

北京大学
本科生毕业论文

基于 AdaBoost 算法的人脸检测

Face Detection Based on AdaBoost

姓 名：赵楠
学 号：00105029
院 系：物理学院物理学系

指导老师：查红彬 教授
导师单位：视觉与听觉信息处理国家重点实验室
信息科学技术学院智能科学系

二 五年六月



摘要 Abstract

人脸检测是人脸分析的首要环节，其处理的问题是确认图像（或影像）中是否存在人脸，如果存在则对人脸进行定位。人脸检测的应用领域相当广泛，是实现机器智能化的重要步骤之一。

AdaBoost 算法是 1995 年提出的一种快速人脸检测算法，是人脸检测领域里程碑式的进步，这种算法根据弱学习的反馈，适应性地调整假设的错误率，使在效率不降低的情况下，检测正确率得到了很大的提高。

本论文第一章和第二章简述了人脸检测的一般情况，第三章对一些人脸检测的经典方法进行了说明。

第四章讲述了 AdaBoost 算法的发展历史。从 PCA 学习模型到弱学习和强学习相互关系的论证，再到 Boosting 算法的最终提出，阐述了 **Adaptive Boosting** 算法的发展脉络。

第五章对影响 AdaBoost 人脸检测训练算法速度的至关重要的两方面：矩形特征和积分图的概念和理论进行了仔细的阐明。

第六章给出了 AdaBoost 的算法，并深入探讨了其中的一些关键问题——弱学习器的构造、选取等问题。

最后一章，用编写的实现了 AdaBoost 算法的 **FaDét** 程序，给出了相应的脸检测实验结果，并和 Viola 等人的结果做了比较。

关键词 Keywords

AdaBoost 方法、人脸检测、Boosting 方法、PCA 学习模型、弱学习



谨以此论文献给

A-腺嘌呤、T-胸腺嘧啶、G-鸟嘌呤、C-胞嘧啶、1和0

——智能的基本构件

和

开拓智能研究的伟大先驱者们

This dissertation is dedicated to

A, T, G, C, 1 and 0, the building blocks of intelligence.

and

to the pioneers uncovering the foundations of intelligence.



正文目录 **Contents**

摘要 ABSTRACT	II
正文目录 CONTENTS	IV
图目录 LIST OF FIGURES	VII
表目录 LIST OF TABLES	VIII
1 人脸检测	1
1.1 概念	1
1.2 难点与展望	2
1.3 人脸检测方法的性能评测	4
1.3.1 人脸图像数据库	4
1.3.2 性能评测	6
2 检测方法分类	8
2.1 基于知识的方法	8
2.2 特征不变量方法	9
2.3 模板匹配的方法	10
2.4 基于表象的方法	11
3 经典方法概述	12
3.1 神经网络 NEURAL NETWORK	12
3.2 特征脸 EIGENFACE	13
3.3 基于样本学习方法 EXAMPLE-BASED METHODS	14
3.4 支持向量机 SUPPORT VECTOR MACHINE (SVM)	15
3.5 隐马尔科夫模型 HIDDEN MARKOV MODEL (HMM)	15
4 ADABOOST 方法概述	16
4.1 引言	16



4.2 PAC 学习模型	16
4.2.1 概述.....	16
4.2.2 数学描述.....	17
4.3 弱学习与强学习	18
4.4 BOOSTING 方法	19
5 矩形特征与积分图	20
5.1 引言	20
5.2 矩形特征 RECTANGLE FEATURE.....	20
5.2.1 概述.....	20
5.2.2 特征模版.....	21
5.2.3 检测器内特征总数.....	22
5.2.3.1 子窗口内的条件矩形.....	22
5.2.3.2 条件矩形的数量.....	23
5.2.3.3 子窗口的特征矩形数量.....	23
5.2.3.4 结果.....	24
5.3 积分图 INTEGRAL IMAGE.....	25
5.3.1 概念.....	25
5.3.2 利用积分图计算矩形特征值.....	27
5.3.2.1 图像区域的积分图计算.....	27
5.3.2.2 矩形特征的特征值计算.....	28
6 ADABOOST 训练算法	30
6.1 训练基本算法.....	30
6.1.1 基本算法描述.....	30
6.1.2 基本算法流程图.....	32
6.2 弱分类器 WEAK CLASSIFIER.....	33
6.2.1 特征值 $f(x)$	33
6.2.2 阈值 q 、方向指示符 p	38
6.2.3 弱分类器的训练及选取.....	38



6.3 强分类器 STRONG CLASSIFIER.....	40
6.3.1 构成.....	40
6.3.2 错误率上限	40
7 程序实现及结果.....	43
7.1 样本集	43
7.2 训练难点及优化	44
7.2.1 计算成本.....	44
7.2.2 减少矩形特征的数量	44
7.2.3 样本预处理	45
7.3 检测结果.....	46
7.3.1 检测器	46
7.3.2 实验结果.....	47
7.3.2.1 实验对比.....	47
7.3.2.2 更多实验结果.....	49
7.3.3 结论.....	53
致谢 ACKNOWLEDGMENTS.....	54
参考文献 REFERENCES	54

Ver 0.76



图目录 List of Figures

图 1 人脸分析流程.....	2
图 2 人脸的遮挡、不同表情、图像的质量、旋转等等都会影响人脸检测.....	3
图 3 典型的正面人脸图像数据库中的人脸图像.....	5
图 4 左侧为测试图像，右侧为检测结果。不同的标准会导致不同的检测结果。	6
图 5 基于知识的人脸检测方法抽象出人脸的基本特征规则.....	9
图 6 一种人脸检测模板：这个模板由 16 个区域（图中灰色部分）和 23 种区域 关系（用箭头表示）组成.....	10
图 7 ROWLEY 的带有图像预处理的神经网络系统	13
图 8 人脸高斯簇和非人脸高斯簇.....	14
图 9 矩形特征在人脸上的特征匹配。上行是 24×24 子窗口内选出的矩形特征， 下行是子窗口检测到的与矩形特征的匹配。	21
图 10 计算 $m \times m$ 检测器内所有可能的矩形的数量。	22
图 11 “积分图”与积分的类比.....	25
图 12 坐标 $A(x, y)$ 的积分图定义为其左上角矩形所有像素之和（图中阴影部 分）。 $s(x, y)$ 为 $A(x, y)$ 及其 y 方向向上所有像素之和（图中粗黑竖线） ..	26
图 13 区域 D 的像素和可以用积分图计算为： $i^u_4 + i^u_1 - (i^u_2 + i^u_3)$	27
图 14 矩形特征的特征值计算，只与此特征端点的积分图有关	29
图 15 ADABOOST 训练强分类器算法流程图.....	32
图 16 所有矩形特征对所有图片的特征值平均数分布（横坐标是按特征值平均 值排序后的特征编号）	34
图 17 矩形特征 A 对人脸和非人脸图像的特征值分布（横坐标为排序表编号）， 这里看不出 A 有区分能力。	36
图 18 矩形特征 B 对人脸和非人脸图像的特征值分布（横坐标为排序表编号）， 这里 B 表现了很强的分辨能力。	37



图 19 训练并选取最佳分类器算法	39
图 20 训练用的非正面人脸图像	43
图 21 样本归一化前后对比	45
图 22 与 VIOLA 实验结果比较 1 : 总数 15/漏检 0/错检 5	47
图 23 与 VIOLA 实验结果比较 2 : 总数 7/漏检 1/错检 4	48
图 24 多人图像检测结果 : 总数 14/漏检 2/错检 1	49
图 25 多人图像检测结果 : 总数 6/漏检 1/错检 4	49
图 26 多人图像检测结果 : 总数 33/漏检 0/错检 4	50
图 27 多人图像检测结果 : 总数 33/漏检 5 /错检 14	50
图 28 多人图像检测结果 : 总数 3/漏检 0/错检 0	51
图 29 多人图像检测结果 : 总数 3/漏检 0/错检 0	51
图 30 多人图像检测结果 : 总数 2/漏检 0/错检 1	51
图 31 多人图像检测结果 : 总数 2/漏检 0/错检 0	52
图 32 单人图像检测结果 : 总数 1/漏检 0/错检 2	52
图 33 不含人脸图像检测结果 : 总数 0/漏检 0/错检 45	52

表目录 List of Tables

表 2 常用的人脸图像数据库.....	5
表 3 特征模板.....	22
表 4 特征模板的 (s,t) 条件	24
表 5 特征总数量的计算.....	24
表 6 不同子窗口大小内 , 特征的总数量.....	25
表 7 特征 A、B 对所有图像特征值的统计表	35



1 人脸检测

1.1 概念

长期以来，计算机就好像一个盲人，需要被动地接受由键盘、文件输入的信息，而不能主动从这个世界获取信息并自主处理。人们为了让计算机看到这个世界并主动从这个世界寻找信息，发展了机器视觉；为了让计算机自主处理和判断所得到的信息，发展了人工智能科学。人们梦想，终有一天，人机之间的交流可以像人与人之间的交流一样畅通和友好。

迄今为止，机器视觉的发展已经历了一个漫长的过程。经过研究者们的不懈努力，新的信息技术和媒体手段的出现，使得更加有效和友好的人机交互方式得到了发展，新型的人机交互将不再依赖传统的输入设备。而且，计算性价比的提高和视频获取成本的降低，使得计算机视觉系统能够向桌面级和嵌入式系统发展，这意味着计算机视觉系统能够安装在一切电子系统之中。相信在不久的将来，拥有高级视觉系统的智能电子产品会给我们的生活带来更大的方便。

计算机视觉要处理的一个重要内容，就是对人脸的视觉处理。人脸分析的相关研究希望用户的身份、状态和意图的信息能够从图象中提取出来，然后由计算依此做出反应（比如通过观察用户脸部表情来分析心情并进行相应反应）。由于人脸和脸部表情已经被心理学家、神经科学家和工程师们研究了多年，所以人脸和脸部表情识别的研究得到了更多的关注。

人脸分析（如图 1），主要包括人脸检测（face detection）和人脸识别（face recognition）两部分。最初人脸分析主要集中在人脸识别领域。与指纹、视网膜、虹膜、基因、掌纹等其它人体生物特征识别系统相比，人脸识别更加直接、友好，其使用者无心理障碍。人脸识别技术应用背景十分广泛，可用于公安系统刑侦破案的罪犯身份识别、身份证件及驾驶执照等证件验证、银行及海关的监控、自动门



卫系统、视频会议、机器人的智能化研究以及医学等方面。

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者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。人脸检测开始作为独立的研究内容发展起来了。

人脸检测是人脸分析的第一步，指对于任意一幅给定的图像，采用一定的策略对其进行搜索以确定其中是否含有人脸，如果含有则返回人脸的位置、大小和姿态。本篇论文关注的是人脸检测相关的技术。

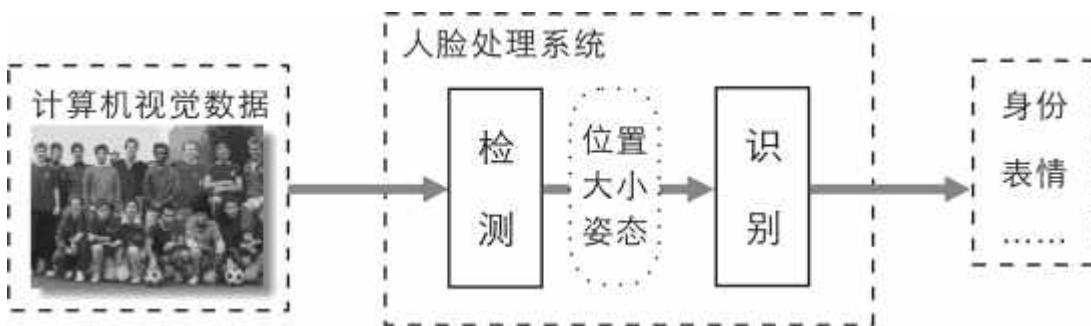


图 1 人脸分析流程

1.2 难点与展望

虽然人类可以毫不困难地检测到空间中的人脸，但计算机进行完全自动的人脸检测仍存在许多困难。它是一个复杂的具有挑战性的模式分类问题，其主要的难点来自两方面（参见图 2 的示例）：

一、由于人脸内在的变化所引起的：

1. 人脸具有相当复杂的细节变化，不同的外貌，如脸形、肤色等，不同的表情，如眼、嘴的开与闭等，甚至可能有器官的缺失；
2. 人脸的遮挡，如眼镜、头发和头部饰物以及其他外部物体等；

二、由于外在条件变化所引起的：

1. 由于成像角度的不同造成人脸的多姿态，如平面内旋转、深度旋转以及上下旋转，其中深度旋转影响较大；
2. 光照的影响，如图像中的亮度、对比度的变化和阴影等。



3. 图像的成像条件，如摄像设备的焦距、成像距离，图像获得的途径等等。



图 2 人脸的遮挡、不同表情、图像的质量、旋转等
等都会影响人脸检测

虽然人脸检测技术有了很大的发展，但是由于各种变化因素的影响，还有很多需要解决的问题：

1. 由于人脸模式的多样性，人脸图像的空间分布非常复杂，有限的样本集难以覆盖全部人脸图像子空间，在高维空间中建立准确的分布模型又很困难。因此，如何有效地描述人脸在子空间的分布值得研究。非线性变换和混合模型将是描述人脸特征分布的一个有效方法，也需要研究新的人脸描述方法以获得准确的人脸描述和与非人脸的区别信息。人脸局部和整体信息的相互结合能较好地描述人脸特征，有效地提取和组合局部与整体信息也是提高系统性能的一个途径。
2. 对于复杂背景的图像，如何有效地区分类似人脸的区域和真正的人脸区域很重要。神经网络、SVM、贝叶斯分类器等通常能获得好的分类结果，



研究新的分类方法是提高检测性能的另一个重要途径。也可以借鉴字符、语音、指纹等识别中比较成功的分类方法。

3. 目前的人脸检测算法还不能较好地处理任意姿态、光照和遮挡等变化条件，开发高性能的人脸检测方法仍是主要的研究方向。由于不同的人脸描述方式有不同的特点，不同的分类器产生的分类结果也不一样，多个特征融合和分类器融合的方法可以有效提高检测性能。
4. 由于大多数应用都是面向实时性处理，这要求人脸检测算法简单、快速。
5. 在压缩域，由于不需要完全解码，且可直接利用压缩域系数，因此大大提高了检测速度。对于视频，还能直接利用运动矢量等特征。因此，压缩域的人脸检测是面向对象的图像和视频检索技术的一个重要研究主题。

随着图像处理、模式识别、人工智能以及生物心理学的研究进展，人脸检测技术将会获得更大的发展。

1.3 人脸检测方法的性能评测

1.3.1 人脸图像数据库

人脸图像数据库可用于人脸检测的训练和测试。大部分的人脸检测方法都需要人脸训练图像数据，原先用于人脸识别实验的数据库现在都可以用来进行人脸检测的训练。通常人脸识别实验所采用的人脸库不大，一般仅包括 100 幅左右的人脸图像，如 MIT 库、Yale 库、CMU 库等都是小型库。

下表列出了常见的一些人脸识别图像数据库：

数据库	描述	位置
MIT	共有 16 个人的脸，每个人有 27 幅在不同光线、不同尺度、不同头部角度情况下的图像	ftp://whitechapel.media.mit.edu/pub/images
FERET	收集了大量的男性、女性人脸，每张图像包含了一张特定表情的人脸	http://www.nist.gov/humanid/feret

UMIST	分为 20 个主题 , 共 564 张图像 , 每个主题包括从侧面到正面的不同人脸姿态	http://images.ee.umist.ac.uk/danny/database.html
Univ. of Bern	30 个人的 300 张正面人脸(每人 10 张) 和 150 张侧面人脸 (每人 5 张)	ftp://ianftp.unibe.ch/pub/images/faceimages/
Yale	在不同光照条件下 , 戴眼镜的表情人脸	http://cvc.yale.edu
AT&T (Olivetti)	40 个主题 , 每个主题 10 张图像	http://www.uk.research.att.com
Harvard	在不同光照条件下的剪裁好的人脸图像	ftp://ftp.hrl.harvard.edu/pub/faces
M2VTS	一个包含不同图像序列的多态数据库	http://poseidon.csd.auth.gr/m2vts/index.html
Purdue AR	3276 张在不同光照条件下的表情和有遮挡的人脸图像	http://rvl1.ecn.purdue.edu/~aleix/aleix_face_db.html

表 1 常用的人脸图像数据库



图 3 典型的正面人脸图像数据库中的人脸图像



如图 3 , 就是 MIT 的一个典型的正面人脸数据库中的图像 , 图像的尺寸都为 20×20 。本论文的检测算法所用的训练图像即为这个数据库中的。由于不同研究任务的需要 , 必要时可以设计和建立具有特色的专用数据库。

1.3.2 性能评测

由于不同人脸库所取得的图像条件和质量不同 , 因此不同的识别程序之间很难进行比较。为促进人脸识别算法的深入研究和实用化 , 美国国防部发起了人脸识别技术(Face Recognition Technology , FERET)工程 , 它包括一个通用人脸库和一套通用测试标准 , 用于定期对各种人脸识别算法进行性能测试 , 其分析测试结果对未来的工作起到了一定的指导作用。



图 4 左侧为测试图像 , 右侧为检测结果。不同的标准会导致不同的检测结果。

对于人脸检测来说 , 一般关注两个指标 :

1. 人脸检测率 (Detection Rate): 在给定图像中 , 检测出来的人脸和人脸总数的比率。但是 , “ 人脸 ” 的定义并没有统一标准 , 例如 , 如果一个人脸被遮挡 , 那么究竟遮挡了多大部分后 , 就能判定剩余部分不再是一个 “ 人脸 ” ? ;
2. 错误检测数(False Detection): 有多少的非人脸被当成是人脸检测出来了。这个指标非常重要 , 有些检测算法可以给出甚至 100% 的检测率 ,



但是其错误检测的数量可能非常巨大。

一个理想化完美的人脸检测算法，应该有 100% 的人脸检测率和 0 错误检测数。现阶段也许只有人的大脑有这个能力。

虽然现在有很多检测算法，但是它们中的大部分还没有经过大量的图像数据集测试。而且，这些方法的实验结果大都是用不同的测试集得出的。

即使采用了相同的训练集和测试集，由于研究者对“检测成功”的看法不尽相同，使得检测率的标准不唯一，也会导致对不同检测算法评测的困难。如图 4，假设右图中的所有子图像都被检测器认为是人脸，那么一个宽泛的标准会认为检测到的这所有的人脸都是成功的，但是一个严格标准可能认为这些检测到的人脸有很多是错误的。

为了公平地比较和评价这些方法，也有必要像 FERET 一样定义一套标准的测试数据集和测试方法。



2 检测方法分类

可以将人脸检测的方法分为四类^[1]：

1. 基于知识的方法 (Knowledge-based)
2. 特征不变量方法 (Feature invariant)
3. 模板匹配的方法 (Template matching)
4. 基于表象的方法 (Appearance-based)

当然，有许多人脸检测的方法并不能简单地归于上述中的某一类，而是几类的组合。也可以将人脸检测的方法简单地分为两类^[2]：基于特征的和基于图像的。基于特征的方法指以某种特征为最小处理单元的方法；基于图像的方法指以图像中的像素为处理单元的方法。

2.1 基于知识的方法

这个方法将人类有关典型的脸的知识编码成一些规则。通常这些规则包括了脸部特征之间关系的知识。这个方法主要用于人脸的定位。

一些关于人脸的知识，可以归纳成下面几个简单的规则^[3]：

一、轮廓规则：

人脸的轮廓可近似地被看成一个椭圆，则人脸检测可以通过检测椭圆来完成。对任意一幅图像，首先进行边缘检测，并对细化后的边缘提取曲线特征，然后计算各曲线组合成人脸的评估函数来检测人脸。

二、器官分布规则：

虽然人脸因人而异，但都遵循一些普遍适用的规则，即五官分布的几何规则。检测图像中是否有人脸，即是否存在满足这些规则的图像块。这种方法一般是先对人脸的器官或器官的组合建立模板，然后检测图像中几个器官可能分布的位置，对这些位置点分别组合，用器官分布的知识规则进行筛选，从而找到可能存在人脸的区域。



在的脸。

三、对称性规则：

人脸具有一定的轴对称性，各器官也具有一定的对称性。

四、运动规则：

若输入图像为动态图像序列，则可以利用与人脸或人脸的器官相对于背景的运动来检测人脸，比如利用眨眼或说话等动作的探测来实现人脸与背景的分离。

基于知识的方法是一种自上而下的方式。它其中一个困难是如何将人类知识转化成为有效的规则：如果规则制定得太细，那么可能有许多人脸无法通过规则的验证；如果规则制定得太宽泛，那么可能许多非人脸会被误判为人脸。

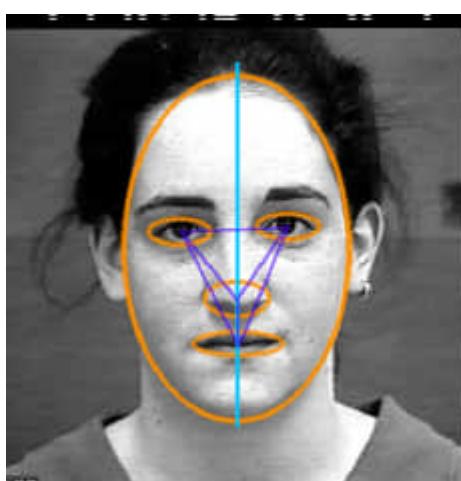


图 5 基于知识的人脸检测方法抽象出人脸的基本特征规则

2.2 特征不变量方法

这个方法的目标是寻找那些即使当姿势、视角和光线条件变化时仍然存在的结构特征，并利用这些特征来定位人脸。由于人类能够毫不费劲地“看到”在不同光线和姿态下的人脸和物体，因此研究人员认为有一个潜在的假设：存在一些关于人脸的不依赖于外在条件的属性或者特征。有许多方法就是按照这个潜在假设，首先去寻找这种脸部特征（通过大量样本学习的方法），然后用寻找到的特



征去检测人脸。对比基于知识的自上而下的方法，这种基于特征的方法是自下而上的。

人的肤色被证明是人脸检测的一个有效特征。人脸肤色聚集在颜色空间中一个较小的区域，因此可利用肤色特征能够有效地检测出图像中的人脸。利用肤色特征检测出的人脸区域可能不够准确，但如果在整个系统实现中作为人脸检测的粗定位环节，它具有直观、实现简单、快速等特点，可以为后面进一步进行精确定位创造良好的条件，以达到最优的系统性能。

本论文的 AdaBoost 方法就是基于人脸特征的方法。在后面的章节里面，可以通过 AdaBoost 来看到基于特征方法的主要处理过程。

2.3 模板匹配的方法

模板匹配法是一种经典的模式识别方法。其处理过程如下：

1. 进行预处理手动地预定义或者参数化一个标准人脸图案。预处理要做尺度归一化和灰度归一化的工作。最简单的模板将人脸看成椭圆，更复杂的人脸模板参见图 6；
2. 计算输入图像与标准人脸图案的相关值，这个相关值大都是独立计算脸部轮廓、眼镜、鼻子和嘴各自的匹配程度后得出的综合描述。对于图 6 的模板，则是按照划分的 16 个区域和这些区域间的 23 种关系来计算这个相关值；
3. 根据相关值和预先设定的阈值来确定图像中是否有人脸。

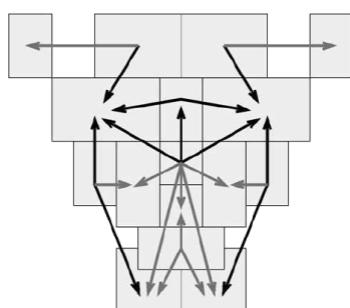


图 6 一种人脸检测模板：这个模板由 16 个区域（图中灰色部分）和 23 种区域关系（用箭头表示）组成



由于基于模板的方法比较成熟，因此其实现起来比较简单，但是这个方法对于人脸检测来说，效率并不高。

2.4 基于表象的方法

对比模板匹配，这个方法的模型是从一系列具有代表性脸部表观的训练图像学习而来，再将学习而成的模板用于人脸检测，而不像基于模板的方法，模板是由专家预先定义的，因此这也是一种自下而上的方法。现在许多人脸检测方法都是这种基于表象的方法。

一般而言，基于表象的方法利用统计分析和机器学习的技术来寻找人脸和非人脸图像的有关特性。学习而来的特性总结成分布模型或者判别函数，再利用这些分布模型或者判别函数来检测人脸。一般，为了计算的效率和检测的效益，都会先降低图像的维数。

许多基于表象的方法都可以在概率论的框架中理解。特征向量可以看成是随机变量 x ，这个随机变量被分类条件密度函数 $p(x / \text{faces})$ 和 $p(x / \text{nonfaces})$ 分别描述成人脸和非人脸。

图像中备选的人脸或者非人脸位置可以用贝叶斯判决规则 (Bayesian classification) 或者最大概似法 (maximum likelihood) 来判别。不幸地，简单地应用贝叶斯判决是不可行的，这是因为：

1. x 是高维的；
2. $p(x / \text{faces})$ 和 $p(x / \text{nonfaces})$ 是多态的 (multimodal)；
3. $p(x / \text{faces})$ 和 $p(x / \text{nonfaces})$ 的自然参数化形式 (natural parameterized forms) 还没有弄清楚。

因此，基于表象的方法中的许多工作都涉及到用经验来验证 $p(x / \text{faces})$ 和 $p(x / \text{nonfaces})$ 参数和非参数的近似的问题。

另一种方法利用了人脸和非人脸类的判别函数（比如决策面、分离超平面、阈函数等）：图像图案先被投射到低维或者高维空间，之后使用判别函数来进行分类。



3 经典方法概述

人脸检测的方法多不胜数，基于各种数学模型的方法估计有近百种。本节简单描述了用于人脸检测的几个经典方法，以便管中窥豹，这几个方法在人脸检测历史上都发挥过各自的作用。

3.1 神经网络 Neural Network

人工神经网络是一种应用类似于大脑神经突触联接的结构进行信息处理的数学模型。在这一模型中，大量的节点（或称“神经元”、“单元”）之间相互连接构成网络，即“神经网络”，以达到处理信息的目的。

神经网络可以充分逼近任意复杂的非线性关系，采用并行分布处理方法，使快速进行大量运算成为可能。神经网络具有自学习功能，在实现图像识别时，只需要先把许多不同的图像样本和对应的识别结果输入人工神经网络，网络就会通过自学习功能，慢慢学会识别类似的图像。

用神经网络进行人脸检测的好处是很容易训练一个用于检测人脸模式的系统。但是，缺点是网络结构需要大范围的调整（层数、结点数、学习速率等等）才能获得期望的性能。

在使用神经网络的人脸检测方法中，有里程碑式意义的工作由 Rowley 等人最早在 1996 年提出^[4]。Rowley 等采用局部连接的神经网络检测人脸。将每个区域划分成多个方形或矩形的子区域以便更好地描述人脸的局部特征，每个区域对应一个神经网络隐含单元。对神经网络在多个分级上的输出进行判决以降低误检率。如图 7。

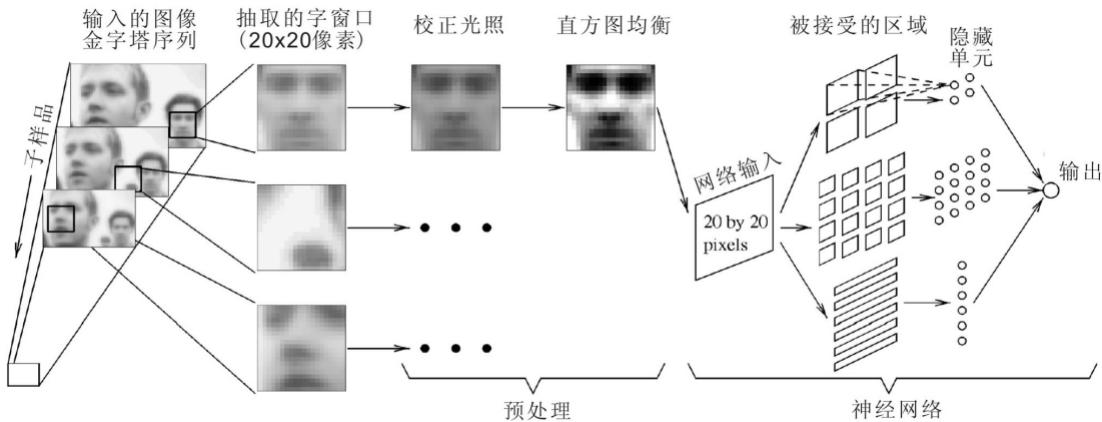


图 7 Rowley 的带有图像预处理的神经网络系统

3.2 特征脸 Eigenface

最早用特征向量来进行人脸检测的是 Kohonen^[5] ,他构建了一个简单的神经网络来实现规范化了的人脸图像的识别。这个神经网络用图像的自相关矩阵的特征向量近似计算了对人脸的描述。这些特征向量后来就发展成了特征脸方法。

特征脸是基于从主成分分析 (PCA)¹的一种人脸检测，它根据一组人脸训练样本构造主元子空间，检测时，将测试图像投影到主元子空间上,得到一组投影系数，再和各个已知的人脸图像模式比较，从而得到检测结果。

在传统特征脸的基础上，研究者注意到特征值大的特征人脸向量（特征脸）并不一定分类性能好，而因此发展了多种特征(子空间)选择方法，如 Belhumeur 的 FisherFace 方法^[6]等。事实上，特征脸方法是一种显式主元分析人脸建模的方法，而一些线性自联想、线性压缩型 BP 网则为隐式的主元分析方法，它们都是把人脸表示为一些向量的加权和，这些向量是训练集叉积阵的主特征向量。特征脸方法是一种简单、快速、实用的基于变换系数特征的算法，但由于它在本质上依赖于训练集和测试集图像的灰度相关性，所以还有着很大的局限性。

¹ 主成分分析(PCA, Principal Component Analysis) ,用于寻找原有的变量之间如何线性组合成为另一组变量，这组变量数目上少于原来的变量，但保持最多的原来的信息。是最小方差准则下最优的描述方法，它将输入数据的维数从图像空间的象素数减少为特征空间的维数。



3.3 基于样本学习方法 Example-Based Methods

MIT 的 Sung 和 Poggio 等提出了基于样本学习的方法^[7]。

他们采用 k-均值聚类方法在特征空间中建立 6 个“人脸”簇和包围“人脸”簇的 6 个“非人脸”簇，以使“人脸”模式与“非人脸”模式的边界更为清晰，如图 8。算法根据样本到各个簇中心的距离来训练一个多层次感知器进行分类，用来实现人脸检测。

值得强调的是，为了解决人脸检测中“非人脸”样本的选取问题，Sung 等人使用了“自举（bootstrap）”方法：首先建立一个仅使用“人脸”样本和少量“非人脸”样本训练的初始分类器对一组图像进行检测，把所有非人脸被错检为人脸的图像加入“非人脸”样本库；然后使用得到的“人脸”和“非人脸”样本训练构造新的分类器重新进行检测。以上过程不断循环，直到收集了足够的“非人脸”样本。这种非人脸的收集方法后来被许多研究者使用。

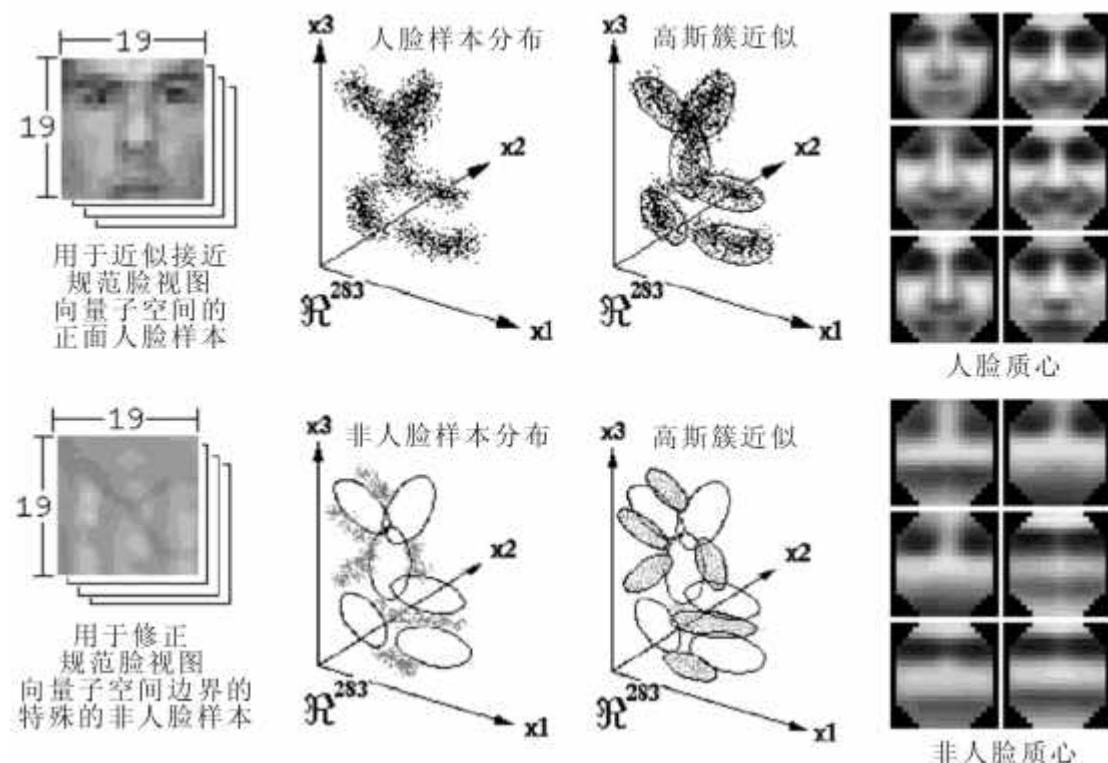


图 8 人脸高斯簇和非人脸高斯簇



3.4 支持向量机 Support Vector Machine (SVM)

支持向量机(Support Vector Machines SVM)是一种基于统计学习理论的模式识别方法，它在结构风险最小化基础上，为两种不同类别的样本数据找到一个最优分类面，可以说是一个受限二次规划问题。此方法由 Boser、Guyon、Vapnik 在 COLT-92 上首次提出。

最先将此方法应用于人脸检测的是 Osuna 等人^[8]，他们采用这个方法的检测效果和 Sung 等人^[7]的方法相似，然而速度却提高了 30 倍。

Heiselet 等人^[9]采用两级 SVM 的方法来检测人脸。根据一些预定义的特征点，从训练集中提取人脸和非人脸图像最有区别的多个局部区域。检测时，根据多个简单的线性 SVM 分别检测各个人脸特征区域。并用一个简单的线性 SVM 检测各个可能的人脸部分的几何组合是否符合人脸结构。相比于将整个人脸作为特征，该方法获得了更高的检测率。

3.5 隐马尔科夫模型 Hidden Markov Model (HMM)

隐马尔科夫模型(Hidden Markov Model，HMM)是用于描述信号统计特性的一组统计模型。HMM 使用马尔科夫链来模拟信号统计特征的变化，而这种变化又是间接地通过观察序列来描述的，因此，隐马尔科夫过程是一个双重的随机过程。在 HMM 中，节点表示状态，有向边表示状态之间的转移，一个状态可以具有特征空间中的任意特征，对同一特征，不同状态表现出这一特征的概率不同。由于 HMM 是一个统计模型，对于同一特征序列，可能会对应于许多状态序列，特征序列与状态序列之间的对应关系是非确定的。这种模型对于状态序列来说是隐的，故称为隐马尔科夫模型。

Nefian 等采用隐马尔可夫模型检测人脸^[10]。检测区域中的每个子区域采用主要的 Karhunen-Loëve 变换系数作为观察矢量，通过 Baum-Welch 算法和 Viterbi 分割算法获得 HMM 的模型参数，根据检测区域的观察序列的输出概率进行判决。



4 AdaBoost 方法概述

4.1 引言

PAC 学习模型的提出，计算学习理论得到了很大的发展，从而使“学习”有了计算上的判定方法。PAC 模型中提出了弱学习和强学习的概念，后来证明通过某种方法（后称为 Boosting）可以将弱学习提升为强学习——这使得训练机器学习来得更加方便。

1995 年，Freund 和 Schapire 提出了 AdaBoost 算法^[11]。AdaBoost 全称为 **Adaptive Boosting**，作者说取名叫作 AdaBoost 是因为这个算法和以前的 Boosting 算法都不同（原先的 Boosting 算法需要预先知道假设的错误率下限），它根据弱学习的反馈适应性地（adaptively）调整假设的错误率——也就是说，AdaBoost 算法不需要任何关于弱学习器性能的先验知识，加上它和原来 Boosting 算法的效率一样，因此可以非常容易地应用到实际问题中。

AdaBoost 算法提出后在机器学习领域受到了极大的关注，实验结果显示无论是应用于人造数据还是真实数据，AdaBoost 都能显著提高学习精度。

下面的章节将简述 AdaBoost 发明之前的 Boosting 算法发展过程。

4.2 PAC 学习模型

4.2.1 概述

可学习理论可以分为统计学习理论和计算学习理论两大部分^[12]。统计学习理论与经验过程有着密切的联系，而计算学习理论是概率理论中发展比较成熟的一个重要分支，它主要用于处理在实验的基础上进行的各种量的估计，研究当采样越来越多的时候，这些估计值是否收敛到未知的真值的问题，它的理论基础主要



是概率理论 ;计算学习理论主要研究如何构造有效的学习算法以及讨论学习算法的计算复杂性问题。

PAC (Probably Approximately Correct) 模型是计算学习理论中常用的模型 , 它是由Valiant于1984年首先提出来的^[13]。这篇论文认为 “ 学习 ” 是当明显清晰的过程或模式不存在时仍能获取知识的一种 “ 过程 ” , 并给出了一个从计算角度来获得这种 “ 过程 ” 的方法 , 这种方法包括 :(1) 适当信息收集机制的选择 ; (2) 学习的协定 ;(3) 对能在合理步骤内完成学习的概念的分类。虽然内在的算法复杂性限制了能够学习的概念的范围 , 论文仍然给出了一些有现实意义的 , 重要的 , 能够学习的概念例子。

PAC学习的实质就是在样本训练的基础上 , 使算法的输出以概率接近未知的目标概念。PAC学习模型是考虑样本复杂度及计算复杂度²的一个基本框架 , 成功的学习被定义为形式化的概率理论。

4.2.2 数学描述

下面简要描述PAC学习模型。给出 :

- 1、 \mathcal{X} 为样本空间 , 包含所有可以用于学习的样本集合 ;
- 2、 \mathcal{C} 为概念空间 , 包含所有可以选取的目标概念 T ;
- 3、 \mathcal{V} 为分类集合 , 其值为目标概念的所有分类 $\{v_1, \dots, v_k\}$ 。最简单的情况为二值 , $\mathcal{V}=\{0,1\}$;
- 4、 \mathcal{H} 为假设空间 , 包含算法所输出的所有假设 $H_m(T, x)$

学习器 L 的目的是找到目标概念的一个假设 , 使其能对每个样本进行分类。我们按照某种固定的(可能未知的)分布 $P(x)$ 独立抽取样本 x_1, \dots, x_m , L 返回 $h_T(x_i)$ 的值 : $h_T(x_i) \in \mathcal{V}$ 为 T 的指示函数 , 表示 L 对 x_i 的分类。

于是可以获得一组数据 :

²样本复杂度(sample complexity)指学习器收敛到成功假设时至少所需的训练样本数。

计算复杂度(computational complexity)指学习器收敛到成功假设时所需的计算量。



$$[(x_1, h_T(x_1)), \dots, (x_m, h_T(x_m))] \in [X \times V]^m$$

构造适当的算法 $\{A_m\}$, A_m 为到概念空间的映射 $A_m : [X \times V]^m \rightarrow C$, 并定义 :

$$H_m(T, x) = A_m((x_1, h_T(x_1)), \dots, (x_m, h_T(x_m)))$$

则 $H_m(T, x)$ 就是目标概念 T , 对样本 x_1, \dots, x_m 的一个假设。

我们希望能够找到一个对所有样本都正确的假设 , 在实际学习中 , 这是不可能的。如果学习器 L 最终将以 $(1 - d)$ 的概率 (d 称为假设的置信度) 输出一个假设 $h \in H$, 而且随机样本被错误分类的概率小于假设错误率 e , 我们就认为这个假设为成功假设。

如果学习器 L 只需要多项式 $p(m, 1/e, 1/d)$ 个样本以及在多项式

$p(m, 1/e, 1/d)$ 时间内就可以获得一个成功假设 , 那么称 L 为可PAC学习的。

因为PAC模型不要求学习器输出零错误率的假设 , 而只要求其错误率被限定在某常数 e 的范围内 (e 可以任意小) ; 同时也不要求学习器对所有的随机抽取的样本集合都能成功 , 只要其失败的概率被限定在某个常数 d 的范围内 (d 也可取任意小) 即可 , 所以学习器将学习到一个可能近似正确的假设。

4.3 弱学习与强学习

如何根据观测数据来学习并得到精确的假设是机器学习领域中人们非常关注的一个问题 , 机器学习的一个重要目标就是对新的样本尽可能给出精确的估计。

随机猜测一个是否的问题 , 将会有 50% 的正确率。如果一个假设能够稍微地提高猜测正确的概率 , 那么这个假设就是弱学习算法 , 得到这个算法的过程称为弱学习。可以使用半自动化的办法为好几个任务构造弱学习算法 , 构造过程需要数量巨大的假设集合 , 这个假设集合是基于某些简单规则的组合和对样本集的性能评估而生成的。如果一个假设能够显著地提高猜测正确的概率 , 那么这个假设就称为强学习。



生成只比随机猜测好一点的弱学习算法很容易，但是构造一个强学习算法却是一件相当困难的事情。Kearns^[14]提出了弱学习算法与强学习算法间的等价问题——是否能把弱学习算法转化为强学习算法。如果两者等价，则只需要找到一个弱学习算法就可以直接将其提升为强学习算法。Kearns 和 Valiant 证明^[15]：只要有足够的数据，弱学习算法就能通过集成的方式生成任意高精度的假设（强学习方法）。

4.4 Boosting 方法

Boosting 原意为提升、加强。现在一般指的是将弱学习算法提升为强学习算法的一类算法。Boosting 算法是在 Kearns 和 Valiant 证明^[15]后才真正成熟起来的。

1990 年，Schapire 最先构造出一种多项式级的算法^[16]，即最初的 Boosting 算法。这种算法可以将弱分类规则转化成强分类规则。一年后，Freund 提出了一种效率更高的 Boosting 算法^[17]。1993 年，Drucker 和 Schapire 第一次以神经网络作为弱学习器，应用 Boosting 算法来解决实际的 OCR 问题^[18]。

Boosting 算法在分类、建模、图像分割、数据挖掘等领域均已得到简单而有效的应用。

1995 年，Freund 和 Schapire 提出的 AdaBoost，是对 Boosting 算法的一大提高。下面章节将逐步具体说明 AdaBoost 算法。



5 矩形特征与积分图

5.1 引言

本章节描述了对 AdaBoost 人脸检测训练算法速度很重要的两方面，特征的选取和特征值的计算。

将矩形作为人脸检测的特征向量，称为矩形特征。本算法选取了最简单的 5 个矩形特征模板进行训练，以得到一套用于人脸检测的最适合的矩形特征，事实证明，这种特征选取方法的训练速度虽然不快，但是检测效率很高。

Viola 提出将积分图(integral image)应用到特征值的计算之中^[19]。积分图的引用，可以只对图像进行一次遍历计算，就能在用常量时间完成每个特征值的计算，这使得训练和检测的速度大大提升。

5.2 矩形特征 Rectangle Feature

5.2.1 概述

在给定有限的数据情况下，基于特征的检测能够编码特定区域的状态，而且基于特征的系统比基于象素的系统要快得多。

矩形特征对一些简单的图形结构，比如边缘、线段，比较敏感，但是其只能描述特定走向（水平、垂直、对角）的结构，因此比较粗略。如图 9，脸部一些特征能够由矩形特征简单地描绘，例如，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。

对于一个 24×24 检测器，其内的矩形特征数量超过 160,000 个，必须通过特定算法甄选合适的矩形特征，并将其组合成强分类器才能检测人脸。

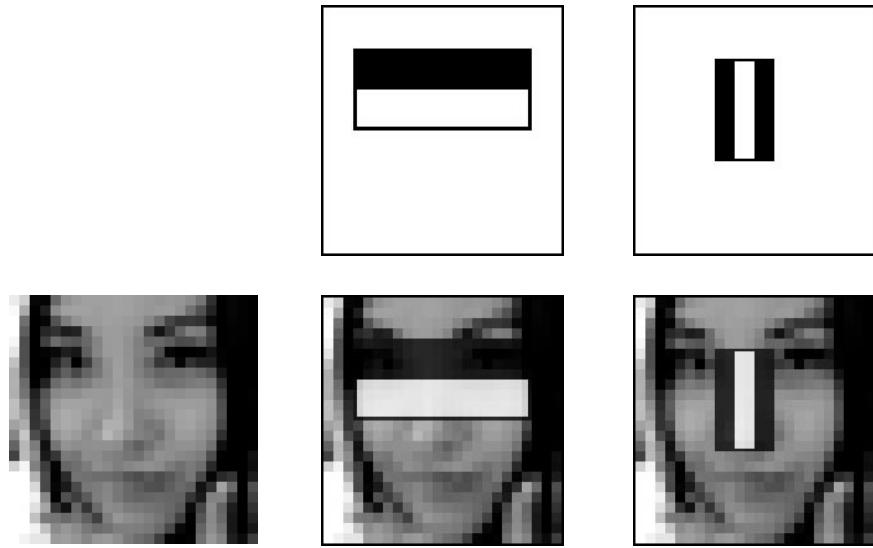
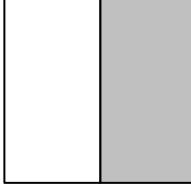
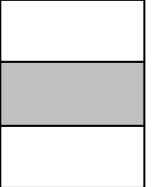
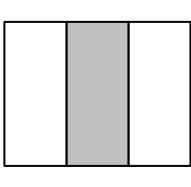


图 9 矩形特征在脸上的特征匹配。上行是 24×24 子窗口内选出的矩形特征，下行是子窗口检测到的与矩形特征的匹配。

5.2.2 特征模版

我们将使用简单矩形组合作为我们的特征模板。这类特征模板都是由两个或多个全等的矩形相邻组合而成，特征模板内有白色和黑色两种矩形（定义左上角的为白色，然后依次交错），并将此特征模板的特征值定义为白色矩形像素和减去黑色矩形像素和。

最简单的 5 个特征模板：

名称（模板号）	特征模板	
边缘特征 (1)(2)		
线性特征 (3)(4)		



特定方向特征 (5)			
---------------	--	--	--

表 2 特征模板

5.2.3 检测器内特征总数³

特征模板可以在子窗口内以“任意”尺寸“任意”放置，每一种形态称为一个特征。找出子窗口所有特征，是进行弱分类训练的基础。

5.2.3.1 子窗口内的条件矩形

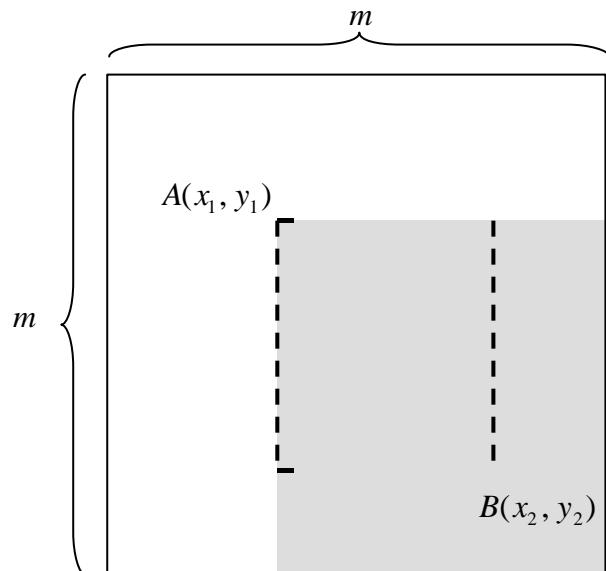


图 10 计算 $m \times m$ 检测器内所有可能的矩形的数量。

以 $m \times m$ 像素分辨率的检测器为例，其内部存在的满足特定条件的所有矩形的总数可以这样计算：

对于 $m \times m$ 子窗口，我们只需要确定了矩形左上顶点 $A(x_1, y_1)$ 和右下顶点 $B(x_2, y_2)$ ，即可以确定一个矩形；如果这个矩形还必须满足下面两个条件（称为 (s, t) 条件，满足 (s, t) 条件的矩形称为条件矩形）：

³ Paul Viola 和 Michael J. Jones 在其关于 boosted cascade 的同一篇论文的三个版本中(参见[19][20][21])，对 24×24 检测器中所含的矩形特征数量的描述各不相同(分别为 160,000、180,000、45,396)。这一节推导出了矩形特征数量的计算公式。



- 1) x 方向边长必须能被自然数 s 整除 (能均等分成 s 段);
 2) y 方向边长必须能被自然数 t 整除 (能均等分成 t 段);
 则, 这个矩形的最小尺寸为 $s \times t$ 或 $t \times s$, 最大尺寸为 $[m/s] \cdot s \times [m/t] \cdot t$ 或 $[m/t] \cdot t \times [m/s] \cdot s$; 其中 [] 为取整运算符。

5.2.3.2 条件矩形的数量

我们通过下面两步就可以定位一个满足条件的矩形:

- 1) 确定 $A(x_1, y_1)$: $x_1 \in \{1, 2, \dots, m-s, m-s+1\}$, $y_1 \in \{1, 2, \dots, m-t, m-t+1\}$;
- 2) 确定 A 点后, B 点只能在图 10 中阴影内 (包括边缘) 取值, 因此有:

$$x_2 \in X = \{x_1 + s - 1, x_1 + 2 \cdot s - 1, \dots, x_1 + (p-1) \cdot s - 1, x_1 + p \cdot s - 1\},$$

$$y_2 \in Y = \{y_1 + t - 1, y_1 + 2 \cdot t - 1, \dots, y_1 + (q-1) \cdot t - 1, y_1 + q \cdot t - 1\},$$

其中 $p = \left[\frac{m-x_1+1}{s} \right]$, $q = \left[\frac{m-y_1+1}{t} \right]$ 。并且知道: $|X| = p$, $|Y| = q$ 。

由上分析可知, 在 $m \times m$ 子窗口中, 满足 (s, t) 条件的所有矩形的数量为:

$$\begin{aligned} \Omega_{(s,t)}^m &= \sum_{x_1=1}^{m-s+1} \sum_{y_1=1}^{m-t+1} p \cdot q \\ &= \sum_{x_1=1}^{m-s+1} \sum_{y_1=1}^{m-t+1} \left[\frac{m-x_1+1}{s} \right] \cdot \left[\frac{m-y_1+1}{t} \right] \\ &= \sum_{x_1=1}^{m-s+1} \left[\frac{m-x_1+1}{s} \right] \cdot \sum_{y_1=1}^{m-t+1} \left[\frac{m-y_1+1}{t} \right] \\ &= \left(\left[\frac{m}{s} \right] + \left[\frac{m-1}{s} \right] + \dots + \left[\frac{s+1}{s} \right] + 1 \right) \cdot \left(\left[\frac{m}{t} \right] + \left[\frac{m-1}{t} \right] + \dots + \left[\frac{t+1}{t} \right] + 1 \right) \end{aligned}$$

5.2.3.3 子窗口的特征矩形数量

实际上, (s, t) 条件描述了矩形特征的特征, 下面列出了不同矩形特征对应的 (s, t) 条件:



特征模板	1、	2、	3、	4、	5、
(s, t)条件	(1, 2)	(2, 1)	(1, 3)	(3, 1)	(2, 2)

表 3 特征模板的 (s, t) 条件

所以 $m \times m$ 子窗口中所有 5 种特征模板的特征总数量 Ω^m 就是分别满足 5 个 (s, t) 条件的矩形特征的数量的总和，即：

$$\Omega^m = \Omega_{(1,2)}^m + \Omega_{(2,1)}^m + \Omega_{(1,3)}^m + \Omega_{(3,1)}^m + \Omega_{(2,2)}^m$$

特别地，由于特征模板 1 和 2、3 和 4 具有旋转对称性，则可以进一步简化为：

$$\Omega^m = 2 \times \Omega_{(1,2)}^m + 2 \cdot \Omega_{(1,3)}^m + \Omega_{(2,2)}^m$$

5.2.3.4 结果

下面以 24×24 子窗口为例，具体计算其特征总数量：

特征模板	数量
1、	$2 \times \Omega_{(1,2)}^{24} = 2 \times \left(\left[\frac{24}{1} \right] + \left[\frac{23}{1} \right] + \dots + \left[\frac{2}{1} \right] + 1 \right) \times \left(\left[\frac{24}{2} \right] + \left[\frac{23}{2} \right] + \dots + \left[\frac{3}{2} \right] + 1 \right)$ $= 2 \times (24 + 23 + \dots + 2 + 1) \times (12 + 11 + 11 + \dots + 2 + 1 + 1)$ $= 2 \times 300 \times 144 = 2 \times 43,200$ $= 86,400$
2、	$2 \times \Omega_{(1,3)}^{24} = 2 \times \left(\left[\frac{24}{1} \right] + \left[\frac{23}{1} \right] + \dots + \left[\frac{2}{1} \right] + 1 \right) \times \left(\left[\frac{24}{3} \right] + \left[\frac{23}{3} \right] + \dots + \left[\frac{4}{3} \right] + 1 \right)$ $= 2 \times (24 + 23 + \dots + 2 + 1) \times (8 + 7 + 7 + \dots + 1 + 1 + 1)$ $= 2 \times 300 \times 92 = 2 \times 27,600$ $= 55,200$
5、	$\Omega_{(2,2)}^{24} = \left(\left[\frac{24}{2} \right] + \left[\frac{23}{2} \right] + \dots + \left[\frac{3}{2} \right] + 1 \right) \times \left(\left[\frac{24}{2} \right] + \left[\frac{23}{2} \right] + \dots + \left[\frac{3}{2} \right] + 1 \right)$ $= (12 + 11 + 11 + \dots + 2 + 1 + 1) \times (12 + 11 + 11 + \dots + 2 + 1 + 1)$ $= 144 \times 144$ $= 20,736$
总数	$\Omega^{24} = 162,336$

表 4 特征总数量的计算



下面列出了，在不同子窗口大小内，特征的总数量：

窗口大小	36×36	30×30	24×24	20×20	16×16
特征数量	816,264	394,725	162,336	78,460	32,384

表 5 不同子窗口大小内，特征的总数量

5.3 积分图 Integral Image

5.3.1 概念

一个简单的微积分类比：如果我们要经常计算 $\int_a^b f(x) dx$ ，那我们会先计算 $F(x) = \int f(x) dx$ ，那么 $\int_a^b f(x) dx = F(b) - F(a)$ 。见图 11。积分图的含义与此类似。

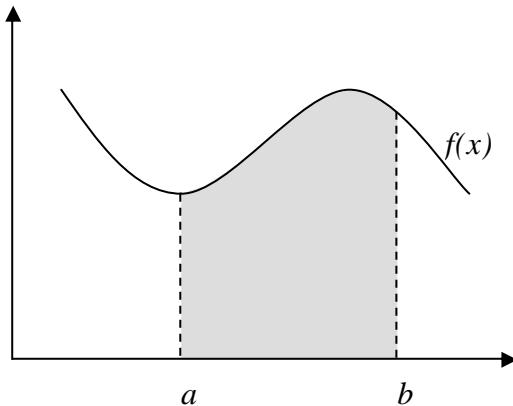


图 11 “积分图”与积分的类比

只需要对每个像素进行少量的计算工作，就能得到一幅图像的“积分图”。
“积分图”能够在多种尺度下，使用相同的时间来计算不同的特征，因此大大提高了检测速度。

对于图像内一点 $A(x, y)$ ，定义其积分图 $ii(x, y)$ 为（如图 12）：



$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') ,$$

其中 $i(x', y')$ 为点 (x', y') 处的“原始图”，是此点的颜色值；对于灰度图象，其值为 0~255。对于彩色图像，可以先按照人臉色彩空间将其转化为灰度取值。

$ii(x, y)$ 也可以通过下式迭代求出：

$$\begin{aligned} ii(x, y) &= ii(x - 1, y) + s(x, y) \\ s(x, y) &= s(x, y - 1) + i(x, y) \end{aligned}$$

其中 $s(x, y)$ 为点 (x, y) 及其 y 方向上所有原始图像之和（如图 12），称为“列积分和”，可以定义为：

$$s(x, y) = \sum_{y' \leq y} i(x, y')$$

并定义 $s(x, 0) = 0$, $ii(0, y) = 0$

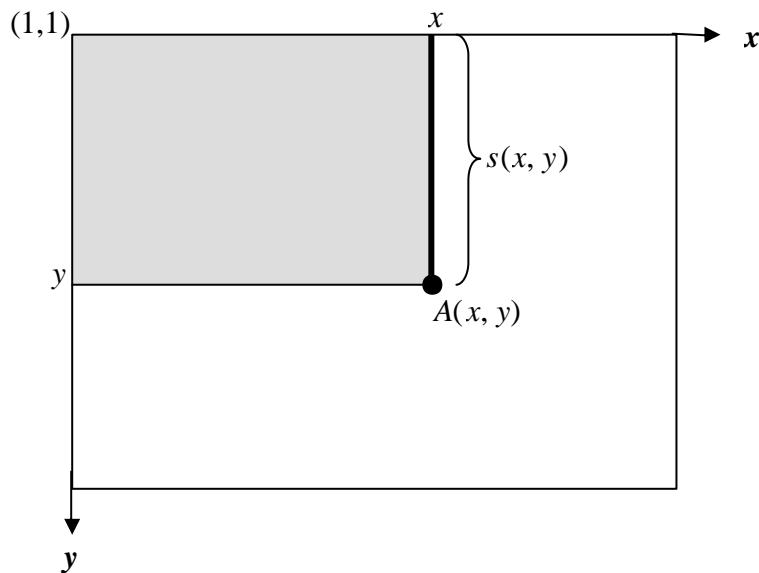


图 12 坐标 $A(x, y)$ 的积分图定义为其左上角矩形所有像素

之和（图中阴影部分）， $s(x, y)$ 为 $A(x, y)$ 及其 y 方向上所有像素之和（图中粗黑竖线）



设图像大小为 $m \times n$ ，则积分图矩阵（图像上所有像素的积分图）为：

$$\left(\begin{array}{cccc} ii(1,1) = ii(0,1) + s(1,1) & ii(2,1) = ii(1,1) + s(2,1) & ii(m-1,1) = ii(m-2,1) + s(m-1,1) & ii(m,1) = ii(m-1,1) + s(m,1) \\ s(1,1) = s(1,0) + i(1,1) & s(2,1) = s(2,0) + i(2,1) & \cdots & s(m,1) = s(m,0) + i(m,1) \\ ii(1,2) = ii(0,2) + s(1,2) & & & ii(m,2) = ii(m-1,2) + s(m,2) \\ s(1,2) = s(1,1) + i(1,2) & & & s(m,2) = s(m,1) + i(m,2) \\ \vdots & & \ddots & \vdots \\ ii(1,n-1) = ii(0,n-1) + s(1,n-1) & & & ii(m,n-1) = ii(m-1,n-1) + s(m,n-1) \\ s(1,n-1) = s(1,n-2) + i(1,n-1) & & & s(m,n-1) = s(m,n-2) + i(m,n-1) \\ ii(1,n) = ii(0,n) + s(1,n) & ii(1,n-1) = ii(0,n-1) + s(1,n-1) & ii(m-1,n) = ii(m-2,n) + s(m-1,n) & ii(m,n) = ii(m-1,n) + s(m,n) \\ s(1,n) = s(1,n-1) + i(1,n) & s(1,n-1) = s(1,n-2) + i(1,n-1) & \cdots & s(m,n) = s(m,n-1) + i(m,n) \end{array} \right)$$

可见，只需要遍历图像一次，迭代 $m \times n \times 2$ 次，即可以得到整个积分图矩阵。

5.3.2 利用积分图计算矩形特征值

5.3.2.1 图像区域的积分图计算

如图 13，区域 D 的像素值，可以利用 1、2、3、4 点的积分图来计算。

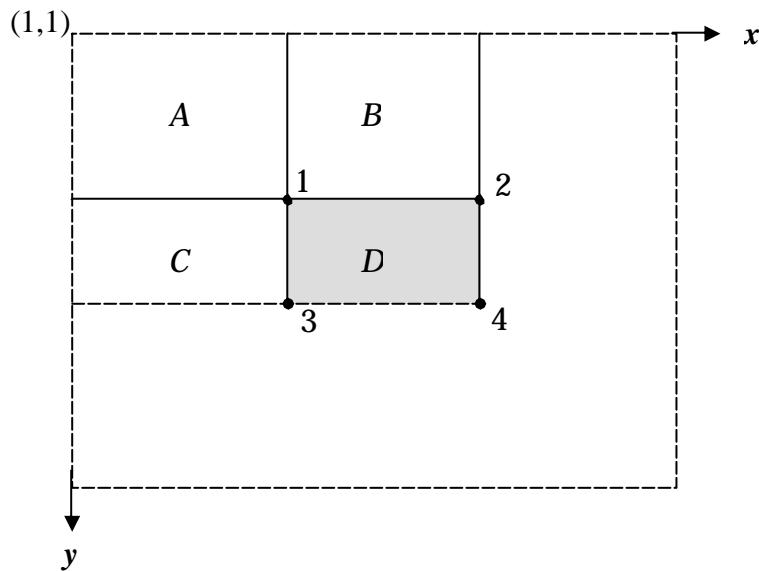


图 13 区域 D 的像素和可以用积分图计算为： $ii_4 + ii_1 - (ii_2 + ii_3)$

因为：

ii_1 = 区域 A 的像素值



$$ii_2 = \text{区域 A 的像素值} + \text{区域 B 的像素值}$$

$$ii_3 = \text{区域 A 的像素值} + \text{区域 C 的像素值}$$

$$ii_4 = \text{区域 A 的像素值} + \text{区域 B 的像素值} + \text{区域 C 的像素值} + \text{区域 D 的像素值}$$

所以解上述方程就可以得到：

$$\text{区域 D 的像素值} = ii_4 + ii_1 - (ii_2 + ii_3)$$

5.3.2.2 矩形特征的特征值计算

以特征模板 1 为例，如图 14。此特征模板的特征值为（按照 5.2.2 节中的定义）：

$$\text{区域 A 的像素值} - \text{区域 B 的像素值}$$

由刚才的证明可知：

$$\text{区域 A 的像素值} = ii_4 + ii_1 - (ii_2 + ii_3)$$

$$\text{区域 B 的像素值} = ii_6 + ii_3 - (ii_4 + ii_5)$$

所以此特征模板的特征值为：

$$(ii_4 - ii_3) - (ii_2 - ii_1) + (ii_4 - ii_3) - (ii_6 - ii_5)$$

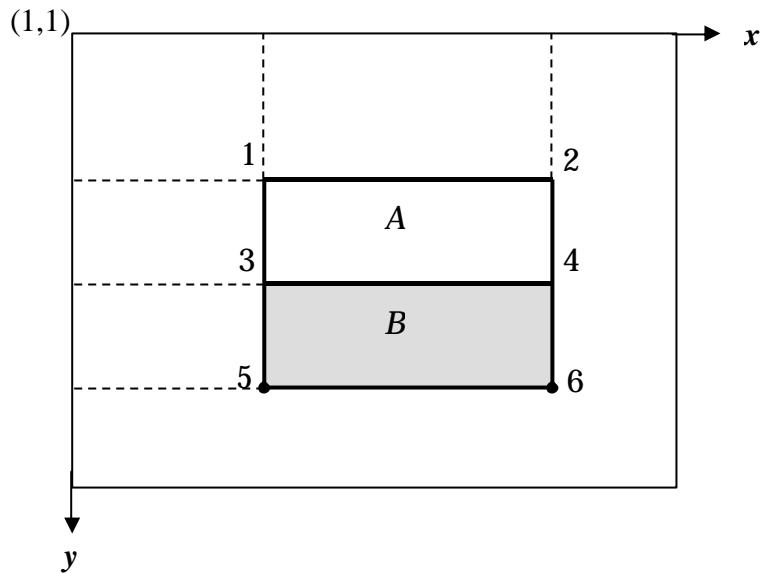


图 14 矩形特征的特征值计算，只与此特征端点的积分图有关

由此可见，矩形特征的特征值计算，只与此特征端点的积分图有关，而与图像坐标值无关。因此，不管此矩形特征的尺度如何，特征值的计算所耗费的时间都是常量（**time constant**），而且都只是简单的加减运算。正因如此，积分图的引入，大大地提高了检测的速度。

其他的矩形特征的特征值计算依此类推。



6 AdaBoost 训练算法

6.1 训练基本算法

6.1.1 基本算法描述

AdaBoost 训练强分类器的算法描述如下^[19] (算法流程图见下节) :

A. 给定一系列训练样本 $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, 其中 $y_i = 0$ 表示其为负样本 (非人脸), $y_i = 1$ 表示其为正样本 (人脸), n 为一共的训练样本数量。

B. 初始化权重⁴ $w_{1,i} = D(i)$;

C. 对 $t = 1, \dots, T$:

1. 归一化权重 :

$$q_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$$

2. 对每个特征 f , 训练一个弱分类器 $h(x, f, p, q)$; 计算对应所有特征的弱分类器的加权 (q_t) 错误率 e_f :

⁴ 可以这样设置 $D(i)$: $D(i) = \frac{1}{2m}$ (对负样本) 或 $D(i) = \frac{1}{2l}$ (对正样本) 。其中 m, l 分别是负样本和正样本的数量 , $m + l = n$



$$\mathbf{e}_f = \sum_i q_i |h(x_i, f, p, \mathbf{q}) - y_i|$$

3. 选取最佳的弱分类器 $h_t(x)$ (拥有最小错误率 \mathbf{e}_t) :

$$\mathbf{e}_t = \min_{f, p, \mathbf{q}} \sum_i q_i |h(x_i, f, p, \mathbf{q}) - y_i| = \sum_i q_i |h(x_i, f_t, p_t, \mathbf{q}_t) - y_i|$$

$$h_t(x) = h(x, f_t, p_t, \mathbf{q}_t)$$

(弱分类器的训练及选取参见 6.2 节)

4. 按照这个最佳弱分类器 , 调整权重 :

$$w_{t+1,i} = w_{t,i} b_t^{1-e_i}$$

其中 $e_i = 0$ 表示 x_i 被正确地分类 , $e_i = 1$ 表示 x_i 被错误地分类 ;

$$b_t = \frac{\mathbf{e}_t}{1 - \mathbf{e}_t}$$

D. 最后的强分类器为 :

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T \mathbf{a}_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \mathbf{a}_t \\ 0 & \text{其它} \end{cases}$$

$$\text{其中 } \mathbf{a}_t = \log \frac{1}{b_t}$$



6.1.2 基本算法流程图

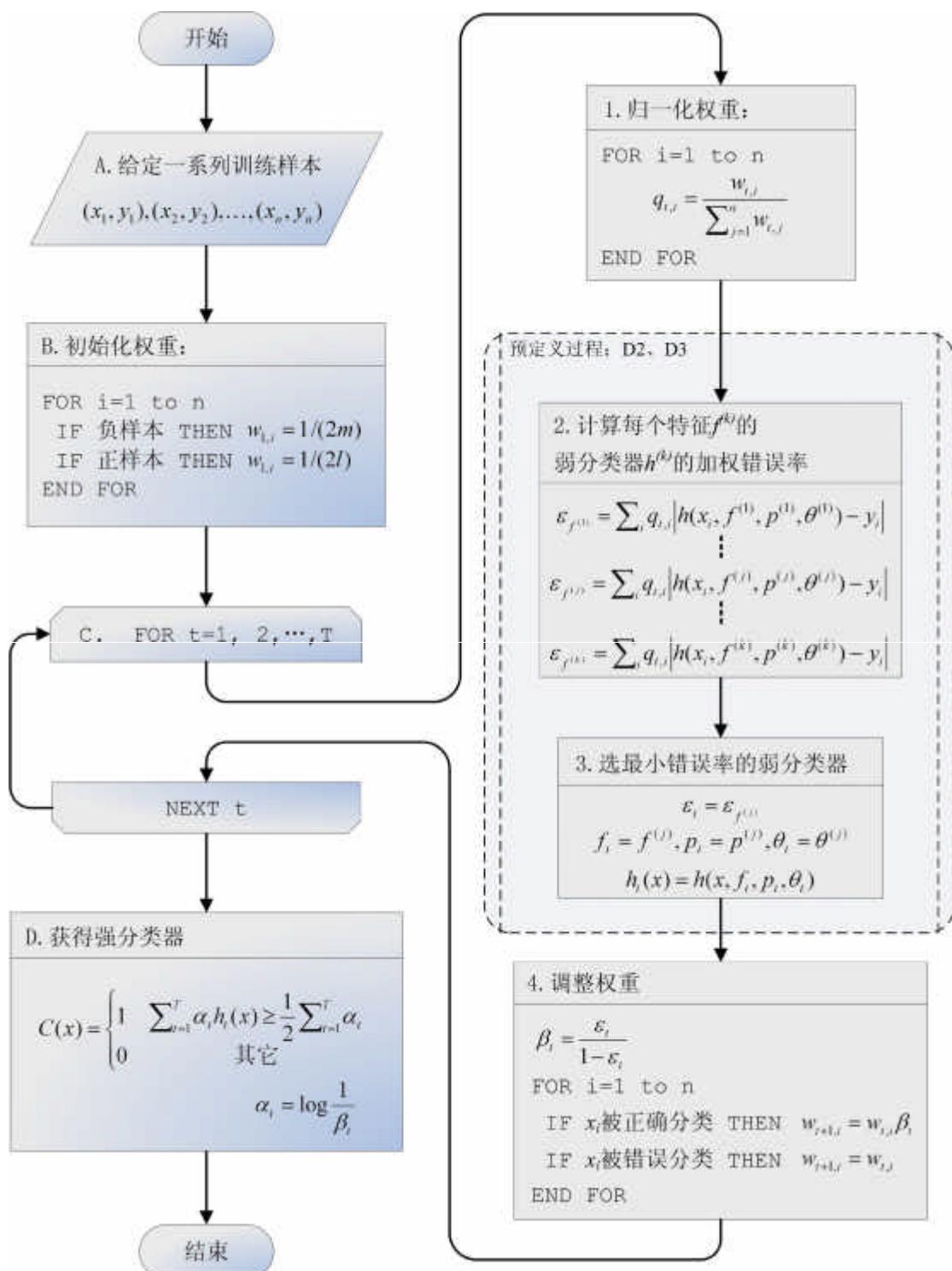


图 15 AdaBoost 训练强分类器算法流程图



6.2 弱分类器 Weak Classifier

一个弱分类器 $h(x, f, p, \mathbf{q})$ 由一个特征 f , 阈值 \mathbf{q} 和指示不等号方向的 p 组成 :

$$h(x, f, p, \mathbf{q}) = \begin{cases} 1 & pf(x) < p\mathbf{q} \\ 0 & \text{其它} \end{cases}$$

6.2.1 特征值 $f(x)$

对于本算法中的矩形特征来说 , 弱分类器的特征值 $f(x)$ 就是矩形特征的特征值^(5.2.2)。由于在训练的时候 , 选择的训练样本集的尺寸等于检测子窗口的尺寸 , 检测子窗口的尺寸决定了矩形特征的数量^(5.2.3.3) , 所以训练样本集中的每个样本的特征相同且数量相同 , 而且一个特征对一个样本有一个固定的特征值。

对于理想的像素值随机分布的图像来说 , 同一个矩形特征对不同图像的特征值的平均值应该趋于一个定值⁵ : k 。

这个情况 , 也应该发生在非人脸样本上 , 但是由于非人脸样本不一定是像素随机的图像 , 因此上述判断会有一个较大的偏差。

对每一个特征 , 计算其对所有的一类样本(人脸或者非人脸)的特征值的平均值 , 最后得到所有特征对所有一类样本的平均值分布。

如图 16 , 显示了 20×20 子窗口里面的全部 78,460 个矩形特征对全部 2,706 个人脸样本和 4,381 个非人脸样本⁶ 的特征值平均数的分布图。由分布看出 , 特征的绝大部分的特征值平均值都是分布在 0 前后的范围内。出乎意料的是 , 人脸样本与非人脸样本的分布曲线差别并不大 , 不过注意到特征值大于或者小于某个值后 , 分布曲线出现了一致性差别 , 这说明了绝大部分特征对于识别人脸和非人脸的能力是很微小的 , 但是存在一些特征及相应的阈值 , 可以有效地区分人脸样

⁵ 这个应该很好理解 , 对于随机分布的像素值 , 由于矩形特征中的矩形块的大小相同 , 则每个块内的总像素值应该在统计上是相同的。特别地 , 对于有相同数目白块和黑块的矩形特征 , 这个定值 k 应为 0

⁶ 本文使用的是一套 MIT 的人脸训练样本库 , 共有人脸图像 2706 张和非人脸图像 4381 张。其中人脸图像的示例参见图 3 , 非人脸图像的示例参见图 20。



本与非人脸样本。

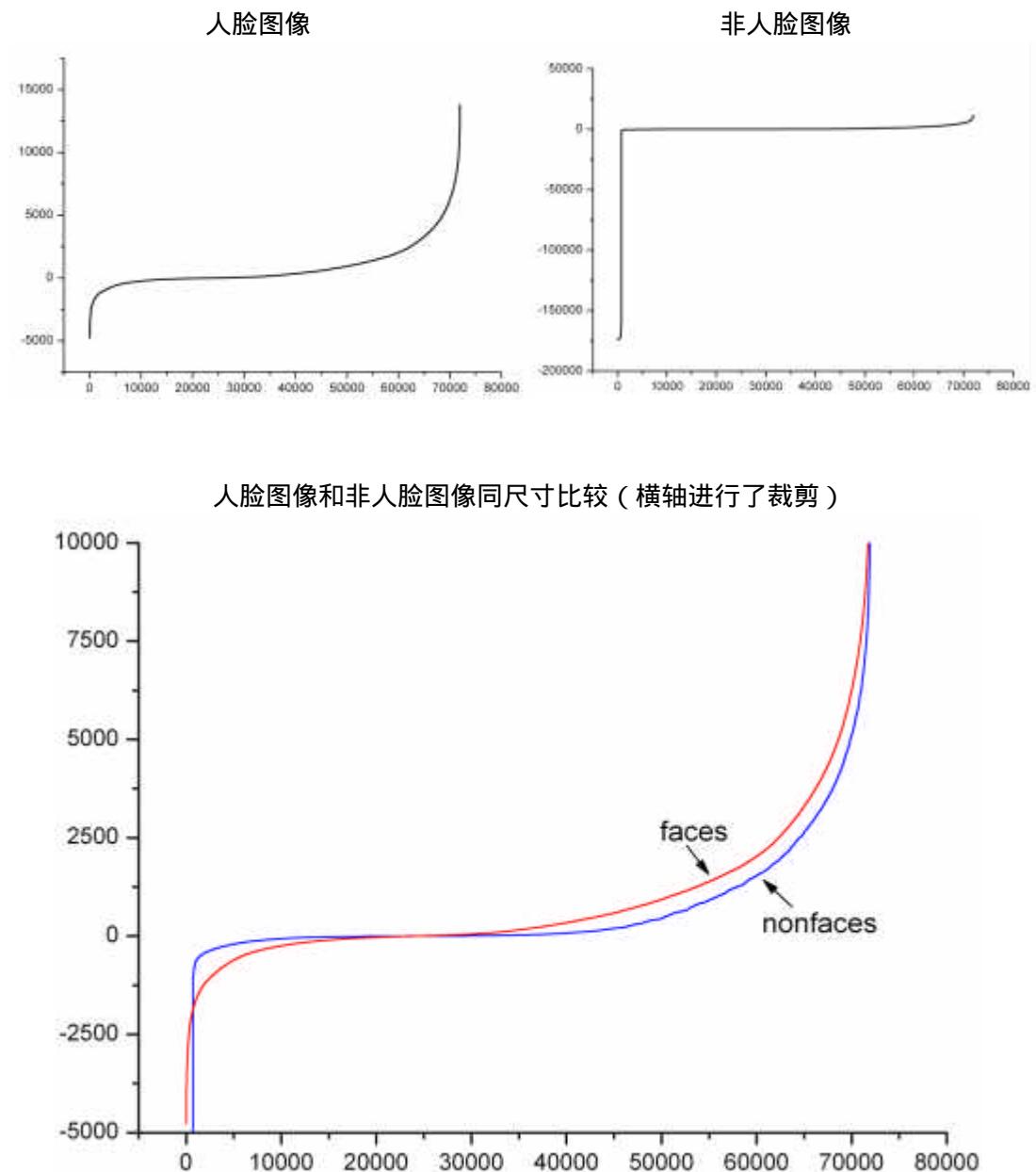


图 16 所有矩形特征对所有图片的特征值平均数分布
(横坐标是按特征值平均值排序后的特征编号)

为了更好地说明问题，我们从 78,460 个矩形特征中随机抽取了两个特征 A 和 B⁷，这两个特征遍历了 2,706 个人脸样本和 4,381 个非人脸样本，计算了每

⁷ 在 FáDét 程序里面，A 和 B 对应的矩形特征索引分别是 12435 和 32789。这两个特征都是选取的 1x2 形式的，为的是更简单地说明问题。



张图像对应的特征值，最后将特征值进行了从小到大的排序，并按照这个新的顺序表绘制了分布图，如图 17 和图 18。一些统计数据如下：

特征	特征 A			特征 B		
样本	人脸	非人脸	差值	人脸	非人脸	差值
样本数量	2,706	4,381	-	2,706	4,381	-
总和	330,953	-137,143	468,096	-4,480,530	-1,601,750	-2,878,780
平均值	122.30	-31.30	153.6	-1655.78	-365.61	-2,021.39
中值	116	-17	133	-1,484	-144	-1628
范围差	14,133	23,503	-9,370	8,763	14,760	-5997
0 点左边特征百分比	46.5%	51.5%	-5%	93.4%	59.4%	34%

表 6 特征 A、B 对所有图像特征值的统计表

特征 A 和特征 B 的表现大相径庭。

如图 17，特征 A 对人脸和非人脸样本的特征值为 0 的点几乎处于相同位置（46.5%，51.5%），且都在所有特征的中间范围⁸(-5%)。这说明矩形特征 A 对于人脸和非人脸几乎没有分辨能力。

如图 18，特征 B 对非人脸样本的分布，符合我们的预想，特征值为 0 的点处于所有特征的中间范围(59.4%)，这说明特征 B 也“看不到”非人脸的特点。但是对于人脸样本，特征 B 表现了很一致的倾向性，93.4% 的特征在 0 点的一侧，与非人脸样本的相差 34%。这说明特征 B 能够相当可靠地分辨人脸和非人脸。

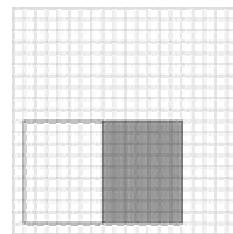
上述的分析说明，确实存在优势的矩形特征，能够在一定的置信范围内区分人脸和非人脸。由于是使用统计的方法计算人脸图像和非人脸图像的差别，因此最后得到的区分阈值，也只能是在某个概率范围内准确地进行区分。

⁸ 对于特征 A 和 B 来说，其对图像的特征值越接近 0，可以在一个方面说明这个图像在矩形特征这个区域的随机性越大；如果 0 点越接近中心，则说明越多的图像被这个特征“平均”(也可是随机)地被区分在两侧，也就是说这个特征对“看不到”这些图像的特点。

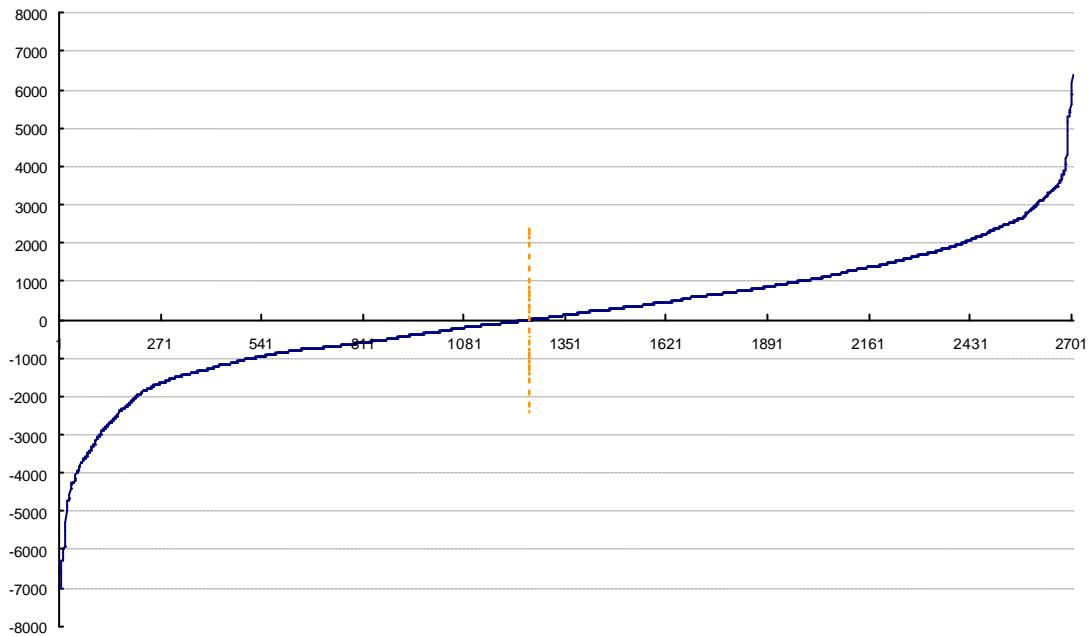


矩形特征 A 在 20×20 子窗口中位置如右图

参数：坐标 (2, 11)-(15, 19) (s, t) 条件 (2, 1)



人脸图像特征值分布



非人脸图像特征值分布

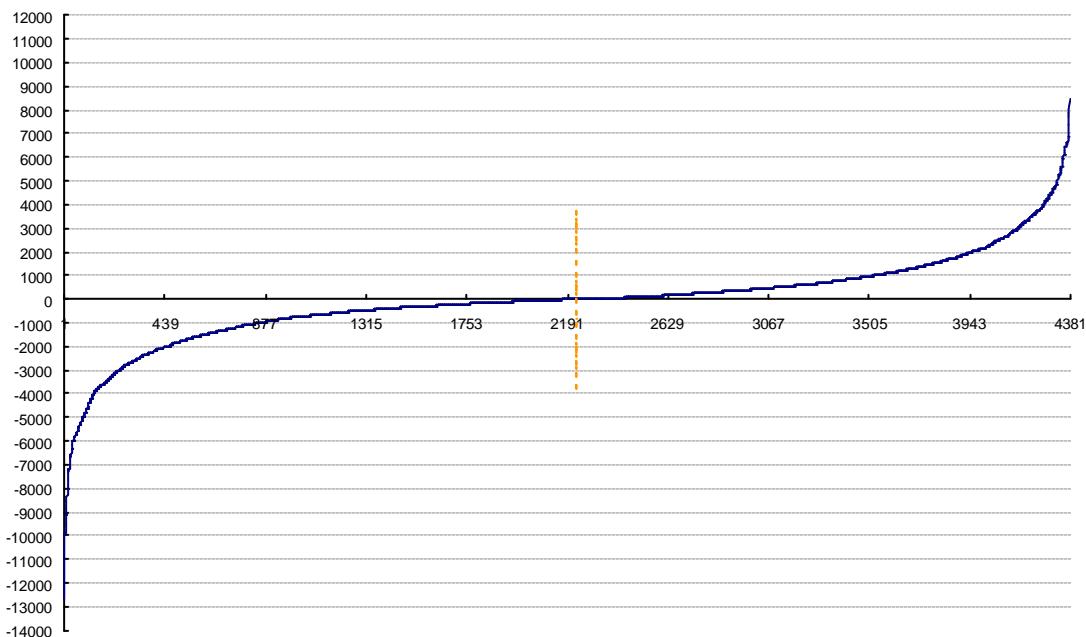
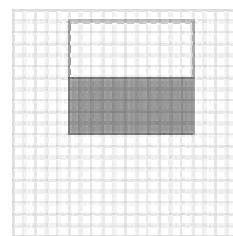


图 17 矩形特征 A 对人脸和非人脸图像的特征值分布(横坐标为排序表编号)。这里看不出 A 有区分能力。

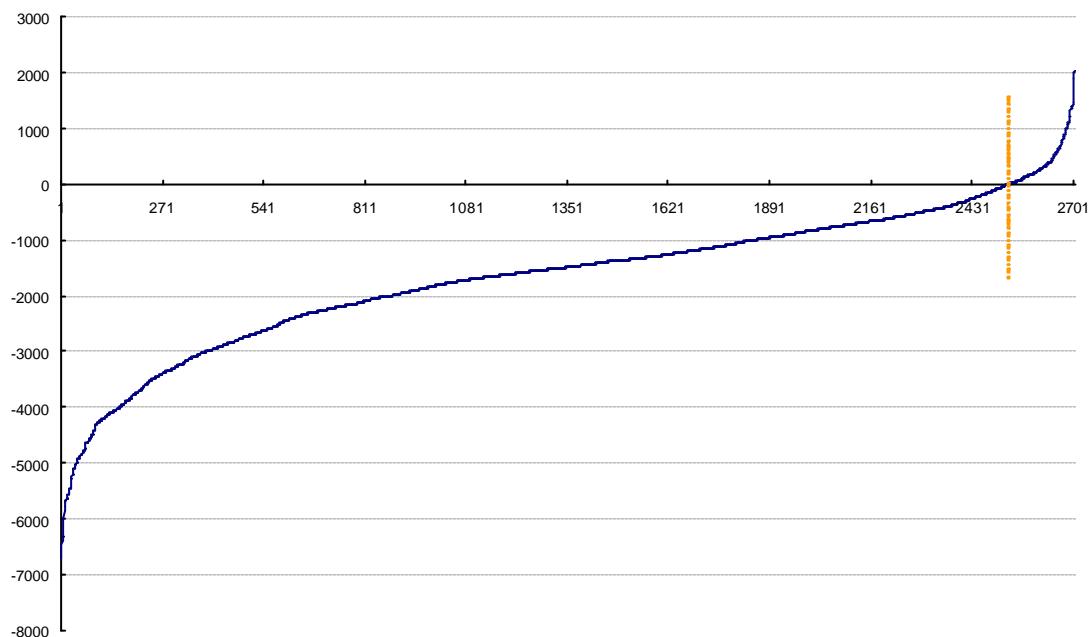


矩形特征 B 在 20×20 子窗口中位置如右图

参数：坐标 (6, 2)-(16, 11) (s, t) 条件 (1, 2)



人脸图像特征值分布



非人脸图像特征值分布

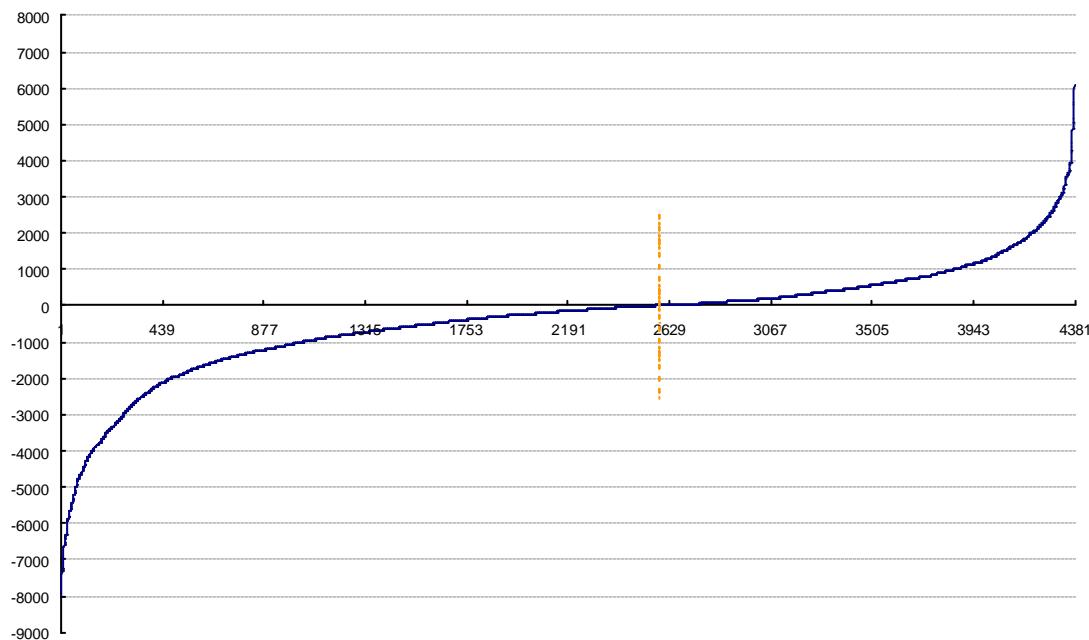


图 18 矩形特征 B 对人脸和非人脸图像的特征值分布(横坐标为排序表编号)。这里 B 表现了很强的分辨能力。



6.2.2 阈值 q 、方向指示符 p

由上述的分析，阈值 q 的含义就清晰可见了。而方向指示符 p 用以改变不等号的方向。

一个弱学习器（一个特征）的要求仅仅是：它能够以稍低于50%的错误率来区分人脸和非人脸图像，因此上面提到只能在某个概率范围内准确地进行区分就已经完全足够。

按照这个要求，可以把所有错误率低于50%的矩形特征都找到（适当地选择阈值，对于固定的训练集，几乎所有的矩形特征都可以满足上述要求）。每轮训练，将选取当轮中的最佳弱分类器（在算法中，迭代 T 次即是选择 T 个最佳弱分类器），最后将每轮得到的最佳弱分类器按照一定方法提升（Boosting）为强分类器（参见6.3节）。

下节将说明如何训练并选取最佳弱分类器。

6.2.3 弱分类器的训练及选取

训练一个弱分类器（特征 f ）就是在当前权重分布的情况下，确定 f 的最优阈值，使得这个弱分类器（特征 f ）对所有训练样本的分类误差最低。

选取一个最佳弱分类器就是选择那个对所有训练样本的分类误差在所有弱分类器中最低的那个弱分类器（特征）。

对于每个特征 f ，计算所有训练样本的特征值，并将其排序。通过扫描一遍排好序的特征值，可以为这个特征确定一个最优的阈值，从而训练成一个弱分类器（参考图19）。具体来说，对排好序的表中的每个元素⁹，计算下面四个值：

- 1) 全部人脸样本的权重的和 T^+ ；
- 2) 全部非人脸样本的权重的和 T^- ；
- 3) 在此元素之前的人脸样本的权重的和 S^+ ；
- 4) 在此元素之前的非人脸样本的权重的和 S^- ；

⁹ 这里一个元素指排序表中的一个对应某个样本的特征值



这样，当选取当前元素的特征值 F_{k_j} 和它前面的一个特征值 $F_{k_{j-1}}$ 之间的数作为阈值时，所得到的弱分类器就在当前元素处把样本分开——也就是说这个阈值对应的弱分类器将当前元素前的所有元素分类为人脸（或非人脸），而把当前元素后（含）的所有元素分类为非人脸（或人脸）。

可以认为这个阈值所带来的分类误差为¹⁰：

$$e = \min(S^+ + (T^- - S^-), S^- + (T^+ - S^+))$$

于是，通过把这个排序的表扫描从头到尾扫描一遍就可以为弱分类器选择使分类误差最小的阈值（最优阈值），也就是选取了一个最佳弱分类器。

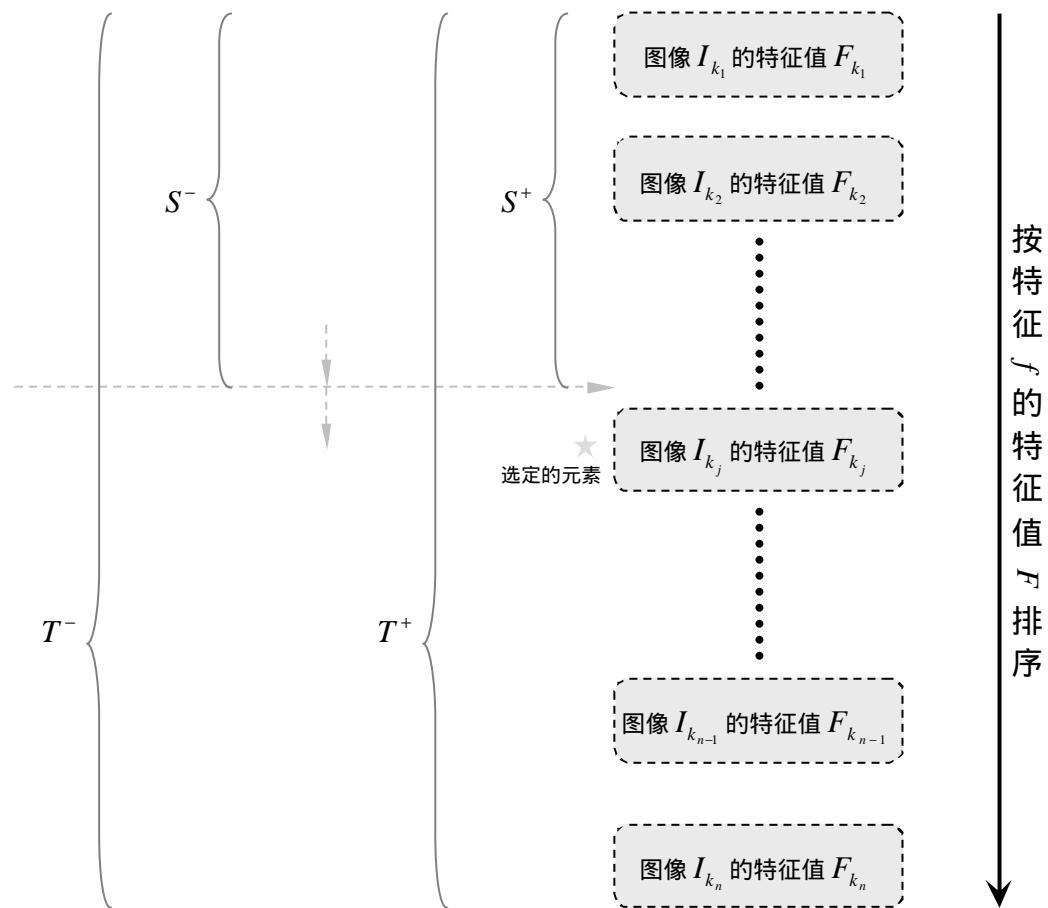


图 19 训练并选取最佳分类器算法

¹⁰ 这个分类误差就是加权错误率。



6.3 强分类器 Strong Classifier

6.3.1 构成

经过 T 次迭代后，获得了 T 个最佳弱分类器 $h_1(x), \dots, h_T(x)$ ，可以按照下面的方式组合成一个强分类器：

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T \mathbf{a}_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \mathbf{a}_t \\ 0 & \text{其它} \end{cases}$$

其中 $\mathbf{a}_t = \log \frac{1 - \mathbf{e}_t}{\mathbf{b}_t} = \log \frac{1 - \mathbf{e}_t}{\mathbf{e}_t} = -\log \mathbf{e}_t$

那么，这个强分类器对待一幅待检测图像时，相当于让所有弱分类器投票，再对投票结果按照弱分类器的错误率加权求和，将投票加权求和的结果与平均投票结果比较得出最终的结果。

平均投票结果，即假设所有的弱分类器投“赞同”票和“反对”票的概率都相同下，求出的概率平均， $\frac{1}{2}(\sum_{t=1}^T \mathbf{a}_t \cdot 1 + \sum_{t=1}^T \mathbf{a}_t \cdot 0) = \frac{1}{2} \sum_{t=1}^T \mathbf{a}_t$ 。

实验结果表明，当 $T=200$ 时，构成的强分类器可以获得很好的检测效果。

6.3.2 错误率上限

可以证明用上述强分类器进行人脸检测的错误率上限为：

$$\mathbf{e} \leq 2^T \prod_{t=1}^T \sqrt{\mathbf{e}_t(1 - \mathbf{e}_t)}$$

现证明如下。首先，



$$\begin{aligned}
\sum_{i=1}^n w_{t+1,i} &= \sum_{i=1}^n w_{t,i} \mathbf{b}_t^{1-e_i} \\
&\leq \sum_{i=1}^n w_{t,i} [1 - (1 - \mathbf{b}_t)(1 - e_i)] \\
&= \left(\sum_{i=1}^n w_{t,i} \right) \left[1 - (1 - \mathbf{b}_t) \left(1 - \sum_{i=1}^n \frac{w_{t,i}}{\sum_{i=1}^n w_{t,i}} e_i \right) \right] \\
&= \left(\sum_{i=1}^n w_{t,i} \right) \left[1 - (1 - \mathbf{b}_t) \left(1 - \sum_{i=1}^n q_{t,i} e_i \right) \right] \\
&= \left(\sum_{i=1}^n w_{t,i} \right) [1 - (1 - \mathbf{b}_t)(1 - \mathbf{e}_t)]
\end{aligned}$$

上式用到了 $(1+x)^m = 1 + mx + \dots$; 将上式按照 $t=1, \dots, T$ 个分类器迭代, 可以得到:

$$\sum_{i=1}^n w_{T+1,i} \leq \prod_{t=1}^T [1 - (1 - \mathbf{b}_t)(1 - \mathbf{e}_t)] \quad (1)$$

由强分类器的判断条件:

$$\sum_{t=1}^T \mathbf{a}_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \mathbf{a}_t$$

且:

$$\begin{aligned}
\sum_{t=1}^T \mathbf{a}_t h_t(x) &= \sum_{t=1}^T \log \frac{1}{\mathbf{b}_t} h_t(x) = \log \prod_{t=1}^T \mathbf{b}_t^{-h_t(x)} , \\
\frac{1}{2} \sum_{t=1}^T \mathbf{a}_t &= \frac{1}{2} \sum_{t=1}^T \log \frac{1}{\mathbf{b}_t} = \log \left(\prod_{t=1}^T \mathbf{b}_t \right)^{-1/2}
\end{aligned}$$

所以强分类器对样本 i 判断错误的条件为:

$$\prod_{t=1}^T \mathbf{b}_t^{-e_i} \geq \left(\prod_{t=1}^T \mathbf{b}_t \right)^{-1/2} \quad (2)$$

又有样本 i 最终的权重为:

$$w_{T+1,i} = D(i) \prod_{t=1}^T \mathbf{b}_t^{1-e_i} \quad (3)$$

由(2)(3)式可以得到所有的权重和的下限:



$$\begin{aligned}
\sum_{i=1}^n w_{T+1,i} &\geq \sum_{i:e_i=1} w_{T+1,i} \\
&\geq \left(\sum_{i:e_i=1} D(i) \right) \left(\prod_{t=1}^T b_t \right)^{1/2} \\
&= e \cdot \left(\prod_{t=1}^T b_t \right)^{1/2}
\end{aligned} \tag{4}$$

$e = \sum_{i:e_i=1} D(i)$ 即为最终的错误率。由(1)及(4)可得：

$$\begin{aligned}
e &\leq \left(\prod_{t=1}^T b_t \right)^{-1/2} \prod_{t=1}^T [1 - (1 - b_t)(1 - e_t)] \\
&= \prod_{t=1}^T \frac{1 - (1 - b_t)(1 - e_t)}{\sqrt{b_t}} \\
&= \prod_{t=1}^T \frac{2e_t}{\sqrt{\frac{e_t}{1 - e_t}}} \\
&= 2^T \prod_{t=1}^T \sqrt{e_t(1 - e_t)}
\end{aligned}$$

证毕。

下面简单分析一下强分类器的错误率。设 $g_t = \frac{1}{2} - e_t$ ，那么上式可改写为：

$$\begin{aligned}
e &\leq \prod_{t=1}^T \sqrt{1 - 4g_t^2} \\
&= \exp(\ln \sqrt{1 - 4g_1^2}) \times \cdots \times \exp(\ln \sqrt{1 - 4g_T^2}) \\
&= \exp\left(\frac{1}{2} \sum_{t=1}^T \ln(1 - 4g_t^2)\right) \\
&\leq \exp\left(-2 \sum_{t=1}^T g_t^2\right)
\end{aligned}$$

上式最后一步用到了泰勒展开 $\ln(1 - x) = -(x + \cdots)$ 。

特别的，如果所有的弱分类器错误率 $e_t = e_0$ 相等， $g_t = g_0$ ，上式可简化为：

$$e \leq \exp(-2Tg_0^2)$$

可见，强分类器错误率随着弱分类器数量增多和弱分类器错误率的降低而迅速降低。具体地，如果 $T = 200, e_0 = 0.40$ ，那么 $e \leq 0.018$ （正确率高于 98%）。这也说明了，只要有足够多的弱分类器，就可以使强分类器达到任意低的错误率。



7 程序实现及结果

7.1 样本集

AdaBoost 算法效果如何的重点在于前期训练时候的弱分类器的选取，而弱分类器的选取在一定程度上依赖于样本集的选取。因而训练样本集显得十分重要。在 1.3.1 节中提到了一些图像库，最大最全的是 FERET 数据库，但是其由美国军方开发，在美国以外地区无法获得。

本文使用的是一套 MIT 的人脸训练样本库，共有样本 7,087 个，其中人脸样本 2,706 个，非人脸样本 4,381 个。其中人脸图像的示例参见图 3，非人脸图像的示例参见下图。



图 20 训练用的非正面人脸图像

这个样本库中的人脸样本都是 20×20 像素、正面正向人脸，有同一个人的多种光照条件下的样本。非人脸样本尺寸和人脸样本相同，是从图像中随机自动截取的。这个样本库的将侧面人脸、非正向人脸、有遮挡的人脸都归入了非人脸样本。因此，依此样本库训练的强分类器只对正面正向人脸敏感。



7.2 训练难点及优化

7.2.1 计算成本

由于特征数量极其多，训练的时间和空间开销特别大。在每轮循环中，都要计算所有样本对所有特征的特征值，则需要运算：

$$78,460 \times 7,087 = 556,046,020 \text{ 次}$$

在 PIII 850MHz、512Mb 的机器上实测得到，上述运算共花时 675 秒（11 分钟 15 秒）。为了训练 200 个弱分类器，则需要进行 200 轮运算，则共需要 37.5 小时。但这并不是最终的运算时间，最终的运算时间要加上每轮运算中选择最佳阈值所要进行的排序、遍历等运算。时间的开销是巨大的。

因为一个特征对一幅图像的特征值是固定的，那么可以只运行一次上述运算，将得到的特征值结果都保存在硬盘上，这样可以减少每次计算的时间开销，需要的磁盘空间为（采用 4 字节的 unsigned long 存储）：

$$556,046,020 \times 4 = 2,224,184,080 \text{ Bytes} = 2.07 \text{ Gbytes}$$

空间开销是巨大的。但是，对于一般的 PC 机，硬盘的峰值读取速度为 66Mb/s，则读取上述内容的时间开销仍有 32 秒。

7.2.2 减少矩形特征的数量

为了提高训练速度，得从其他方面考虑。分析知道，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，主要有两个途径。

第一是在特征生成阶段人为剔出。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。这一步，特征数量可以减少 6,520 个，变成 71,940 个。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¹¹，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到 46,953 个。

第二步是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。

进行了上述处理后，运算速度可以得到 40% 以上的提升。

7.2.3 样本预处理

为了减小图像本身由于灰度分布¹²带来造成的影响，需要对训练样本做归一化处理。

对于 MIT 训练库中的所有样本，采用直方图归一化处理所有的样本（见图 21）。进行人脸检测时，也可以按此值进行归一化。

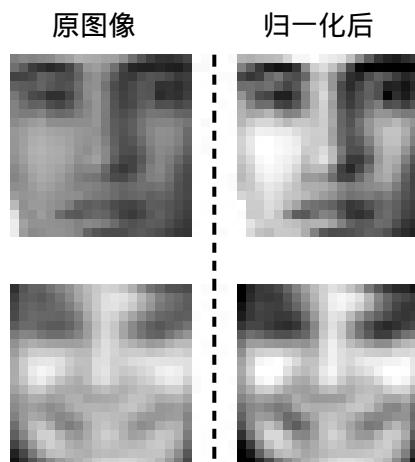


图 21 样本归一化前后对比

¹¹ 这种减少和训练集有相当的关系，如果训练集的人脸样本的截取已经考虑了边缘效应，那么本方法将增加训练的难度

¹² 这种灰度值分布偏差往往是因为拍摄时的光线、曝光等因素造成的。直方图归一化也可以由 Photoshop 处理完成



7.3 检测结果

7.3.1 检测器

用于检测的图像一般都大于子窗口 20×20 的分辨率。于是，在对一副图像作检测的时候需要从不同的尺度和不同的位置逐个检测。

对于不同尺度的检测，我们选择对检测器进行缩放而不是对图像本身进行缩放，因为在任何尺度下，特征都可以用同样的代价求出^(5.3.2.2)。我们的使用的检测器基本大小为 20×20 ，实验结果表明，检测器每次放大 1.25 倍是最佳的。

此外，检测器还需要在不同的位置逐个检测。每一次，检测器平移 k 个象素；如果放大后的检测器尺度为 S ，则检测器窗口每次平移 kS 。这里， k 的选择非常重要，它既影响检测的精度，又影响检测的速度。如果 k 取的较大，检测速度可以很快，但是同时很多人脸由于检测器的移动步长较大最终被忽略了；反之，如果 k 取的较小，虽然可能会提高检测的精度，但是将使检测速度大大降低。在实际检测中，按照人脸结构的“三庭五眼”准则¹³，选择 $k = 20 \cdot \frac{1}{5} = 4$ 。

由于检测器对于尺度和位置上的微小变化不敏感，所以，对于同一张人脸，检测结果中可能会把它检测出很多次，因而我们需要对检测结果进行消减。消减的方法十分简单：对于边界有相交区域的检测结果，对它们的四个顶点分别取平均值作为新的顶点，从而合成一个检测结果。

¹³ “三庭”指脸的长度比例，把脸的长度分为三个等分，从前额发际线至眉骨，从眉骨至鼻底，从鼻底至下颌，各占脸长的 $1/3$ 。“五眼”指脸的宽度比例，以眼形长度为单位，把脸的宽度分成五个等分，从左侧发际至右侧发际，为五只眼形。两只眼睛之间有一只眼睛的间距，两眼外侧至侧发际各为一只眼睛的间距，各占比例的 $1/5$ 。这来自中国古代画家总结出来的规则。按照这个准则，检测器的移动基本上总可以包含一个人脸。



7.3.2 实验结果

7.3.2.1 实验对比

本节和 Viola 实验的结果^[19]进行了比较。



图 22 与 Viola 实验结果比较 1：总数 15/漏检 0/错检 5



Viola 的结果



本文结果

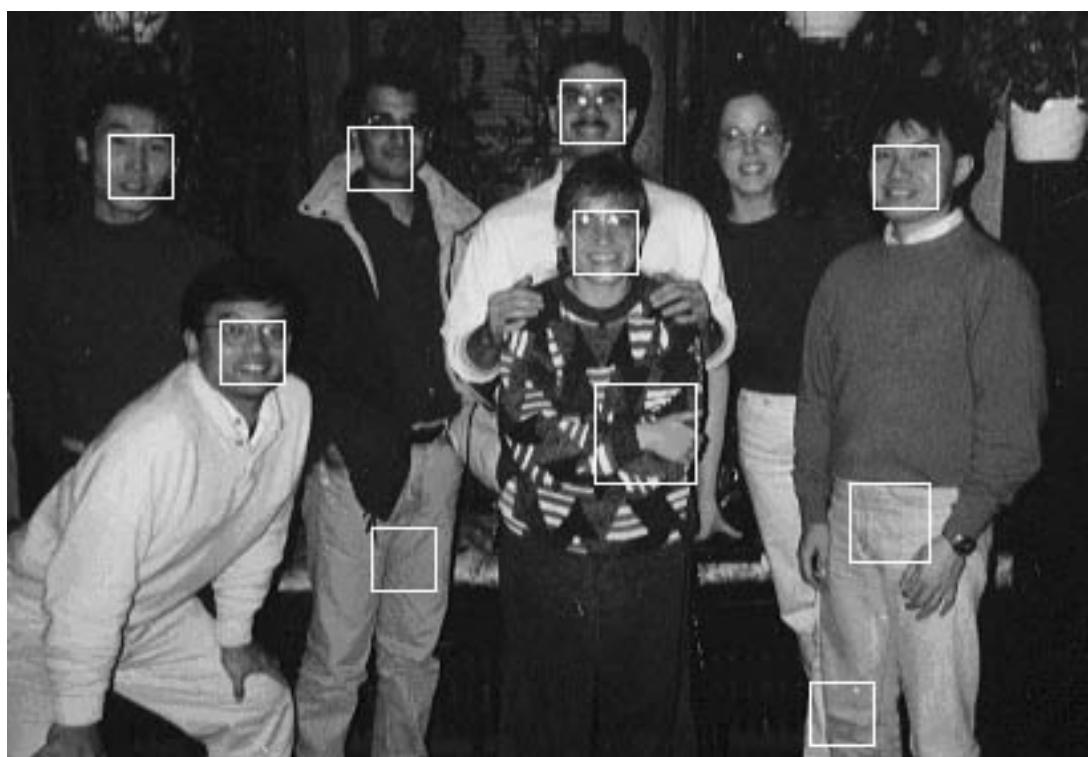


图 23 与 Viola 实验结果比较 2 : 总数 7 / 漏检 1 / 错检 4



7.3.2.2 更多实验结果

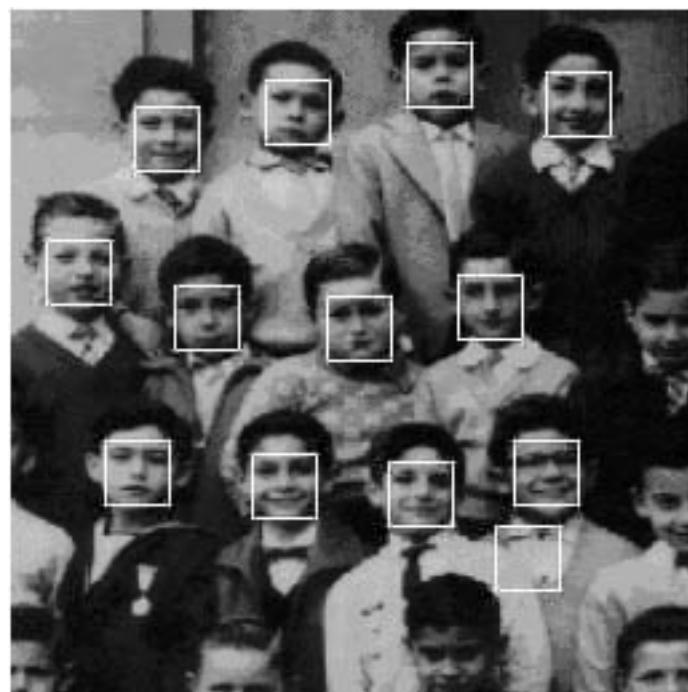


图 24 多人图像检测结果：总数 14/漏检 2/错检 1

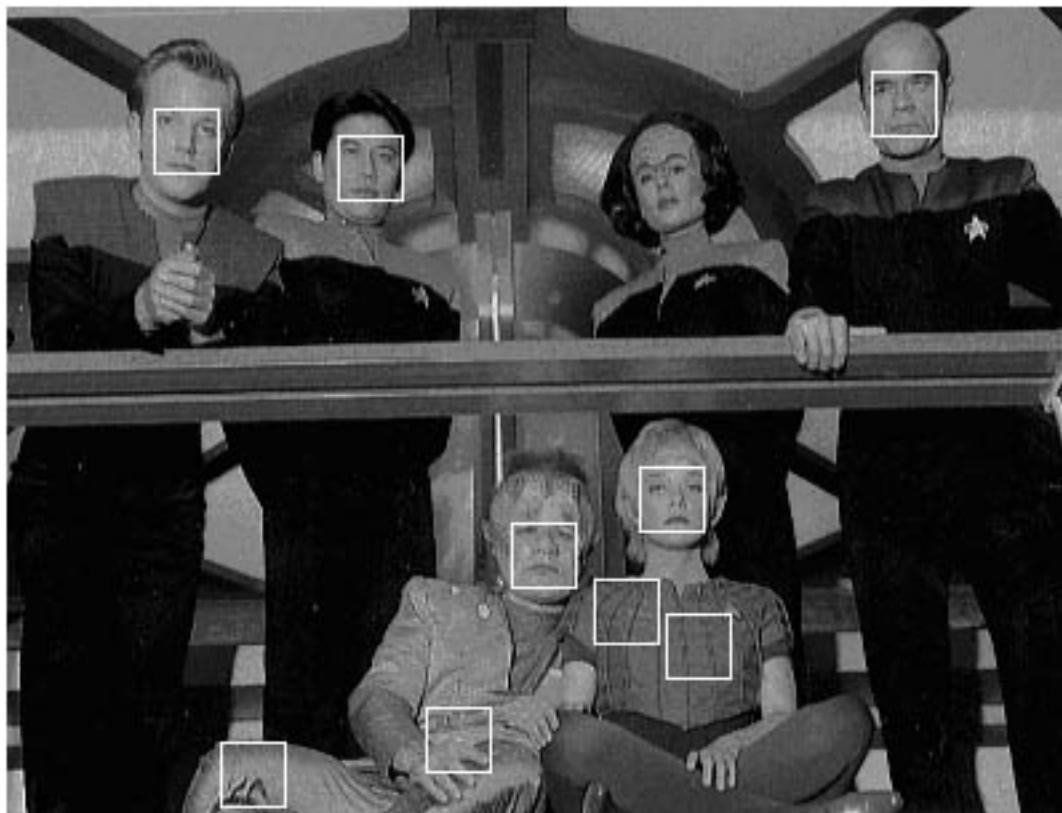


图 25 多人图像检测结果：总数 6/漏检 1/错检 4



图 26 多人图像检测结果：总数 33/漏检 0/错检 4



图 27 多人图像检测结果：总数 33/漏检 5/错检 14



图 28 多人图像检测结果：总数 3/漏检 0/错检 0



图 29 多人图像检测结果：总数 3/漏检 0/错检 0

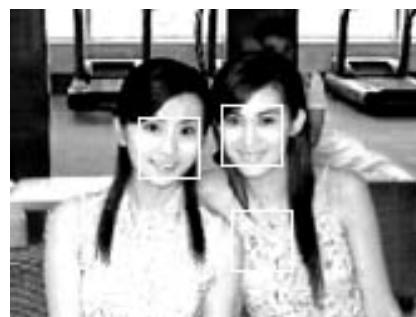


图 30 多人图像检测结果：总数 2/漏检 0/错检 1

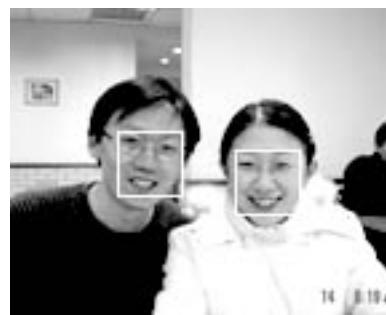


图 31 多人图像检测结果：总数 2/漏检 0/错检 0



图 32 单人图像检测结果：总数 1/漏检 0/错检 2

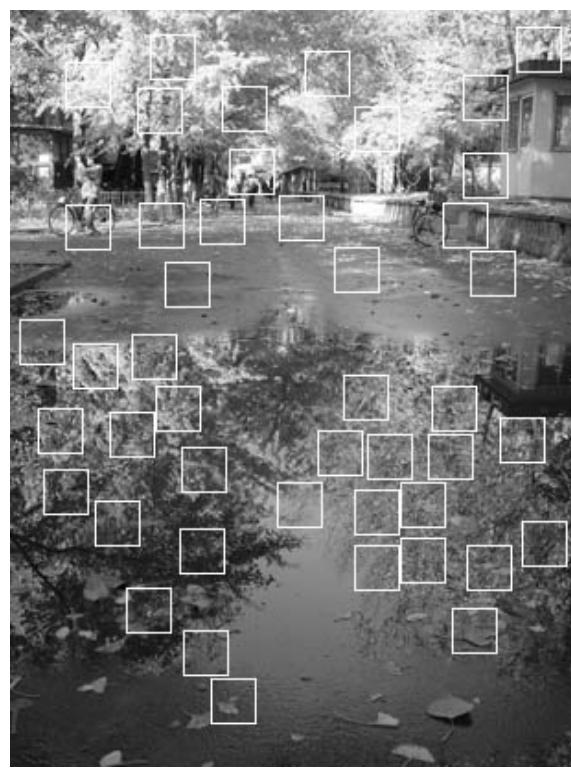


图 33 不含人脸图像检测结果：总数 0/漏检 0/错检 45



7.3.3 结论

Viola 由于采用了层级处理方法，大大提高了检测速度，对于 384×288 像素的图像，在 PIII 700MHz 的计算机上，检测速度可以达到 15 帧/秒。本文的算法在 PIII 850MHz 的计算机上，检测 384×288 像素的图像，约需要 30 秒，比 Viola 的层级处理方法慢了许多。

另外，本方法的漏检和错检率还比较高。这主要是因为训练集数量太少，而且非人脸样本比重偏低。事实上，世界上的非人脸样式要远远大于人脸样式，因此有足够的非人脸样本，才能够让分类器对非人脸图案的“分辨”能力大大提高。



致谢 Acknowledgments

感谢我的导师查红彬教授给了一个走进人工智能的机会，即使只是管中窥豹，也让我深深为这个领域的博大精深感到惊奇和感动。感谢查老师悉心的指导，他对这个领域独特的眼光、敏锐的触觉和对学术的严谨都让我获益匪浅；感谢查老师的平易近人和对我的包容，让我能够坚持走下来；

感谢武宇文博士在论文选题方面给我的莫大帮助和鼓励，扩展了我的知识面，让我能够从各方面了解这个领域并最终选择了自己最感兴趣的课题；

感谢刘一师兄给予的最热情的指导和最无私的帮助，每当陷入困境，他都会即使施与援手，让我看到困难后面的希望，也让我充满了奋斗的激情；

感谢崔锦实老师对本论文的审阅，并提出了许多中肯宝贵的修改建议；

感谢实验室里的师兄、师姐们，将实验室营造得家一般团结和温暖，让人愉悦和舒畅；

最后，感谢物理学院高政祥教授为我担当外院系做毕设的负责人。有了高老师的支特，我才能没有顾忌地在自己喜欢的领域里翱翔。

参考文献 References

- [1] Yang Ming-Hsuan , David J. K. , Narendra A., *Detecting Faces in Images: A Survey*、*IEEE Transactions on Pattern Analysis and Machine Intelligence*、 Vol 24(1):34-58, 2002/01
- [2] Hjelmas E. , Low B. K.、*Face Detection: A Survey*、*Computer Vision and Image Understanding* 83,236~274(2001)
- [3] 李刚，高政、人脸自动识别方法综述、*计算机应用研究*、2003, Vol 8:4-9
- [4] Rowley H. A. , et al、*Neural Network-Based Face Detection*、*IEEE Transactions on Pattern Analysis and Machine Intelligence*、 Vol 20(1):23-38, 1998/01



- [5] T. Kohonen , *Self-Organization and Associative Memory* , Springer 1989
- [6] Belhumeur P.N. ,Hespanha J. P. ,Kriegman D. J., *Eigenfaces vs. Fisherface: Recognition Using Class Specific Linear Projection*, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19(7), 711-720, July 1997
- [7] K. K. Sung , T. Poggio, *Example-Based Learning for View-Based Human Face Detection*、*Technical Report AI Memo 1521*, Massachusetts Inst. of Technology AI Lab, 1994
- [8] E. Osuna , R. Freund , and F. Girosi, *Training Support Vector Machines: An Application to Face Detection*、*Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 130-136, 1997
- [9] B. Heiselet , T Serre , M Pontil et al, *Component-based Face Detection*、*Proc IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001, 1:I657-I662
- [10] Nefian A. , Hayes M., *Hidden Markov Models for Face Recognition*、*IEEE International Conference on Acoustics, Speech and Signals Processings*, Seattle, Washington, 1998:2721-2724
- [11] Freund Y., Schapire R. E. *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*. *Journal of Computer and System Sciences*, 1997, 55(1):119-139
- [12] 宋春雷 ,王龙等、学习理论与鲁棒控制、控制理论与应用、Vol 17(5), 633-636, 2000/10
- [13] Valiant L.G., *A Theory of the Learnable*、*Communications of the ACM*、 Vol 27 (11), 1134-1142, 1984/11
- [14] Kearns M., *The Computational Complexity of Machine Learning*、*Cambridge: MIT Press*, 1990
- [15] Kearns M. ,Valiant L. G., *Cryptographic Limitations on Learning Boolean Formulae and Finite Automata*, *Journal of the ACM*, 41(1):67-95, 1994/01
- [16] Schapire R. E., *The Strength of Weak Learnability*、*Machine Learning*, 1990, 5(2):197-227



- [17] Freund Y., *Boosting a Weak Learning Algorithm by Majority, Information and Computation*, 1995, 121(2):256-285
- [18] Drucker H. , Schapire R. E. , Simard P., *Boosting Performance in Neural Networks, International Journal of Pattern Recognition and Artificial Intelligence*, 1993, 7(4):705-719
- [19] Viola P. , Jones M. J., *Robust Real-Time Face Detection, International Journal of Computer Vision* 57(2), 137-154, 2004
- [20] Viola P. , Jones M J., *Rapid Object Detection using a Boosted Cascade of Simple Features, Computer Vision and Pattern Recognition*, 2001, Volume 1, 8-14
- [21] Viola P. , Jones M. J., *Robust Real-time Object Detection, Cambridge Research Laboratory, Technical Report Series*, CRL 2001/01

Powered by
iveaStudio



Copyright 2005 **Zhao Nan**, ivenan@21cn.com · All Rights Reserved
Powered by ivenaStudio, 2005 · **FaDét** Boosting Version 1.00

