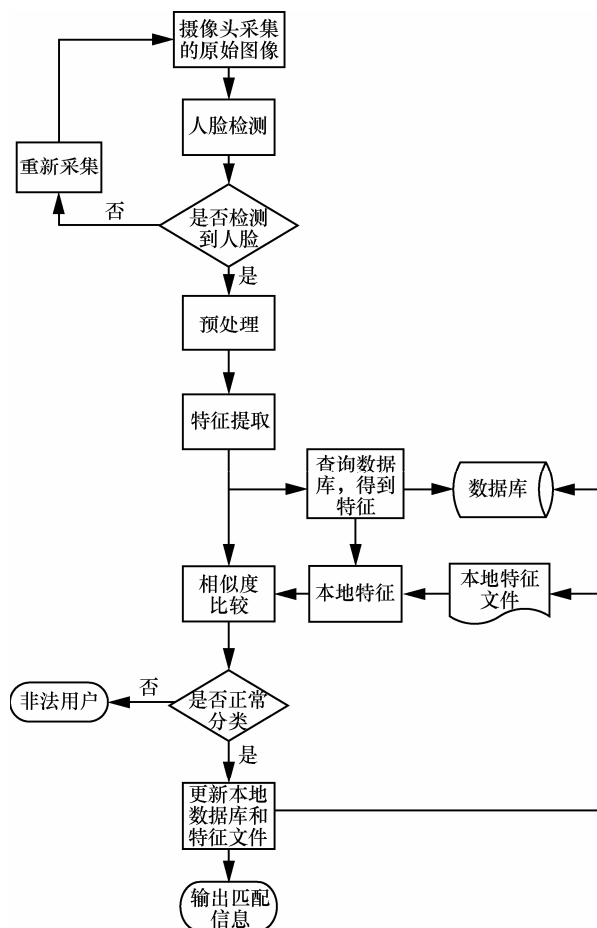


图 3 基于智能终端的人脸识别流程

#### 4.3.2 功能模块设计

本系统主要分为 2 个部分, 注册部分和识别部分。

注册部分包含拍照、检测、预处理、个人信息输入、特征提取、数据存储 6 个模块。识别部分包含拍照、检测、截取、特征提取、匹配 5 个模块。可以看到 2 个部分中有一些模块是可以复用的, 可以不用重复设计, 这样可以保证编码的简洁并提高代码的复用度。下面分别设计每个模块。

##### 1) 拍照模块

此模块用一个 `LinearLayout`, 内部包含一个

SurfaceView 用来实时的预览摄像头采集的图像,在 SurfaceView 下方有 2 个按钮,一个用来拍摄,一个是返回按钮。Android 系统框架支持对设备的摄像头的调用,有丰富的 API 供开发人员使用,用户不仅可以简单快捷的调用摄像头拍照,还可以定制一些高级功能。

## 2) 人脸检测模块

人脸的检测部分,本文采用的是 Android 提供给开发者的 FaceDetector 类。此模块比较简单,首先需要新建一个 FaceDetector 类,入口参数是要检测图片的长、宽和允许检测人脸数目的最大值,然后直接调用 FaceDetector 类的 findFaces 方法,参数为待检测图像和一个用来保存检测脸的 FaceDetector.Face 类型的数组,检测成功时得到目标数量,否则,返回 0。

## 3) 预处理模块

通过 Face 类的 getMidPoint 方法得到双眼的中点坐标,以及 eyesDistance 方法得到双眼间的距离,设中点坐标为  $(x, y)$ , 双眼间距为  $d$ , 则左眼坐标为  $\left(x - \frac{dd}{2}, y\right)$ , 取距左眼左侧  $\frac{d}{2}$  处、上侧  $d$  处的一点,即  $(x-d, y+d)$ , 作为面部截取矩形的一个顶点,然后垂直方向以  $3d$  的长度、水平方向以  $2d$  的长度截取面部区域。截取后的图片存储到指定的本地存储器上。

## 4) 个人信息输入模块

注册时,当截取完人脸后,会弹出对话框,以提示用户输入个人信息。采集完个人资料后,持久化到数据库,根据需求调用。

## 5) 特征提取模块

此模块中用到了 Android 系统的 JNI (Java native interface) 调用。采用混合模式,除了用 Opencv 的 C++库文件外,还使用了 Opencv4Android 的库文件。Opencv4Android 的库文件中提供了一个相对方便的工具类 Utils,调用 bitmapToMat 方法可以快捷的将 Android 中 bitmap 格式转化为 Opencv 中 Mat 格式,后者是 Opencv 中许多函数调用时的参数格式,但是用这个方法转化的矩阵颜色格式是 BGR 的,还需要通过 Imgproc 类的 cvtColor 方法,把 BGR 格式转换为 RGB 格式,否则在显示时会不正常。

此模块属于耗时操作,需要异步处理和检测时一致,都放入一个新的线程中,操作完成时

通过 handler 通知 UI 线程。个人信息采集技术后,将个人信息、特征子本地存储路径、图片本地存储路径一并写入数据库,生成一个训练样本。

## 6) 匹配模块

匹配模块也采用混合模式,核心代码是 C++实现的,用 NDK 方便的生成了 so 文件(和上一个模块中 C++代码共同产生一个 so 文件),入口参数是测试样本的矩阵以及训练样本的特征子本地存储路径。

## 7) 数据存储模块

在其他模块中会用到一些诸如数据库操作、文件读取等操作,将这些操作封装成一个工具类。主要有以下 3 部分。

### ①数据库存取

表 1 展示了人脸数据表的表结构, \_id 用于标示一条记录的编号,设置为主键并自增。name、major、tutor 是在个人信息模块中提到了用户的个人信息。picId1、picId2、picId3 分别代表每个人脸训练样本的采集图片的文件名,可以构造出采集图片的本地存储路径和样本描述子的本地存储路径。

表 1 数据表结构

字段名	数据类型	描述
_id	整型	人脸数据表的主键,记录默认编号
name	字符型	姓名
major	字符型	专业
tutor	字符型	导师
picId1	字符型	图片名 1,可以构造出图片的本地存储路径和描述子的本地存储路径
picId2	字符型	图片名 2
sim	双精度浮点型	图片 1 与图片 2 的相似度

### ②图片操作

主要是图片的存取、缩放和旋转等。图片的存储和读取是一般的文件操作,注意判断 Android 系统是否存在外部存储空间,Bitmap 类中的 compress 方法可以压缩图片成指定格式,压缩率等参数都可选;图片的缩放和旋转操作都用了 Bitmap 类中的方法,缩放用了 createScaledBitmap 方法,可指定长宽,旋转用了 createBitmap 方法,参数中有个 Matrix 实例来控制旋转角度。

### ③特征点文件存取

存储时,首先用 FileStorage 产生一个文件句柄,指定参数为 FileStorage::WRITE,用连续的 2 个 IO 操作符 “<<” 将描述子矩阵以键值对的形式存入,将 write 函数把关键点 vector 容器存入,之后释放文件句柄。

读取时,和存储类似,先用 FileStorage 生成一个文件句柄,指定参数为 FileStorage::READ,用 IO 操作符“>>”从 yml 文件中读取描述子矩阵,然后如同读取数组的一个元素,通过 FileNode features=fs[“keypointVector”]得到 yml 文件的一个节点,接着用 read 函数从节点中读取关键点向量,最后释放文件句柄。

## 5 实验结果与分析

本文对改进算与主要经典算法进行了对比实验,并测试了智能终端下改进算法的性能与有效性。实验表明改进算法有更优良的性能,且在 Android 智能终端上可有效实现。

### 5.1 改进算法性能测试

在 ORL 和 Yale 人脸库测试本文改进算法与 Lowe 教授经典 SIFT 算法性能比较,主要测试指标仍然是识别率和运行时间,这个是智能移动设备较关心的因素。

1) 选择 Yale 人脸库,随机抽每组 1 张用作训练,15 组共 15 个训练样本,同组内其余 10 张作为测试,15 组共 150 张测试样本。划分方式为改进的方式,测试 5 轮取平均。

2) 选择 ORL 前 20 组(编号为 s1~s20)人脸,每组的随机抽取 1 用作训练,其余 9 张为测试样本,一共是 20 个训练样本和 180 个测试样本,每组实验分别测试 5 轮取平均,实验结果如表 2 所示。

通过上述实验数据,可以看到本文方法在 Yale 和 ORL 人脸库均相对于经典 SIFT 算法提升。在 Yale 人脸库中,识别率提高 14.11%,运行时间提高 4.7%;在 ORL 人脸库中,识别提高 6.13%,运行时间提高 9.1%。如表 2 所示,在 Yale 人脸库上识别率的增幅表现更好一点,是因为该数据库中脸部旋转角度较小,可以看到本文算法在表情和小姿态条件下有很好的表现,对于大角度的旋转下,仍有较好的识别率。通过表 2 的运行时长的数据,可以看到在运行时间上本文的方法都有很大的提升。

表 2 Yale 人脸库改进算法性能测试

算法	识别率	测试时间/s
经典 SIFT	0.602 2	306
仅划分 7 块(无权值)	0.66	293
L-SIFT(本文算法)	0.687 2	292
经典 SIFT	0.741 6	602
仅划分 7 块(无权值)	0.748 8	547
L-SIFT(本文算法)	0.787 1	547

改进算法和其他算法在单样本下的性能比较。保持和文献[14]中测试方法相同,即选择 ORL 库,从 s1 到 s40 每组抽一张用作训练,其余 9 张用来测试,即一共有 40 个训练样本,360 个测试样本,测试结果如表 3 所示。

表 3 本文算法和其他算法的性能比较

算法	识别率
PCA	0.663 9
2DPCA	0.697 2
LBP	0.547 2
SIFT	0.708 7
L-SIFT(本文算法)	0.740 5

通过表 3 实验数据,可以看出本文算法相对一些经典算法都有不错的提升,同时由于计算时间的优势,使得是一种不错的在智能终端上的轻量级人脸识别算法,加之增加可容忍的训练样本数量,使效率进一步提升。

### 5.2 基于 Android 的系统性能测试

智能移动终端实验过程采用三星 Galaxy Note 2 手机,此手机处理器为 Exynos 4412 四核 1.6 GHz,内存 2 GB,摄像头(前置)像素 190 万,系统采用 Android 4.1。测试时,选择光照正常、表情自然、脸部旋转较小的条件。

#### 1) 系统可用性测试

本软件按照系统设计编码实现,通过不同用户的多次测试,各模块调用正常,基本功能运行正常,达到安装简单、操作便捷、兼容性好的设计要求。

#### 2) 性能测试

测试选择 15 名用户,其中 10 名作为合法用户,

5 名作为非法用户，即没有在系统中注册的用户。在注册阶段分别记录 10 名用户的注册时间，取平均值得到注册速度；每个用户测试 5 次，分别记录非法用户识别成合法用户次数、合法用户识别为其他注册用户的次数、识别成功的次数及识别所用时间，得到表 4 中的数据。

表 4 性能测试

项目	测试结果
注册速度/ms	746
注册失败率	0
误识率	0.24
拒识率	0.34
识别率	0.66
识别速度/ms	1 323

通过以上数据，可以看到注册速度和识别速度都控制在 2 s 之内，达到用户可接受范围内。由于数据测试较少，一些数据不够完全的精确，但在小样本的条件下，本算法在智能移动终端的移植是成功的，达到的预期，说明本算法在轻量级智能终端人脸识别上的工作方向是正确的，改进是有效的；同时看到本文算法通过在 PC 端的 ORL 人脸库中测试，识别率达到了 74.05%，远高于其他算法，但在实际的系统中，识别率依然很低，主要原因是在实际系统中，照片在无制约的条件下，自由度更多，环境影响更大，同时为了节省手机的存储和计算能力，本系统的图像通过压缩且尺寸控制的较小。目前仍没有可靠的算法在智能移动的实际系统发挥出最理想的效果，在智能终端的人脸识别还有很长的路要走。

## 6 基于云架构的人脸识别方案

通过上述实验，可以看到在实际环境下，识别率并没有在预期高，除了环境影响外，智能终端自身运算及存储能力也制约了识别率的提高，考虑到移动互联网的普及，提出一种基于云架构的识别方案。

如图 4 所示，移动端仍然采用本算法进行特征提取，然后对特征进行压缩后加密传输到云端，云端服务器负责存贮特征数据库和匹配运算，云端服务器处理后的数据在通过网络加密传输到移动终

端上。这个结构减轻移动端的运算成本，且可以通过适当增大云端训练样本的空间，并使用 SVM 或其他分类算法来，进一步提高识别率。这种方案缺点也很明显，即太过依赖于网络，系统可以再运行时检测网络状态，若无网络或网络状态不好，可以切换到第 4 节中的单机模式，当网络良好时使用云架构模式。

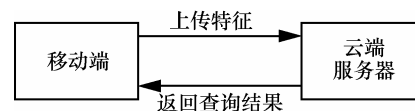


图 4 云架构的人脸识别方案

## 7 结束语

本文在 SIFT 算法的基础上，提出一种改进的轻量级智能终端人脸识别系统，实验证明，在人脸姿态、表情方面有着良好的性能，并在小样本条件下，识别率较高，同时运算速度较快。在 ORL 人脸库和 Yale 人脸库做了相关实验，验证本算法比 PCA、LBP、经典 SIFT 算法在小样本且计算受限的条件下表现出更好的性能。并在 Android 系统中做了测试，验证本文算法在智能移动设备中的性能。最后提出一种基于云架构的改进方案。

### 参考文献：

- [1] LI B, WU S, ZENG F T. Implementation of face identification in intelligent mobile telephone[J]. Computer Engineering, 2006, 32(7): 272-274.
- [2] LI W. Implementation of face identification in intelligent mobile telephone[J]. Computer Technology and Development, 2008, 18(1): 161-163.
- [3] HU X P, CHEN D S, CHEN W B. Design and implementation of face detection system on smart phone[J]. Computer Engineering and Design, 2010, 31(3): 672-675.
- [4] CHEN Y N, CHEN X G. A novel identify recognition system based on mobile phone and face recognition[J]. Computer Application Software, 2011, 28(3): 77-79.
- [5] AHONEN T, HADID A, PIETIKÄINEN M. Face Recognition With Local Binary Patterns[M]. Computer vision-eccv 2004. Springer Berlin Heidelberg, 2004. 469-481.
- [6] LOWE D G. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110.
- [7] LOWE D G. Object recognition from local scale-invariant features[A]. The Proceedings of the Seventh IEEE International Conference on

- Computer vision[C]. 1999.1150-1157.
- [8] CAO S. A fast SURF way for human face recognition with cell similarity[A]. 2011 6th IEEE Conference on Industrial Electronics and Applications (ICIEA)[C]. 2011.166-169.
- [9] ZHOU Z M, YU S Y, ZHANG Z R. An algorithm for face recognition based on sift descriptor[J]. Journal of Image and Graphics, 2008, 13(10): 1882-1885.
- [10] LUO J, SHI Y X, DUAN D Y. Face recognition method based on SIFT feature[J]. Computer Engineering, 2010, 36(13): 173-174.
- [11] HUANG A. Similarity measures for text document clustering[A]. Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC2008)[C]. Christchurch, New Zealand, 2008. 49-56.
- [12] STREHL A, GHOSH J, MOONEY R. Impact of similarity measures on web-page clustering[A]. Workshop on Artificial Intelligence for Web Search (AAAI 2000)[C]. 2000. 58-64.
- [13] ALY M. Face Recognition Using SIFT Features[R]. CNS/Bi/EE Report, 2006. 186-193.
- [14] ZHAO R Z. Study on Face Recognition Based on Single-sample Training Set[D]. Chongqing: Chongqing University, 2012.

#### 作者简介:



马国峻 (1978-), 男, 安徽无为, 西安文理学院讲师, 主要研究方向为信息安全、数字内容保护、嵌入式系统。



周海东 (1989-), 男, 甘肃两当, 西安电子科技大学硕士生, 主要研究方向为信息安全、传感器网络。