

# 一个基于多层结构的快速眼睛定位算法

车昊<sup>1),2)</sup> 黄磊<sup>1)</sup> 刘迎建<sup>1)</sup> 刘昌平<sup>1)</sup>

<sup>1)</sup>(中国科学院自动化研究所,北京 100080) <sup>2)</sup>(中国科学院研究生院,北京 100080)

**摘要** 眼睛的位置是人脸识别过程中一个很重要的信息。针对灰度人脸图像上的眼睛定位问题,提出了一个基于多层结构的快速眼睛定位算法。该算法首先利用极小值区域(MER)进行眼睛粗定位;然后使用三层筛选算法逐步去除错误的候选保留唯一的正确候选;最后使用眼睛精确定位算法来校正眼睛的位置。该算法在CAS-PEAL人脸数据库、JAFFE数据库和自建数据集上均有良好表现,是一个行之有效的办法。

**关键词** 眼睛定位 人脸识别 极小值区域 多层结构

中图法分类号:TP391.41 文献标识码:A 文章编号:1006-8961(2008)03-0472-08

## A Fast Hierarchical-based Algorithm of Eye Locating

CHE Hao<sup>1),2)</sup>, HUANG Lei<sup>1)</sup>, LIU Ying-jian<sup>1)</sup>, LIU Chang-ping<sup>1)</sup>

<sup>1)</sup>(Institute of Automation, Chinese Academy of Sciences, Beijing 100080)

<sup>2)</sup>(Graduate University of the Chinese Academy of Sciences, Beijing 100080)

**Abstract** The eye positions are very important landmarks in face recognition. In this paper a fast hierarchical-based algorithm is proposed to locate eyes from face images. This new algorithm involves three main steps. First, MER (minimum extremal region) is introduced here to locate eye roughly. Then, a three-layer-filter is applied to rule out false detections step by step. Finally, an accurate eye-locating method is used to adjust the eye locations precisely. This algorithm has good results in CAS-PEAL, JAFFE and our own face database. Therefore, it is considered to be an effective method.

**Keywords** eye locating, face recognition, minimum extremal region, hierarchical structure

## 1 引言

近20年来,人脸识别一直是模式识别中最活跃的研究领域之一<sup>[1]</sup>。一个人脸识别算法通常由3个主要步骤来完成<sup>[2]</sup>:首先是人脸检测与定位,然后是特征提取,最后是对提取的特征进行分类识别。人脸的主要特征包括眼睛、鼻子、嘴巴、下巴、脸颊等,这些特征点的定位是否准确将直接关系到整个人脸识别系统的性能。在这些特征中,眼睛区域与鼻子、嘴巴等区域相比较,包含了更丰富的用于个体区分的重要信息<sup>[3]</sup>。眼睛定位可以使其他特征点的定位变得容易和准确,并且可以提高人脸识别和验证的准确性。因此,在许多人脸识别算法中眼睛

定位是第1步。

基于灰度图像的眼睛定位问题是一个很有挑战性的工作,国内外许多学者在这方面进行了深入的研究。Zhou提出了使用通用投影函数(GPF)来定位眼睛的方法<sup>[4,5]</sup>,这种方法的优点是实现简单,计算量小,容易理解,但是实际图像与理想情况往往差别很大,通用投影函数的极值点不明显,很难在此基础上直接准确地找到眼睛位置。Dieckmann先用垂直和水平投影的方法来粗定位眼睛,再使用模板匹配的方法来进一步确定眼睛的位置<sup>[6]</sup>,但是模板匹配方法的计算量太大,而且对于原图像的尺度、光照情况、旋转角度非常敏感。Hamouz等人使用Gabor滤波器检测眼睛的内眼角、外眼角和眼睛的中心位置<sup>[7]</sup>,但是这个方法并不适用于复杂的图像。

Kawaguchi 提出可以在谷图 (Valleys) 的基础上用 Hough 变换来定位眼睛<sup>[8]</sup>,但是 Hough 变换需要高维的参数空间,而且传统的椭圆检测算法的运算量很大,另外,Hough 变换也对图像的尺度、光照、旋转敏感,具有较大的局限性。

本文在总结前人研究方法的基础上,提出了一种基于多层结构(图1)的快速眼睛定位算法,既降低了计算复杂度,又能够有效地定位眼睛,而且还可以在在一定程度上适应图像的尺度、光照和旋转角度的变化,也能够很好地排除眼镜的干扰。该算法采用多层结构,首先使用极小值区域进行眼睛粗定位,得到一组候选眼睛位置,再利用眼睛对的几个自然约束进行粗筛选,然后依次使用两个分类器逐步减少眼睛对的候选数量,最后得到唯一的候选眼睛对。

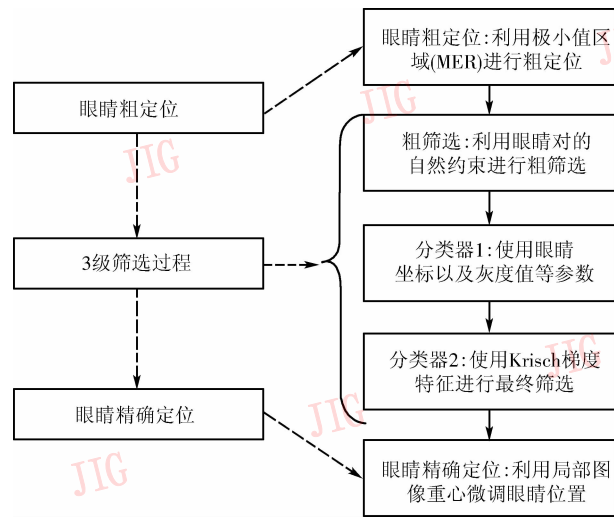


图1 基于多层结构的快速眼睛定位算法结构

Fig. 1 Structure of the fast hierarchical-based algorithm

## 2 使用极小值区域进行单只眼睛粗定位

在应用本文眼睛定位算法之前,先用文献[9]的方法定位人脸(使用 OpenCV 库<sup>[10]</sup>),然后再把人脸区域使用双线性插值法<sup>[11]</sup>进行大小归一化。后续算法中某些特征的值以及算法的实现将依赖于归一化后人脸的尺寸。

人脸图像中的眼睛有一些共性,最显著的特点就是眼球的灰度值比周围区域低许多,极小值区域(MER)正反映了单只眼睛局部图像的这个特点,下面详细介绍了极小值区域的定义、性质以及利用极小值区域进行眼睛粗定位的算法。

极小值区域以及相关的一类概念最先由

Matas<sup>[12]</sup>提出。极小值区域(minimum extremal region)的定义如下(参考了 Matas 的最稳定极值区域 MSER 的定义):

**定义1(区域的定义)** 区域  $Q$  是图像  $I$  的一个连续的子集,并且  $Q$  的像素之间满足邻域关系(如4邻域关系,8邻域关系等,该邻域关系满足可传递性);

**定义2(区域外边界的定义)** 区域  $Q$  的外边界是满足如下条件的点的集合,这些点不属于区域  $Q$ ,但是和区域  $Q$  中至少1个像素满足非传递的邻域关系;

**定义3(极小值区域的定义)** 极小值区域  $Q$  指的是这样的区域  $Q$ ,区域  $Q$  中所有像素的灰度值均小于它的外边界的所有像素的灰度值。如果区域  $Q$  中所有像素的灰度值均小于或等于  $G$ ,则  $G$  被成为是极小值区域  $Q$  的一个临界值。

极小值区域缩写为 MER。另外,为了算法表述方便,临界值为  $g$  的极小值区域被缩写为  $MER(g)$ 。

假设图像  $I$  中的每个像素点灰度值都在区间  $[0, 255]$  内,由定义可知,临界值为 0 的 MER 就是像素值为 0 的黑点(或其连通域),临界值为 255 的 MER 就是图像本身,即  $MER(255) = I$ 。另外,对于  $g < 255$ , $MER(g)$  总被某个  $MER(g + 1)$  包含,如果把这种包含关系定位为父子关系,则图像  $I$  的所有 MER 根据包含关系可以构成一棵多叉树,记为 MER-T。因为所有像素点灰度值都在区间  $[0, 255]$  内,所以 MER-T 的高度为 256。

图2是图像  $I$  以及它的所有 MER 的示意图,图3则是由图2中的图像  $I$  生成的 MER-T(只画出了最具代表性的4层节点,其实树的高度应该是 256)。

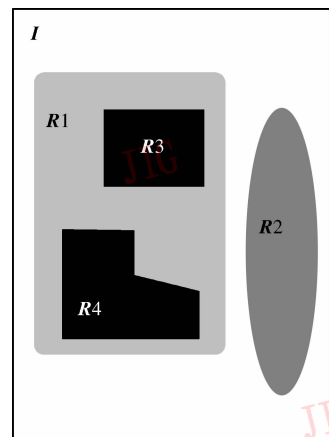
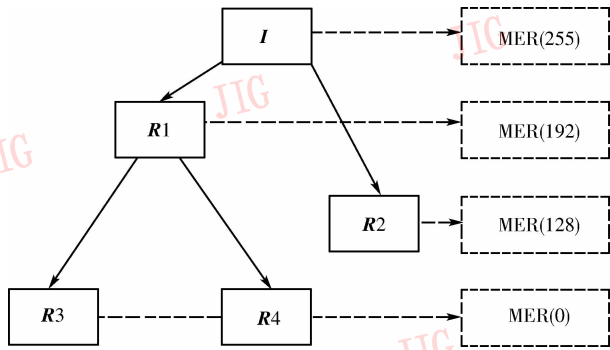


图2 输入图像  $I$

Fig. 2 Input image  $I$

图 3 图像  $I$  的 MER-TFig. 3 The MER-T of image  $I$ 

要想利用极小值区域进行眼睛粗定位,首先要建立输入图像的 MER-T,在文献[13]里对 MER-T 的建立算法有比较详细的讨论。它的主要步骤是首先把所有像素的灰度值以及像素的位置信息按照像素的灰度值从低到高排序,存为数组  $P$  中;其次,从前往后(像素值从低到高)遍历数组  $P$ ,针对每个像素,判断它的邻域情况并做出不同的操作:要么建立一个新的 MER,要么把当前像素加入某个 MER,要么把某几个 MER 合并,也许还需要根据情况建立一些 MER 之间的父子关系。这个算法的时间复杂度是  $O(n \log \log n)$ ,空间复杂度是  $O(n)$  [12]。

在输入图像的 MER-T 已经建立的基础上,眼睛粗定位可以比较容易地实现。眼睛本身有许多自然的约束条件,这里可以使用的几个约束条件有:

- (1) 尺寸约束 单只眼睛的外切矩形尺寸应该在一定范围内(人脸区域的尺寸要先归一化);
- (2) 形状约束 单只眼睛的外切矩形的形状(即高宽比)应在一定范围内;
- (3) 饱和度约束 单只眼睛的面积与其外切矩形面积之比应大于某阈值(如 0.5)。

根据上述约束件,在 MER-T 的所有节点中筛掉不合适的候选,在剩下的所有 MER 里,如果有直接的或者传递的父子关系,则去除父 MER 保留子 MER,最后剩下的就是眼睛粗定位的结果,即候选眼睛。

### 3 对粗定位结果进行筛选

要进行筛选,首先要将前面算法得到的单只眼睛位置组合成为眼睛对的位置。假设使用极小

值区域进行粗定位的结果是  $N$  个候选眼睛位置,将它们两两组合,根据水平坐标的大小可以确定左眼和右眼,这样得到的候选眼睛对的数量是  $N \times (N - 1) / 2$ 。下面的筛选算法将应用在这些眼睛对上。

本文的筛选算法包括 3 层,每个层次的算法都能去除一些错误候选。第 1 层使用自然约束条件进行粗筛选;第 2 层提取眼睛对的坐标、相对位置、灰度值等参数作为特征,然后构造分类器进行筛选;第 3 层提取眼睛局部图像的分块 Krisch 梯度特征,然后和第 2 层一样构造分类器进行筛选。经过上述 3 个层次的筛选,眼睛位置就可以唯一地确定了。

#### 3.1 使用自然约束条件进行粗筛选

要排除错误眼睛对的干扰,一个简单的方法是利用眼睛对的自然约束条件,下面列了两个简单有效的约束:

(1) 双眼间距约束 这里考虑的是水平间距和垂直间距。水平间距应该被约束在一定范围内;人脸图像已经做了旋转角度的归一化,所以双眼垂直间距的绝对值不应该过大,越接近零越好。

(2) 双眼位置约束 因为单独考虑了间距约束,所以这里的位置约束可以只考虑双眼中心点(位于鼻梁附近)的位置。显然,这个中心点应该大体在人脸图像垂直的对称轴上,而且不能太偏下,因为太偏下意味着脸部感兴趣区域(Region of Interest,即以双眼的两个中心点和嘴部中心点为特征点确定的矩形区域)很可能不完整,影响人脸识别后序工作的准确性。

根据上述约束件,为每一个候选眼睛对计算可信度。首先从候选眼睛对的坐标  $(x_l, y_l)$  和  $(x_r, y_r)$  里提取 4 维特征  $F_i$ (该特征依赖于归一化后人脸区域的尺寸):

$$F_i = \left( x_r - x_l, y_r - y_l, \frac{x_r + x_l}{2}, \frac{y_r + y_l}{2} \right)$$

式中,  $i = 1, \dots, \frac{N(N-1)}{2}$ 。

根据实验统计结果,特征  $F_i$  满足多维高斯模型(实际上,这里的高斯模型可以与 MER-T 的建立过程相结合以加快算法速度),因此可以令可信度为

$$C_i = (2\pi)^{-\frac{d}{2}} \left| \sum_{i=1, \dots, \frac{N(N-1)}{2}} \left|^{-\frac{1}{2}} e^{-\frac{(F_i - \mu)^T \Sigma^{-1} (F_i - \mu)}{2}} \right. \right.$$

式中,  $d=4$ ,  $\Sigma$  是特征  $F$  的协方差矩阵,  $\mu$  是特征  $F$  的均值。将所有候选眼睛对按照可信度  $C_i$  降序排列序, 寻找正整数  $t$  使得前  $t$  个候选眼睛对的可信度之和大于所有眼睛对的可信度之和的 90%。这样, 前  $t$  个候选眼睛对被筛选出来进入下一层的筛选。

### 3.2 提取自然约束条件特征使用分类器进行细筛选

前面的粗筛选只是为了降低问题的规模, 把一些很容易就可以判别为错误的眼睛对及早地排除。下面构造一个分类器, 提取所有的自然约束条件作为特征进行训练, 然后观察结果去掉一些影响不大甚至起反作用的特征。该分类器最终选取的特征是: 双眼的基本坐标  $(x_l, y_l)$  和  $(x_r, y_r)$ 、双眼水平间距  $d_h = x_r - x_l$ 、双眼垂直间距  $d_v = y_r - y_l$ 、双眼灰度值  $g_l$  和  $g_r$ 、均值  $(g_l + g_r)/2$  和差值  $(g_l - g_r)/2$ , 一共是 10 维特征。下面先使用 K 均值对正负样本分别进行聚类, 得到均值模板, 然后用 LVQ<sup>[14]</sup> 对聚类结果进行优化。这个分类器采用输出值最大的 3 个候选眼睛对送入下一个分类器。

### 3.3 提取分块的 Krisch 梯度特征使用分类器进行最终的筛选

经过前面两层的筛选, 许多错误的候选眼睛对已被去除了, 但是眉毛、黑色眼镜框、鼻孔等干扰仍然存在。通过观察这些干扰样本可以发现, 它们的位置、中心点的灰度值等参数与真正的眼睛并无差异, 所不同的就是周围区域的灰度值分布情况。因此可以以候选眼睛位置为中心, 在周围选取一定大小(记为  $M \times N$ )的矩形作为特征提取区域, 将这个特征提取区域划分为若干个相同大小(记为  $m \times n$ )的块, 在每一小块里提取  $k$  维的 Krisch 梯度特征( $k=4$ 或 $8$ ), 方法如下(以  $k=8$  为例): 遍历该小块, 考察点  $(i, j)$  的 8 邻域(图 4):

$P_0$	$P_1$	$P_2$
$P_7$	$(i, j)$	$P_3$
$P_6$	$P_5$	$P_4$

图 4 8 邻域示意图

Fig. 4 Eight-neighbourhood indication

令  $S_k$  和  $T_k$  为(其中下标里的加法为模 8 加法):

$$S_k = P_k + P_{k+1} + P_{k+7}$$

$$T_k = P_{k+2} + P_{k+3} + P_{k+4} + P_{k+5} + P_{k+6}$$

点  $(i, j)$  8 个方向的梯度依次为

$$G_d(i, j) = 3S_d - 5T_d, d = 1, \dots, 8$$

当前小块的 8 维 Krisch 特征为

$$K = \sum_{i,j} (G_0(i, j), G_1(i, j), \dots, G_7(i, j))$$

这样, 每个眼睛对提取的特征维数为  $\lfloor M/m \rfloor \times \lfloor N/n \rfloor \times k \times 2$  维(其中,  $\lfloor \cdot \rfloor$  表示向上取整)。经过多次实验发现, 512 维特征可以取得较好效果。下面仍然先使用 K 均值对正负样本分别进行聚类, 得到均值模板, 然后用 LVQ<sup>[14]</sup> 优化结果。这个分类器得到的输出值最大的候选眼睛对就是最终定位结果。

## 4 眼睛精确定位

前面的定位已经大体上标示出了眼睛的位置, 但是对于人脸识别这个应用目标而言还需要对眼球的位置做进一步修正和精确定位。如前所述, 眼球最显著的特征就是灰度值比周围区域明显要低, 而且眼球的对称性非常好。因此这里先在前面定位结果的周围选取一块区域(但是至少应该包含完整的眼睛), 再选择该区域中灰度值比较低的一部分像素, 然后计算它们的重心位置。眼睛位置的修正公式如下:

$$(E_x, E_y) = \frac{\sum_{x,y} f(x, y) (T - g(x, y)) (x, y)}{\sum_{x,y} f(x, y) (T - g(x, y))}$$

$$f(x, y) = \begin{cases} 1 & g(x, y) < T \\ 0 & \text{其他} \end{cases}$$

式中,  $g(x, y)$  表示图像中点  $(x, y)$  的灰度值,  $T$  是一个灰度阈值, 低于这个阈值的像素点才被考虑。一般来说,  $T$  的选取应当使得眼睛周围区域中至少一半的像素点都能被考虑进来。经过这个公式修正后的眼睛位置就是本文算法最终的定位结果。

## 5 实验分析

### 5.1 实验数据

为了说明并验证本文算法的效果, 分别在 CAS-PEAL 人脸数据库<sup>[15]</sup>、JAFPE 数据库<sup>[16]</sup> 和自建数据集上做了实验。在 CAS-PEAL 人脸数据库中, 本文选取了“\Fronta\Normal”目录下的所有不分性别、年龄的 1 040 张人脸图像, 每人一张。JAFPE 数据库包括 10 名女性人脸图像, 分 7 种表情, 一共 213 张图像。自建数据集是实验室采集的实际考勤照片, 由 86 个人, 每个人 2 张到 170 张不等, 一共 2 025 幅图像组成。

## 5.2 实验方法及实验结果

使用通用的指标  $err^{[4]}$  来定量地衡量算法的效果。假设  $d_l(d_r)$  是左(右)眼定位结果和左(右)眼标定位置之间的欧氏距离,  $D$  是双眼标定位置之间的欧氏距离, 则  $err$  可以定义如下:

$$err = \frac{\max(d_l, d_r)}{D}$$

以下 4 个实验是为了说明并且检验算法各部分的作用而设计的。实验以 (CAS-PEAL) 数据库为例。表 1 给出了这 4 个实验与算法各步骤的关系:

表 1 4 个实验与算法各步骤的关系

Tab. 1 Relations between four experiments and algorithm steps

	实验号			
	1	2	3	4
粗定位	√		√	√
第 1 层	√		√	√
筛选过程		√	√	√
第 2 层		√	√	√
第 3 层		√	√	√
精确定位				√

**实验 1** 为了检验眼睛粗定位算法以及粗筛选算法的有效性, 实验将全部 1 040 个样本作为输入, 先使用极小值区域进行眼睛粗定位, 再使用自然约束条件进行粗筛选, 并给出  $N$  个候选眼睛对。文献 [4]、[5] 认为当  $err < 0.25$  时即是定位正确, 这里也采纳这个标准, 前  $N(41)$  选正确率达 98.56%。

**实验 2** 为了单独检验 3.2 节和 3.3 节所描述的两个分类器对样本的筛选效果, 同时要消除实验 1 的错误累积影响, 实验仅将实验 1 定位正确的 1 025 个样本作为输入, 然后依次应用两个分类器得到筛选结果。两个分类器均随机使用样本的 80% 作为训练样本, 余下的 20% 作为测试样本。实验结果是首选正确率达 99.71%。

**实验 3** 为了检验粗定位算法和筛选算法的效果, 将全部 1 040 个样本作为输入, 先使用极小值区域进行眼睛粗定位, 再使用粗筛选算法和后两个分类器进行筛选。实验的首选正确率为 98.27%。

**实验 4** 为了检验整个算法的效果, 将全部 1 040 个样本作为输入, 先使用极小值区域进行眼睛粗定位, 再使用 3 个层次的筛选算法, 最后再应用眼睛精确定位算法进行微调。实验的首选正确率为 98.46%。

表 2 给出了这 4 个实验的正确率结果, 表 3 给

出了算法各步骤输出的眼睛对(或单只眼睛)数目的平均值。

表 2 4 个实验的实验结果 ( $err < 0.25$ )

Tab. 2 Results of four experiments ( $err < 0.25$ )

	样本数	正确数	正确率 (%)
实验 1	1 040	1 025	98.56
实验 2	1 025	1 022	99.71
实验 3	1 040	1 022	98.27
实验 4	1 040	1 024	98.46

表 3 算法各步骤的输出规模

Tab. 3 The output size of every step of the algorithm

眼睛对数目(对)	
粗定位	$N = 41$ , 即 41 只候选眼睛
第 1 层	< 16
筛选过程	第 2 层 第 3 层
第 2 层	3
第 3 层	1
精确定位	1

图 5 详细地给出了实验 3 和实验 4 的对比效果。从图中可以清楚地看出, 眼睛精确定位算法大幅度提高了整个算法的精度。

图 6 显示了算法各步骤的执行结果。

该算法还在自建的数据集上做了测试。该数据集由于是在实际条件下采集的, 所以光照条件、人脸倾斜角度、表情等都比较复杂。本文算法在该数据集上测试, 得到了 97.5% 的定位准确率, 一些定位结果如图 7 所示。

本文算法还与文献 [4] 的系统进行了比较。在文献 [6] 中,  $err < 0.25$  即可认为定位正确, 其算法在 JAFFE 数据库上的正确率为 97.18%。本文算法也在 JAFFE 数据库上做了测试, 在  $err < 0.10$  时的正确率即为 97.18%, 在  $err < 0.12$  时即可达到 100% 的定位准确率, 优于文献 [4] 的算法性能。

眼睛定位在人脸识别中用作基准点, 其精确度直接影响识别精度, 因此,  $err < 0.25$  的标准显得宽松。若采取  $err < 0.05$  的标准, 则本文算法的正确率为 87.05%; 若采取  $err < 0.10$  的标准, 则本文算法的正确率为 97.12%。在 BioID (<http://www.bioid.com>) 数据集(该数据集被认为是比较复杂<sup>[7]</sup>)上, 本文 62% 的首选正确率 ( $err < 0.05$ ) 高于文献 [7] 的 51% 的正确率。

本文的眼睛定位算法运行效率很高, 在 CPU 为

P4 2.4GHz, 内存为 512M 的 PC 机上, 处理一幅 CAS-PEAL 图像 (480 × 360) 时间仅为 0.25s, 处理一

幅 JAFFE 图像 (256 × 256) 的时间仅为 0.1s, 处理自建数据库的图像 (216 × 160) 的时间为 0.05s。

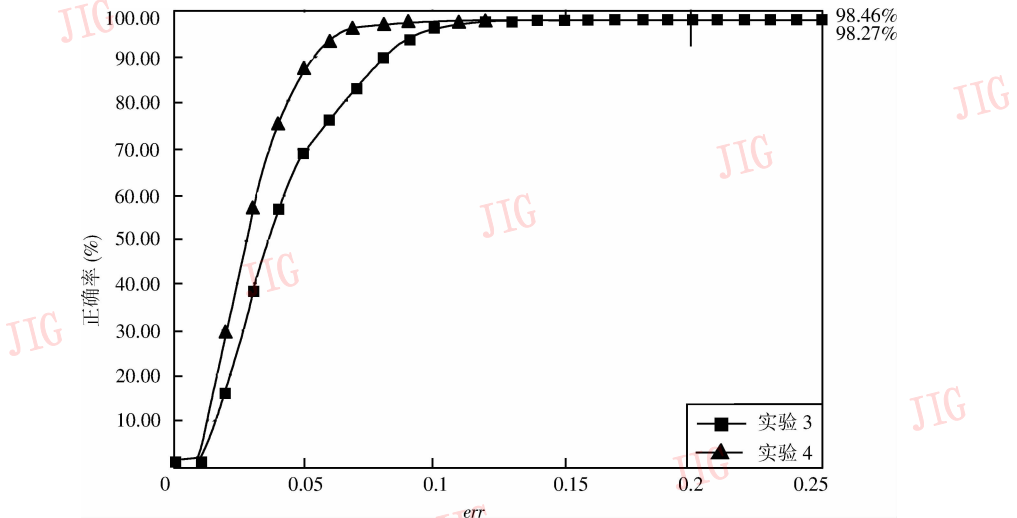


图5 实验3和实验4的对比结果

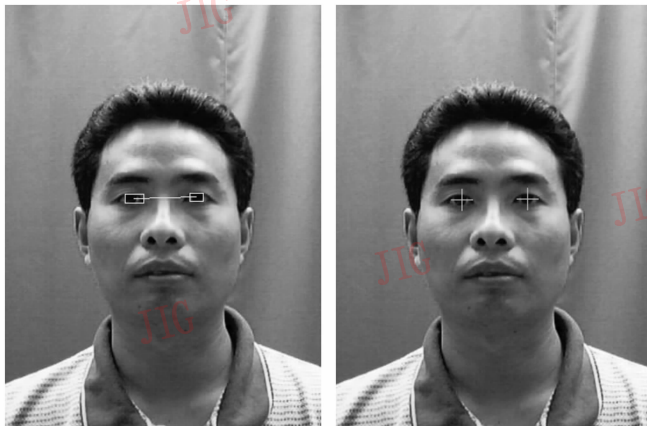
Fig. 5 Contradistinction from experiment No. 3 to No. 4



(a) 输入图像

(b) 使用极小值区域进行眼睛精确定位的结果

(c) 使用自然约束条件进行粗筛选的结果



(d) 使用两个分类器进行眼睛细筛选的结果

(e) 眼睛精确定位的结果

图6 算法执行过程示意图

Fig. 6 The execution of this algorithm



图 7 自建数据库的部分结果

Fig. 7 Some results on our own database

### 5.3 实验分析

从实验 1 到实验 4 的结果可以看出,眼睛粗定位算法的正确率对后面的筛选成功率以及眼睛精确定位算法的正确率的影响比较大,因此,使用极小值区域进行眼睛粗定位的方法是整个算法成功的基础,同时也是系统中下一步需要改进的地方。分析实验结果还可以得出,3 层筛选算法有效地剔除了错误的眼睛候选,保留了正确眼睛候选,使得整个系统在  $err < 0.25$  的条件下最终可以达到 98.46% 的正确率。与其他系统的比较也可以看出,本文算法是有效的。另外,从图 5 可以看出,眼睛精确定位算法对于校正眼睛的位置有很明显的作用。在  $err$  比较低的条件下仍然达到了很高的正确率,使得定位结果更接近于标定的真实位置。

## 6 结 论

针对灰度图像的眼睛定位问题,提出了一种基

于多层结构的快速眼睛定位算法。算法首先使用极小值区域进行眼睛粗定位,然后使用 3 级筛选算法去除错误的候选,挑选出唯一正确的候选,最后再用眼睛精确定位算法对眼睛位置进行校正,得到最终结果。本文算法可以在一定程度上消除年龄、性别、表情、姿态和光照的影响,显示出了良好的鲁棒性。

人脸识别及验证系统的研究和开发可以在本文眼睛定位结果的基础上进行,充分地利用眼睛位置可以大幅度提高人脸识别及验证的正确率。今后也将在这方面开展更多的工作。

### 参考文献 (References)

- 1 Gao Yong-sheng, Leung M K H. Face recognition using line edge map [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(6): 764 ~ 779.
- 2 Zhao W, Chellappa R, Phillips P J, et al. Face recognition: A literature survey [J]. ACM Computing Surveys, 2003, 35(4): 399 ~ 458.
- 3 Chellappa R, Wilson C L, Sirohey S. Human and machine recognition of faces: a survey [J]. Proceedings of the IEEE, 1995, 83(5): 705 ~ 741.

- 4 Zhou Z H, Geng X. Projection functions for eye detection [ J ]. Pattern Recognition, 2004, **37**(5): 1049 ~ 1056.
- 5 Geng X, Zhou Z, Chen S. Eye location based on hybrid projection function [ J ]. Journal of Software, 2003, **14**(8): 1394 ~ 1400.
- 6 Dieckmann U, Plankensteiner P, Wagner T. SESAM: A biometric person identification system using sensor fusion [ J ]. Pattern Recognition Letters, 1997, **18**(9): 827 ~ 833.
- 7 Hamouz M, Kittler J, Kamarainen J K, *et al.* Feature-based affine-invariant localization of faces [ J ]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, **27**(9): 1490 ~ 1495.
- 8 Kawaguchi T, Hidaka D, Rizon M. Detection of eyes from human faces by Hough transform and separability filter [ A ]. In: Proceedings of International Conference on Image Processing [ C ], Vancouver, BC, Canada, 2000: 49 ~ 52.
- 9 Viola P, Jones M J. Robust real-time object detection [ R ]. Cambridge, MA, USA: Cambridge Research Laboratory, 2001.
- 10 OpenCV Library [ CP/OL ]. <http://www.intel.com/technology/computing/opencv/index.htm>.
- 11 Jain A K. Fundamentals of Digital Image Processing [ M ], Upper Saddle River, New Jersey, USA: Prentice-Hall, 1989.
- 12 Matas J, Chum O, Urban M, *et al.* Robust wide-baseline stereo from maximally stable extremal regions [ J ]. Image and Vision Computing, 2004, **22**(10): 761 ~ 767.
- 13 Jančošek M. Feature Detection and Tentative Correspondence Estimation in Wide Baseline Stereo [ D ]. Bratislava, Slovakia: Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava, 2005.
- 14 Teuvo K. Learning vector quantization, The Handbook of Brain Theory and Neural Networks [ M ]. Cambridge, MA, USA: MIT Press, 1998: 537 ~ 540.
- 15 Gao Wen, Cao Bo, Shan Shi-guang, *et al.* The CAS-PEAL Large-Scale Chinese Face Database and Baseline Evaluations [ R/OL ]. <http://www.jdl.ac.cn/peal/files/TechReport4CAS-PEAL-R1.pdf>, 2004.
- 16 The JAFFE Database [ DB/OL ]. <http://www.kasrl.org/jaffe.html>.