

一种改进的 Adaboost 算法的人脸检测分类器

刘 侠, 李 苏, 李廷军

(哈尔滨理工大学 自动化学院, 黑龙江 哈尔滨 150080)

摘 要:针对人脸检测中单一分类器检测非常耗时,而且没有考虑到检测后提取标准人脸对后续工作(如识别)等问题,提出了运用 Adaboost 算法,采用正面人脸和人眼两个特征作为分类器,首先用 Cascade 算法筛选出输入图像中最有可能是人脸的区域,然后根据以正面人脸和人眼为特征的分类器检测人脸区域,并根据检测出来的人眼距离和角度对人脸进行精确的定位。同时针对实际的图像中人脸常常存在一定倾斜,从而影响后续的人脸的识别率这一问题,采取了根据人眼的位置对倾斜图像进行修正。实验结果表明改进的算法能够在保持一定运算速度的基础上取得准确的人脸,具有较好的实用性。

关键词:人脸检测;分类器;Adaboost 算法;人脸特征

中图分类号: TP391.4 **文献标识码:** A **文章编号:** 1009-3516(2009)02-0076-05

20世纪90年代以来,随着需要的剧增,人脸识别技术成为一个热门的研究话题。作为人脸识别系统的重要环节——人脸检测与定位也得到了众多研究者的重视。目前国外对人脸检测问题的研究很多^[1],比较著名的有 MIT、CMU 等。早期的人脸识别算法都是在假设已经得到了一个正面人脸或假设人脸很容易获得的前提下进行的,但是随着人脸分析应用范围的不断扩大和开发实际系统需求的不断提高,这种假设下的研究不再能满足需求^[2],因此人脸检测开始作为独立的研究内容发展起来^[3]。

1997年, Freund 和 Schapire^[4-5]提出了 Adaboost 算法。Adaboost 算法是在一系列矩形特征的基础上,通过 Boosting 算法由一系列“弱”分类器经过线性组合而形成的“强”分类器。这些简单的分类器是由提取的矩形特征组成的,脸部一些特征就能够由矩形特征简单地描绘,通常眼睛要比脸颊颜色更深;鼻梁两侧要比鼻梁颜色要深;嘴巴要比周围颜色更深。目前用于检测的矩形特征^[6-8]主要有三大类,虽然只有十几个矩形特征,但将这十几个特征用在一个系统中虽然会提高检测精度,但在训练分类器过程中所用的时间会成倍增加,检测速度会大受影响。实际应用中检测速度要求较严格,在本文中,我们采用了较为简单的矩形特征。虽然这些特征在灵活性上远远不如其他的可调控分类器,它的可控方向也仅仅是垂直或者水平的,但是这些以图像灰度值为自变量的矩形特征在边缘检测方面有着很好的表现,对人脸部的各种特征有很好的提取和编码能力。加上它计算效率上的绝对优势,这些足以弥补它在灵活性上的缺点。

1 人脸检测

1.1 训练样本集

文中运用了正面人脸检测同时也加入了对眼睛的检测,由于眼睛的特征要比脸部少得多,因此在进行远距离检测时会有一些误检,近距离检测时效果比较理想。

为训练单眼检测器,采用人工标定的方法从人脸图片中切割出单只眼睛图片。共得到 3 464 张人眼样本和 4 874 张非人眼样本作为训练样本集。将收集好的正例训练样本进行尺寸归一化,实验中取 $W = 20, H$

* 收稿日期:2008-10-26

基金项目:黑龙江省自然科学基金资助项目(F2007-13);黑龙江省教育厅科学技术研究资助项目(11531034)

作者简介:刘 侠(1975-),男,河北滦县人,副研究员,主要从事模式识别与智能系统研究。

E-mail:liuxia@hrbust.edu.cn

=12。正例样本集中图像包括单个人脸,主要来源是 ORL 人脸库和 PF01 人脸库。前者共 40 人,每人 10 幅,总计 400 幅;后者共 107 人,每人 17 幅,总计 1 819 幅。其中,前者未做人眼对齐,因此都须重新标定人眼位置;后者已做了人眼对齐,只需要直接取出其眼睛位置即可。在实际训练中,前者部分图像在提取人脸子窗口时,区域越界,因此没有参与训练,实际参与训练数为 340 幅。后者因为部分图像光影过于明显,可能会对训练结果产生负面影响,因而也没有参与训练,实际参与训练数为 1 427 幅。而非人脸样本主要从网上下载得到。

1.2 Adaboost 算法优化

本文应用的 Adaboost 单个简单分类器快速训练算法,是针对传统 Adaboost 算法存在的不足提出的一种直接求简单分类器错误率 ε_j 的方法,避免了迭代训练、统计概率分布等耗时的过程。

假设有 n 个训练样本 $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, $y_i = 0, 1$ 对应样本的负例和正例。在训练样本中共有 m 个负例样本, l 个正例样本。同时有一套权重 $w_i, 1 \leq i \leq n$ 对应于每个样本。

求解一个特征 f_j 的简单分类器 h_j , 是要确定 h_j 的阈值 θ_j 和偏置 p_j 。根据定义,简单分类器的错误率 ε_j 是 θ_j 和 p_j 的函数,记为 $\varepsilon_j(\theta_j, p_j)$ 。由于偏置 p_j 可以等价于 ± 1 两种情况,因此分别进行讨论:

$$1) \text{ 假设偏置 } p_j = 1, \text{ 根据 } h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

特征 f_j 的值小于阈值 θ_j 时为真,那么有:

$$\varepsilon_j(\theta_j, 1) = \sum_{i: h_j(x_i) \neq y_i} w_i = \sum_{\substack{i: f_j(x_i) > \theta_j \\ y_i = 1}} w_i + \sum_{\substack{i: f_j(x_i) \leq \theta_j \\ y_i = 0}} w_i = \sum_{\substack{i: f_j(x_i) > -\infty \\ y_i = 1}} w_i - \sum_{\substack{i: f_j(x_i) \leq \theta_j \\ y_i = 1}} w_i + \sum_{\substack{i: f_j(x_i) \leq \theta_j \\ y_i = 0}} w_i \quad (2)$$

$$\varepsilon_j(\theta_j + \Delta, 1) = \varepsilon_j(\theta_j, 1) - \sum_{\substack{i: \theta_j < f_j(x_i) \leq \theta_j + \Delta \\ y_i = 1}} w_i + \sum_{\substack{i: \theta_j < f_j(x_i) \leq \theta_j + \Delta \\ y_i = 0}} w_i \quad (3)$$

如果将所有训练样本的特征 f_j 的值从小到大进行排列,得到一个次序表 S 。 $S(1)$ 表示训练样本中特征 f_j 的值最小样本的序号, $f_j(x(S(1)))$ 为它相应的特征所得的值,如以该值为阈值则有:

$$\varepsilon_j(f_j(x)(S(1)), 1) = \sum_{y_i=1} w_i \quad (4)$$

对于特征 f_j 的值次小的训练样本 $x(S(2))$, 有:

$$\varepsilon_j(f_j(x)(S(2)), 1) = \varepsilon_j(f_j(x)(S(1)), 1) + \delta w(S(2)) \quad (5)$$

根据式(3)可知,当 $y(S(2)) = 1$ 时, $\delta = -1$; 当 $y(S(2)) = 0$ 时, $\delta = 1$, 以此类推得:

$$\varepsilon_j(f_j(x)(S(N+1)), 1) = \varepsilon_j(f_j(x)(S(N)), 1) + \delta w(S(N+1)) \quad (6)$$

利用递推公式(6),可以求出所有不同阈值下的 $\varepsilon_j(\theta, 1)$, 从中获得最小的 $\varepsilon_{j, \min}(\theta, 1)$, 此时的 θ 值定为 θ_1 , 即 $\theta_1 = \theta_j |_{\varepsilon_j = \varepsilon_{j, \min}(\theta, 1)}$, 为相应特征 j 分类器的最优阈值。

2) 假设偏置 $p_j = -1$, 那么可以求得同样的一组 $\varepsilon_j(\theta, 1)$, 其值和 $p_j = 1$ 的一组值是互补的, 即 $\varepsilon_j(\theta, 1) = 1 - \varepsilon_j(\theta, 1)$, 于是有 $\varepsilon_{j, \min}(\theta, -1) = 1 - \varepsilon_{j, \max}(\theta, 1)$, 此时的 θ 值定为 θ_2 , 即 $\theta_2 = \theta_j |_{\varepsilon_j = \varepsilon_{j, \min}(\theta, 1)}$ 。

因此只需要使用一次累加, 然后比较 $\varepsilon_{j, \min}(\theta, 1)$ 和 $\varepsilon_{j, \max}(\theta, 1)$, 就可以求出相应的该简单分类器最佳的阈值和偏置, 同时也可以得到相应的 ε_j 。

$$p_j = \begin{cases} +1 & \text{if } \varepsilon_{j, \min}(\theta, 1) < 1 - \varepsilon_{j, \max}(\theta, 1) \\ -1 & \text{otherwise} \end{cases} \quad (7)$$

1.3 减少特征数量

为了提高训练速度,除了算法效率外,影响运算次数的因素只有训练集数量和特征数量,而训练集数量不可能减少,这样会直接影响最后得到的强分类器效果;因此必须得减少特征数量才能减少训练时间。对于特征数量的减少,主要有 2 个途径:

1) 在特征生成阶段人为剔除。例如,完全不必考虑宽度为一个像素的矩形特征,因为显然用这样的特征计算特征值随机性太大,不可能寻找到合适的阈值。又因为训练集的人脸特征都集中在中部,边缘上的矩形特征的贡献不会特别大,因此可以适当减少边缘的矩形特征,例如不考虑涉及到边缘 1 像素的矩形特征,则矩形特征的数量可以进一步减少。

2) 在训练过程中动态地减少矩形特征。对于在训练中,阈值的选取对错误率降低成效不大的那些矩形特征,被认为是对人脸和非人脸区分能力很低的,可以设定一个错误率范围(逐轮改变),在每轮训练中按照

这个范围,对矩形特征进行排除。

1.4 基于 Cascade 算法的检测架构

训练好了分类器后,还需要考虑如何用此模板尽快地扫描整个图像。本文采用的是基于瀑布型算法架构(Cascade)。在一般的人脸检测中都需要对被检测图像从大到小做多层的缩放,即所谓的金字塔模型,然后对每一层的图像进行扫描检测,最终定位出人脸的位置。这种方法会有很多的计算成本浪费在那些非主体的背景上,从而大大降低了速度。而 Cascade 的想法就是用尽可能少的运算量把不是主体的背景快速筛选掉,留下更有可能是人脸的子窗口,进行后续更加复杂的计算。

本文采用人脸和眼 2 个特征设计分类器,因为在大的图片中以人脸作为分类器分离出人脸的错误率要比以人眼作为特征的分类器的错误率要低得多。在大的图片中以 Cascade 算法把不是主体的背景快速的筛选掉,留下更有可能是人脸的子窗口,提高后续检测的效率^[9]。在筛选后的图像区域中再以正面人脸为特征的分类器检测人脸区域并检测出人眼,再根据两眼的距离具体标出人脸。检测区域用椭圆标出。例如如图 1 中的(b)、(c),椭圆的短轴为两眼的距离,长轴为短轴的 1.5 倍。系统整个过程处理流程见图 2。

在图 1 中,图 1(a)表示原始人脸图,图 1(b)表示精确定位出的人脸及在此基础上检测出的人眼,图 1(c)表示最终精确检测出的人脸。



图 1 人脸检测流程

Fig. 1 Face Detection flow

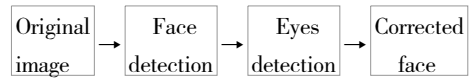


图 2 系统处理流程

Fig. 2 System disposal flow

1.5 对旋转图像的处理

在实际得到的图像中有可能存在倾斜,这会影响人脸的识别率,因此在识别之前的预处理中要对倾斜的图像进行修正,其步骤如下:

- 1) 首先得到人眼的位置, (x_1, y_1) 和 (x_2, y_2) ;
- 2) 计算两眼的中心位置 (x_3, y_3) ;
- 3) 计算其中一只眼睛和中心点的角度;
- 4) 根据角度的大小对图像进行旋转。

图 3 为图像旋转的结果,图 3(a)为原始图像,图 3(b)为检测出来的图像,图 3(c)是经过旋转后得到的最终图像。在图 3(b)中 (x_1, y_1) 的坐标为 $(81, 104)$, (x_2, y_2) 的坐标为 $(125, 990)$, 所以中心点 (x_3, y_3) 的坐标为 $(103, 101.50)$, 旋转角度为 7° 。



图 3 旋转实验结果

Fig. 3 Rotating results

2 实验结果及分析

以下图片从互联网下载得到,包括单人脸、多人脸在本系统上的检测结果,并与 CMU^[10]的结果进行了比对。因为与 CMU 进行比对是很方便的,网上提供了在线检测。其他研究机构虽然有各自的研究方法,但并未提供这样的平台,在进行比对时所用的图片不会统一。

图 4 是单人脸检测结果的比对,CMU 对图片结果的描述信息 Input: 400×300 , Primary filter 10.04 s, Features:0.57 s, Total:10.82 s, 1 faces found...;本文描述结果 Total: 1.359 s, 1 faces found... ,由此看出检测速度得到了提高。



图4 单个人脸检测结果与 CMU 检测结果的对比

Fig. 4 Comparative with the CMU test results

图5 是多人脸检测结果的对比,CMU 对图片结果的描述信息:Input: 560×376 , Primary filter 18.40 s, Features:3.82 s, Total:22.58 s, 5 faces found...;本文描述结果:Total: 3.164 s, 5 faces found...。

从对比结果看出 CMU 对人脸、眼睛及中垂线的检测极其准确,但算法比较耗时,图5耗时 10.82 s,图4耗时 22.58 s,而本文耗时分别为 1.359 s 和 3.164 s,分别提高了 7.96 倍和 7.14 倍。

为了测试系统的鲁棒性,文中对含旋转人脸的图像进行测试,测试中发现对于静态图像,可以检测人脸左右摆动不超过 40° 。虽然 AdaBoost 算法主要用于正面脸检测,但它对于侧面脸(包括面内旋转和面外旋转)也有较好的推广性。算法有较快的检测速度(具体的检测时间和图像大小与内容相关),并且由于在分类器检测图片的过程采用了逐步扩大待检测窗口而不是传统金字塔检测方式,有效地提高了检测图片的速度。



图5 多人脸检测结果对比

Fig. 5 Comparative with the multi-face detection results

图6 和图7 给出了复杂背景和简单背景情况下的实验结果。图6 复杂背景下检测结果:人脸数8、检测数5、漏检数3、错检数0。图7 简单背景下检测结果:人脸数9、检测数10、漏检数0、错检数1。



图6 复杂背景下检测结果

Fig. 6 Test result in complex background



图7 简单背景下检测结果

Fig. 7 Test result in simple background

3 结束语

本文采用人脸和人眼2个特征作为分类器,首先在大的图片中以 Cascade 算法把不是主体的背景快速的筛选掉,提高后续检测的效率。再根据两眼的距离进一步精确定位出人脸。相对于 CMU 检测方法而言,

大大提高了人脸检测的速度。同时对 Adaboost 算法进行了优化,并有效减少特征数量,大大提高了训练速度。本文中的方法能够取得准确的人脸,并根据人眼的位置对倾斜图像进行修正,得到了一个纯人脸样本,使以后的识别及跟踪的准确率大大提高。实验表明本算法切实可行,具有一定的实用性。

同时本文的方法也有待改进的方面,例如,在图像模糊并且人脸极度旋转的情况下还不能准确地检测,本方法的漏检和错检率还比较高,人眼的检测也不太理想。这主要是因为训练集数量太少,而且非人脸样本比重偏低。后续工作将致力于进一步改善这些问题。

参考文献:

- [1] 梁路宏,艾海舟,徐光祐,等.人脸检测研究综述[J].计算机学报,2002,25(5):449-458.
LIANG Luhong, AI Haizhou, XU Guangyou, et al. A Survey of Human Face Detection [J]. Chinese Journal of Computers, 2002, 25(5): 449-458. (in Chinese)
- [2] Belhumeur P, Hespanha J, Kriegman D, et al. Eigenfaces vs. Fisher Faces: Recognition Using Class Specific Linear Projection [J]. IEEE Trans Pattern Analysis and Machine Intelligence, 1997, (7): 711-720.
- [3] Yang M H, Kriegman D, Ahuja N. Detecting faces in Images: A Survey [J]. IEEE Trans Pattern Analysis and Machine Intelligence, 2002, 24(1): 34-58.
- [4] Freund F, Schapire R E. A Decision-theoretic Generalization of on Line Learning and an Application to Boosting [J]. Journal of Computer and System Sciences, 1997, 55(1): 119-139.
- [5] SCHAPIRE R E, FREUND Y, BARTLETT Y, et al. Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods [J]. Annals of Statistics, 1998, 26: 1651-1686.
- [6] Rainer Lienhart, Jochen Maydt. An Extended Set of Haar-like Features for Rapid Object Detection [J]. Intel Labs Intel Corporation, 2002, 1: 900-903.
- [7] Shen L, Bai L, Fairhurst M. Gabor Wavelets and General Discriminant Analysis for Face Identification and Verification [J]. Image and Vision Computing, 2007, 25(5): 553-563.
- [8] Xie X, Lam K M. Gabor-based Kernel PCA with Doubly Nonlinear Mapping for Face Recognition with A Single Face Image [J]. IEEE Trans on Image Process, 2006, 15(9): 2481-2492.
- [9] Rainer Lienhart, Alexander Kuranov, Vadim Pisarevsky. Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection [C]// Proceedings of 25th German Pattern Recognition Symposium, Magdeburg Germany, IEEE Press, 2003: 297-304.
- [10] Rowley H A, Baluja S, Kanade T. Neural Network-based Face Detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(1): 23-38.

(编辑:徐楠楠)

Research on Face Detection Classifier Based on An Improved Adaboost Algorithm

LIU Xia, LI Su, LI Ting-jun

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

Abstract: Aimed at the questions in face recognition by using single classifier, such as the time consuming of single classifier detection, and the significant of face image extracting after detection etc, this paper proposes a method based on Adaboost algorithm to handle it. The classifiers based on the facade and the eyes. First, we screen out the most probably facial area by the Cascade algorithm, then, we detect the facial area with the classifiers, finally, we will perform the precise positioning to the face image with the help of detected distance of eyes and angles, and as the facial pose side effect to the face recognition rate. We correct this problem according to the eyes positions. As the results of experiments indicate the proposed algorithm could detect the face image accurately from the whole image without sacrificing the algorithm running time, and it has practical definitely.

Key words: face detection; classifiers; Adaboost algorithm; face character