

学校代码： 10246
学 号： 081021041

復旦大學

博 士 学 位 论 文

人脸检测算法及其芯片实现关键技术研究

院 系： 信息科学与工程学院

专 业： 微电子与固体电子学

姓 名： 周薇娜

指 导 教 师： 曾晓洋 教授

完 成 日 期： 2012 年 04 月 05 日

博士学位论文

人脸检测算法及其芯片实现关键技术研究

周薇娜

指导老师:

曾晓洋 教授

指导小组:

曾晓洋 教授

胡 波 教授

张建秋 教授

复旦大学信息科学与工程学院微电子系

2012.4

目录

目录.....	I
图片索引.....	III
表格索引.....	V
摘要.....	VII
Abstract.....	IX
第 1 章 绪论.....	1
1.1 人脸检测简介.....	1
1.1.1 人脸检测的研究意义.....	1
1.1.2 人脸检测系统的性能指标.....	3
1.2 人脸检测技术国内外研究状况.....	4
1.3 人脸检测面临的主要困难及今后发展方向.....	7
1.4 本文研究目标和主要研究内容.....	8
1.5 论文结构安排.....	9
第 2 章 基于 AdaBoost 的人脸检测算法及其实现分析.....	11
2.1 基于 AdaBoost 的人脸检测算法介绍.....	11
2.1.1 Haar-like 特征.....	12
2.1.2 积分图和平方积分图.....	15
2.1.3 级联分类器.....	17
2.2 AdaBoost 训练算法.....	19
2.3 基于 AdaBoost 的人脸检测.....	22
2.3.1 检测方法.....	24
2.3.2 检测流程.....	26
2.3.3 区域合并.....	26
2.4 本章小结.....	27
第 3 章 人脸检测算法改进及定点化分析.....	29
3.1 人脸数据库和检测算法性能评测.....	29
3.1.1 人脸数据库介绍.....	29
3.1.2 检测算法性能评测.....	32
3.2 方差预处理.....	33
3.3 图像缩小法人脸检测.....	40
3.3.1 图像缩小法与窗口扩大法的比较.....	40
3.3.2 两类主要的图像缩小法.....	42
3.4 高性能人脸检测算法定点化分析.....	44
3.5 本章小结.....	46
第 4 章 高性能人脸检测芯片研究.....	49
4.1 人脸检测芯片硬件架构.....	49
4.1.1 细胞阵列架构.....	49
4.1.2 帧存储器型阵列架构.....	52
4.1.3 架构比较与确定.....	54
4.2 高效存储方案.....	56
4.2.1 存储器结构与效能比较.....	56
4.2.2 图像存储与检测扫描策略.....	58
4.2.3 图像数据读写策略.....	60
4.3 积分图快速更新.....	61
4.3.1 积分图寄存器的组成与结构.....	61
4.3.2 积分图快速更新法.....	62
4.3.3 平方积分图的产生与更新.....	64

4.4	四级流水特征计算结构.....	66
4.4.1	常用特征值计算结构分析比较.....	66
4.4.2	四级流水处理结构.....	69
4.5	分类器选择与存储.....	71
4.6	图像缩小模块.....	72
4.7	人脸检测芯片的仿真、综合结果与比较分析.....	73
4.8	人脸检测系统的 FPGA 验证.....	75
4.8.1	人脸检测 SOC 系统介绍.....	76
4.8.2	FPGA 硬件平台.....	78
4.8.3	FPGA 软件平台.....	79
4.8.4	系统测试.....	80
4.9	人脸检测 SOC 测试.....	80
4.10	本章小结.....	82
第 5 章	多模式可调人脸检测.....	83
5.1	人脸检测芯片的可配置性研究.....	83
5.2	多模式可调人脸检测分析.....	84
5.3	多模式可调人脸检测模块.....	86
5.4	可调模式相关实验与验证.....	89
5.4.1	可调性测试.....	89
5.4.2	综合仿真结果.....	92
5.5	本章小结.....	92
第 6 章	结论与展望.....	93
6.1	对本文工作的总结.....	93
6.2	对未来工作的展望.....	94
	参考文献.....	95
	攻读博士学位期间发表的论文.....	101
	致谢.....	102

图片索引

图 1.1	人脸检测应用领域.....	2
图 2.1	基于 AdaBoost 的人脸检测算法流程.....	12
图 2.2	三类 Haar-like 特征.....	13
图 2.3	人脸中的 Haar-like 特征.....	14
图 2.4	积分图和矩阵面积计算.....	15
图 2.5	矩阵面积计算.....	16
图 2.6	级联结构分类器.....	18
图 2.7	AdaBoost 训练算法基本流程图.....	20
图 2.8	AdaBoost 人脸检测算法检测流程.....	24
图 3.1	CMU+MIT 中典型的包含人脸的图像.....	29
图 3.2	CMU+MIT 中绘制的假人脸图片.....	30
图 3.3	CMU+MIT 中不包含人脸的干扰图片.....	30
图 3.4	Bao 数据库中的典型的单人脸和多人脸图像.....	30
图 3.5	BioID 人脸数据库中典型人脸图像.....	31
图 3.6	Yale 人脸数据库的不同表情与条件下的人脸.....	31
图 3.7	ESSCX 数据库中的人脸.....	32
图 3.8	20*20 的不同方差的人脸图像.....	34
图 3.9	人脸方差统计结果（一）.....	35
图 3.10	人脸方差统计结果（二）.....	35
图 3.11	增加预处理后的级联结构分类器.....	36
图 3.12	方差预处理使用前后的人脸检测速度对比.....	39
图 3.13	检测窗口扩大法.....	40
图 3.14	图像缩小法.....	41
图 3.15	定点化的小数位数与检测率之间关系图.....	45
图 3.16	定点和浮点情况下检测率、检测速度比较.....	46
图 4.1	积分图存储阵列架构.....	50
图 4.2	检测窗口存储阵列架构.....	51
图 4.3	帧存储器型阵列架构.....	53
图 4.4	人脸检测芯片硬件架构.....	56
图 4.5	图像缓冲器结构图.....	56
图 4.6	存储方案的高效性比较.....	58
图 4.7	图像装载、检测方向.....	59
图 4.8	数据访存的几种正确和错误情况.....	60
图 4.9	数据处理模块图.....	61
图 4.10	积分图寄存器窗口与其同步访问.....	62
图 4.11	积分图产生与更新.....	63
图 4.12	传统方法与提出的快速更新法在 SRAM 访问次数上的比较.....	64
图 4.13	求平方值电路.....	65
图 4.14	平方积分图产生与更新.....	66
图 4.15	强分类器流水线并行结构.....	67
图 4.16	弱分类器特征值计算的四级流水结构.....	69
图 4.17	两种结构电路逻辑门比较.....	71
图 4.18	分类器缓冲器结构.....	72
图 4.19	图像缩小模块结构.....	73
图 4.20	人脸图片及其软件检测结果.....	74
图 4.21	人脸图片的 RTL 级仿真结果.....	74
图 4.22	人脸检测 SOC 结构框图.....	76

图 4.23	MIPS32 结构示意图	77
图 4.24	FPGA 开发平台	78
图 4.25	人脸检测芯片	80
图 4.26	CQFP-160 封装	81
图 4.27	人脸检测芯片测试平台（一）	81
图 4.28	人脸检测芯片测试平台（二）	82
图 5.1	多模式可调人脸检测系统结构图	87
图 5.2	多模式可调人脸检测系统的控制器及其状态机	89
图 5.3	使用不同个数的强分类器时 FAR 和 DR 的变化	90
图 5.4	使用不同个数的强分类器所需要的时间	90
图 5.5	多模式可调人脸检测仿真波形	92

表格索引

表 2.1	不同窗口特征总数量.....	14
表 2.2	每级所对应的 Haar-like 分类器数.....	22
表 3.1	对 CMU+MIT 数据库进行测试的比较结果.....	37
表 3.2	对 Bao 数据库进行测试的比较结果.....	37
表 3.3	对 BioID 数据库进行测试的比较结果.....	37
表 3.4	对 Yale 数据库进行测试的比较结果.....	38
表 3.5	对 ESSEX 数据库中单一背景的人脸进行测试的比较结果.....	38
表 3.6	对 ESSEX 数据库中复杂背景的人脸进行测试的比较结果.....	39
表 3.7	图像缩小法与窗口扩大法对 CMU+MIT 人脸数据库进行测试的比较结果....	43
表 3.8	图像缩小法与窗口扩大法对 Bao 人脸数据库进行测试的比较结果.....	43
表 3.9	图像缩小法与窗口扩大法对 BioID 人脸数据库进行测试的比较结果.....	43
表 3.10	图像缩小法与窗口扩大法对 Yale 人脸数据库进行测试的比较结果.....	43
表 3.11	图像缩小法与窗口扩大法对 ESSEX 人脸数据库进行测试的比较结果.....	44
表 4.1	人脸检测各架构的优缺点比较.....	54
表 4.2	几种架构比较.....	55
表 4.3	三分类器并行结构与四级流水处理结构的比较.....	70
表 4.4	综合结果比较.....	74
表 4.5	开发板硬件资源.....	79
表 4.6	EP2S180F1020 主要资源.....	79
表 4.7	主要软件平台.....	79
表 4.8	FPGA 资源使用情况表.....	80
表 5.1	每级所对应的 Haar-like 分类器数与需检测的子窗口数对照表.....	85
表 5.2	控制信号与可调的输入信号之间的关系.....	87
表 5.3	不同的 sl_num 和 sg_num 所代表的意义.....	88
表 5.4	不同缩小比例因子下检测准确度和速度.....	91
表 5.5	人脸检测对不同规格图片的软硬件检测比较.....	92

摘要

随着目标监控、目标跟踪、人机交互、人流统计等领域的兴起，人脸检测已不仅是人脸识别前期的一个重要步骤，它开始作为一门独立的技术受到了人们越来越广泛的关注。从数码相机、人机交互中的人脸定位，3G 业务的图像通信，到人物跟踪、警戒区报警都离不开人脸检测技术。但由于人脸检测需要进行的计算量大，检测速度的提高和精确度的保证成了相互矛盾的两个方面。而很多应用中对这两方面都是有所要求的。另外，要使人脸检测技术能在未来发挥重要作用，真正有效应用于各种应用场合，那么人脸检测芯片的功耗、成本、可配置性都是值得研究的，这同时也是现今人脸检测技术面临的难题。

本论文致力于研究高性能人脸检测芯片的关键技术，从算法的改进、结构的设计、可配置性等多方面提出了实现高精度高速人脸检测芯片的一些创新性想法。

考虑到算法的有效性是高性能检测的一个重要方面，本文对目前最为高效的基于 AdaBoost 的人脸检测算法进行了深入的分析，介绍了其训练与检测的具体方法。并提出了方差预处理与级联结构相结合的人脸检测算法，以及适合于硬件实现、减少图像访存频率的图像缩小法。然后对目前国际公认的五个人脸数据库进行了总结与归纳，用各个数据库的测试结果证明了改进后的人脸检测算法的有效性。并对算法进行了定点化分析，选择了合适的定点化位数。

本文还从检测速度的提高、芯片的面积消耗、功率消耗各方面综合考虑，提出了高效的存储方案设计、积分图快速更新法、四级流水线处理结构，对芯片内部各模块的划分、各模块的结构、它们之间的工作时序等都进行了分析研究。并对设计的高性能人脸检测芯片进行了仿真、综合、验证与测试。实验结果表明，算法上的改进以及芯片的上述设计技术使得人脸检测芯片可以达到在保证高精度高速的前提下，具有较低的功耗和面积，与性能居于国际前沿的其他人脸检测芯片的测试结果相比，具有其优势。

鉴于人脸检测芯片的可配置直接影响其未来多场合应用的灵活性和开发成本，本文又提出了多模式可调人脸检测的概念。从人脸检测四个关键因子着手，分析其对检测性能的具体影响。并使其作为能调节芯片工作于不同模式的四个输入参数，实现了芯片的参数可调性能。

人脸检测技术是一门值得不断深入研究的课题，该课题的开展能带来巨大的社会效益，其研究成果也能为其他类型的目标检测提供有意义的参考。

关键词：人脸检测，存储方案，积分图快速更新，四级流水线处理，多模式可调

中国图书分类号： TN47, TN492

Abstract

With the rise of fields such as target monitoring, object tracking, human-computer interaction, statistics of people stream etc., face detection has not only been an important step of face recognition, but also become an independent technology being concerned by more and more people. It now has become indispensable in applications such as digital camera and human-computer interaction, the image communication of 3G, the human tracking and alarm of security area. However, due to its huge computation, it is difficult to improve its speed and accuracy to satisfy the demand of many applications at the same time. And what's more, to really play its important role in future and to be efficiently applied to different situations, the promotion of power consumption, cost and processing ability of face detection chip are worthy to be researched. Nevertheless, these are just difficult problems we now confront.

The paper describes the research on the key technology of designing face detection chip with good performance, and puts forward some creative ideas in realizing face detection of high precision and speed from the side of algorithm improvement, architecture design and reconfiguration.

In considering that the efficiency of the algorithm is one of the most important factors in detection, the paper analyses the most efficient face detection method based on AdaBoost algorithm in depth. And after introducing its training and detection procedure, the paper proposes an improved algorithm integrating variance preprocess and cascade structure, and adopts the image scaling method for hardware realizing, which could greatly decrease the access of memory. Five commonly used face databases in the world are summarized then, and are used to test the efficiency of improved face detection algorithm. And the paper also gives out the fixed-point bits of every data in the algorithm for the concrete design of face detection chip later.

When considering the promotion of detection speed, and decrease of the power and area consumption, ideas of efficient memory scheme, quick refresh of integral image, 4-stage pipeline processing method are proposed. The partition, architecture and the work order of each module are all researched in the paper. And they are also simulated, synthesized, validated and tested by corresponding tools. The results show that, the improvement of detection algorithm and creative chip design method are efficient, which makes the design have a relative less consumption of power and area

compared to other advanced international designs, while maintaining a high precision and speed.

As the reconfiguration of the face detection chip will directly affect the flexibility in applications of face detection technology in different situations and the development cost, the paper also brings forward a mode adjustable design method. It mainly makes use of four key factors greatly affecting on the detection capability. And by adjusting these four factors as input parameters, we finally realize the reconfiguration and enable the chip work in different modes.

Face detection technology is a subject which is worthy to be researched continuously. It can bring with huge benefit to the society, and its research fruits can also provide meaningful reference to detection of other kinds of objects.

Key words: face detection, memory scheme, quick refresh of integral image, 4-stage pipeline processing, mode adjustable

CLC number: TN47, TN492

第1章 绪论

本章将首先介绍人脸检测技术的研究意义，指出对其的进一步深入研究是非常必要的。然后解释说明人脸检测系统的各项性能指标，为接下来阐述人脸检测技术的发展带来的各种性能上的变化做铺垫。根据人脸检测技术性能要求在各个发展时期的变化，又将人脸检测技术的发展历程分成四个阶段阐述，并对这四个阶段的代表性研究成果做了总结与归纳，指出目前人脸检测技术面临的主要困难以及今后的发展方向。之后阐述本文的研究目标与主要研究内容。最后引入本论文研究内容的结构安排。

1.1 人脸检测简介

1.1.1 人脸检测的研究意义

在高度信息化的现代社会，信息安全问题日益成为人们的焦点。传统的密码、智能卡等身份鉴别方法由于其易遗忘或易假冒等缺点而远远不能符合现代社会的要求。加之随着近些年来国际形势的变化，特别是 9.11 事件之后，各国对自身重要设施的安保性能越发重视，生物识别技术以其独有的不可替代性而得到极大的发展和日益广泛的应用。

生物特征识别采用人体所固有的生物特征来辨识与验证身份，具有较高的便捷性、精确性、唯一性和安全性，在可靠解决个人身份认证方面具有得天独厚的优势，受到了现今国内外学术界和产业界的高度重视。

目前，可用作身份认证的人体生物特征识别有人脸识别、指纹识别、虹膜识别、掌纹识别、步态识别和语音识别等等。他们采用不同的传感器采集人体独有的特征，作为识别标志鉴别人的身份。其中，人脸特征识别与其它生物特征识别方式相比，具有以下四个优点：

- (1) 人脸所涵的信息量相对其它的人体生物特征来说更为丰富；
- (2) 人脸特征信息可以直接应用，并且不容易盗用或者假冒；
- (3) 人脸特征具有较为友好的人机界面，易于为用户所接受；
- (4) 人脸特征采集所需设备简单，采集器摄像头的应用已比较普遍，为大面积的特征识别应用提供了很好的前提。

因此人脸特征识别自提出以来一直是个研究热点。但对人脸特征识别的研究其实不仅包括人脸识别(Face Recognition)技术的研究，还包括人脸检测(Face Detection)技术的研究。

人脸检测技术是指从一幅给定图像中确定其是否含有人脸，并对人脸位置与

范围进行定位的过程。其正确性能有效提高人脸识别的效率和速度，直接影响人脸识别结果。早期的人脸研究主要集中在人脸识别领域，主要针对具有较强约束条件，如无背景的人脸图像，而且往往假设人脸位置已经或者是人脸很容易获得的前提下进行的，人脸检测问题并未受到重视。但是随着人脸应用范围的不断扩大和实际要求的不断提高，这种假设下的研究不再能满足需求了，人脸检测逐渐作为独立的研究课题发展起来。不仅如此，随着人脸检测技术的发展、社会的进步，人脸检测已不仅仅服务于人脸识别，它在许多其他领域都起到了举足轻重的作用。在金融、司法、海关、军事及人们日常生活等各个领域中都扮演越来越重要的角色，发展速度异常迅猛。

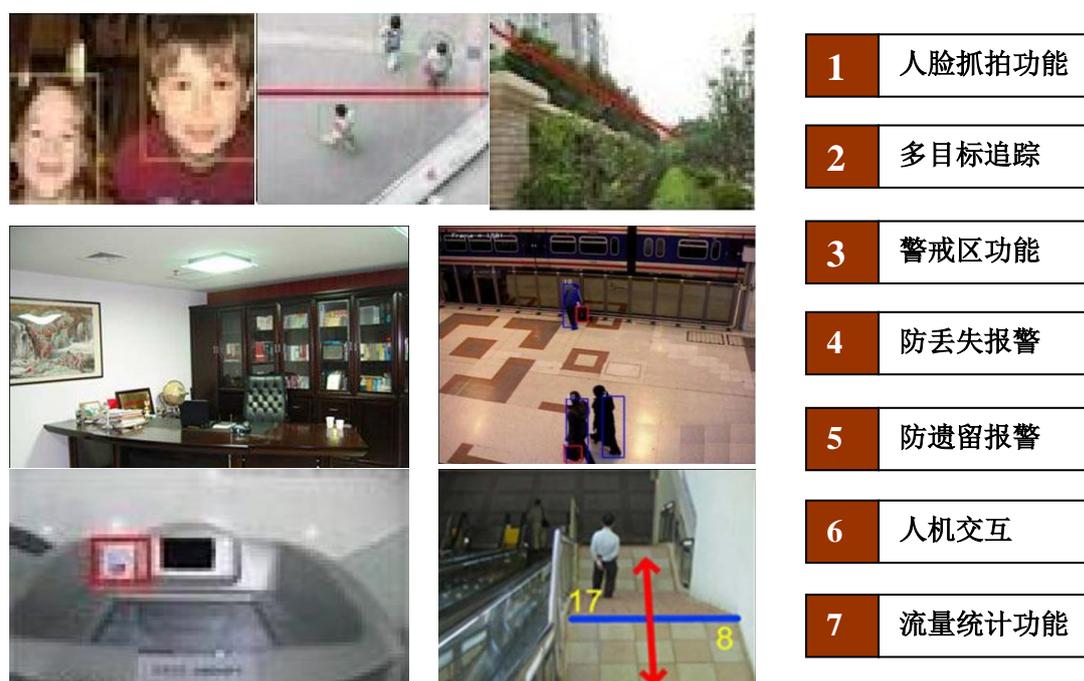


Figure 1.1 Application fields of face detection technology

图 1.1 人脸检测应用领域

例如在数码相机中，人脸检测可以使相机的聚焦、曝光更为有效，更具有针对性，提高拍摄的质量。在人机交互、智能机器人[1]领域，人脸检测作为机器辨识人类目标的手段，是必不可少的。3G 业务飞速发展，精确定位人脸，使人脸区域重建质量好于背景区域重建图像，可在有限的通信带宽下提高通话质量[2-3]。而在许多视频监控场合，由于要作视频录像的备份，在采用了人脸检测技术之后可以只需要存储人脸区域的图像。如果在此基础上进一步采用相关的特征提取技术，则可以极大的减小对硬盘存储空间的消耗。如果用于网络传输则可以显著的降低对网络带宽的占用。另外在监控与查缉领域[4-7]，人脸检测可用于人物跟踪、警戒区报警、人流计数、广告关注度统计等等。另外，还可以通过对检

测出的人脸中人眼形状的分析,进行驾驶员的疲劳检测[5]。由此可见,人脸检测不仅是各种人脸处理系统中一个重要的前提技术条件,在诸多其他领域也都具有潜在的应用价值,具有广阔的应用前景。许多国内外的文献、重要的国际会议中也都涉及到人脸检测问题的研究,其在未来的重要性可见一斑。

从理论意义上看,人类与生俱来的人脸检测能力至今没有明晰的生理学、心理学解释,人脸检测的研究将有利于推进人类对自身视觉过程的认知。另外,人脸检测作为目标检测领域中的一个难题,对其进行研究,将有助于模式识别、图像处理、计算机视觉等相关学科的发展,同时也可为其它目标检测的研究提供重要的理论借鉴。

另外这里要指出的是,人脸检测可细分为两类:一类是在静止图像中采用一定的策略进行搜索,以判断其中是否存在人脸,若存在,则返回人脸的位置、大小和姿态信息,这也正是本文要研究的问题;另一类是在视频图像序列中判断是否存在人脸,若存在,则动态地跟踪人脸,本文不讨论动态人脸跟踪。

1.1.2 人脸检测系统的性能指标

人脸检测系统的性能指标主要包括准确度、速度、便携性、可配置性、低成本、低功耗等。

这几个指标的具体解释如下:

- (1) 准确度(Detection Accuracy, DA)反映了检出来的人脸的正确性,准确度较高代表检测出的结果确实是人脸的比例很高,而非人脸被误认为是人脸或人脸没有被检测出来的比例很低。
- (2) 速度(Detection Speed, DS)反映了检测工作完成的时间。速度越快,所用的检测时间越少。
- (3) 便携性(Portable)是指人脸检测系统体积较小,且一般具有较低的功率损耗,方便携带移动。
- (4) 可配置性(Configuration)反映了人们对人脸检测系统更高的期许,希望通过简单的重新配置实现具有不同性能的人脸检测系统。
- (5) 低成本(Low Cost)是指人脸检测使用的硬件资源比较少,这是人脸检测系统未来在各个领域大规模存在的一个前提。
- (6) 低功耗(Low Power)是指系统工作时消耗的功率少,比较节约能量。

一般而言,没有一个人脸检测系统能同时拥有上述所有最优的性能。它们之间通常存在着互相制约的关系。例如,提高速度往往会以降低检测准确度或增加成本为代价,所以通常人们总是需要根据不同应用系统的具体要求来设计人脸检测系统的指标参数。比如,用于监控、警戒的系统首先要求具有较高的准确度;而像跟踪、人机交互等系统则偏重较快的检测速度,具备实时性等。但随着各领

域的飞速发展，各指标的整体趋势都是在不断改进提高的。

1.2 人脸检测技术国内外研究状况

人脸检测技术的研究可以追溯到 20 世纪 60-70 年代[6]，通过对人脸检测文献的详尽分析后会发现，其发展过程大致是按照其逐步走向实用的规律发展的。因为任何技术的发展都离不开人们生活发展对其提出的具体需求。我们可将其过程大致划分为四类：初始期，发展期，转折点和成熟期进行阐述。

在初始阶段，人脸检测应用范围小研究以检测准确度提高为主，各种视角的人脸检测是其主要研究内容，对别的性能考虑较少。如 MIT 的 Sung 等提出的基于样本学习的方法[7]。算法采用 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，利用训练样本到各个簇中心的距离，训练一个多层感知器进行分类，用来实现人脸的检测。其中的“非人脸”样本的收集方法后来被许多研究者沿用。M.H.Yang 提出了一种基于 SNoW (Sparse Network of Winnows) 的学习体系，用于检测具有不同特征、表情、姿态和光照条件下的人脸[8]。检测结果与采用神经元网络、贝耶斯分类器以及支持向量机等方法比较，在准确度方面效果很好。H. Schneiderman 等[9]提出了一种基于后验概率估计的方法来对三维的物体进行检测，也取得了较好的检测效果。

之后由于其安全监控、自动跟踪等多应用领域的发展，人们慢慢意识到了检测速度的重要性，对速度问题进行了考虑，人脸检测进入了发展期。这一时期的人脸检测采用了诸如选择计算量小的特征检测[10-11]、由粗到精分层处理[12-13]、多线程法[14]、降低特征空间维数[15]或其他各种手段[16-17]，尽量减少总的计算量，达到速度上的提升。

利用肤色、边缘等图像的基本信息检测人脸属于选择计算量小的特征检测，但是单独用这些方法对速度的提高很有限，而且往往会损失检测的准确度。在这方面所作的研究有：如 Michael J. Jones 等人基于大量的样本图像对肤色检测问题进行的统计分析，比较了用直方图模型和混合高斯模型进行肤色检测的情况[13]；B. Martinkauppi 对 Skin locus 的理论和方法在不同光照条件下的肤色分布问题作了细致的分析[14]；Terrillon JC 等分别使用了高斯模型和混合高斯模型在不同颜色空间中建立肤色模型来进行人脸检测，并将其结果进行了对比[15]；H.Kim 等利用人脸的边缘轮廓相似于椭圆的特性，建立椭圆模板进行椭圆匹配来检测人脸[20]。Wang 等又对椭圆模板进行了改进，采用一个椭圆环模型来代替椭圆，并加入了边缘的方向特征，实现了简单背景的人脸检测[21]。

为了提高准确度，又出现了将肤色与边缘等简单的特征与其它特征检测方法进行结合，利用简单的特征对人脸位置进行初定位，然后再用其它的特征去除

初检结果中的假脸,这样可以既提高检测的速度又保证了检测的准确度。这方面的例子有: Dorin Comaniciu 使用非参数的核函数概率密度估计法来建立肤色模型,并使用 mean shift 方法进行局部搜索实现了人脸的检测和跟踪[16-17]; R. L. Hsu 为了解决光照问题,提出了针对不同光照进行补偿,然后再检测图像中的肤色区域等[18]。

而由粗到精的搜索策略,其思想是利用加大检测步长,有的学者还利用了人的头部轮廓特征进行分层处理。如 Craw 等首先在低分辨率图像中使用一个轮廓模板匹配出人脸的大致范围,然后在高分辨率图像中使用 Sobel 算子获得边缘的位置和方向,并连接出完整的人脸轮廓[26]。H. Schneiderman 等也使用了多分辨率的信息复用和由粗到精的搜索策略提高检测速度[24]。在这方面做出贡献的还有通过采用分级的方法减少检测过程中每级的计算量以加快速度的方法[27-28],采用神经网络系统实现快速的人脸检测等[29]。

另外, Shpungin 使用了多线程的方法对两个模块进行并行处理,这样系统能够及时把处理器资源分给不同的模块,从而在一定程度上解决了速度问题,使系统达到了实时[30]。C. J. Liu 提出了一种基于 Bayes 的特征判别人脸检测方法,并采用 PCA 进行降维以减小计算量[31]。此外,还有一些学者采用其它算法来达到提高速度的目的。如 C. Sanderson 提出了快速的特征提取方法[38], Edgar Osuna 提出了减少 SVM 在运行时的时间复杂度[39], Raphael Feraud 的基于神经网络的人脸检测系统[41]等等。

经过发展期众多学者们的共同努力,人脸检测的速度有了一定的提高,但相对于人脸检测的准确度而言进步相对较慢。2001年, Viola 和 Jones 提出的 AdaBoost 级联算法被认为是人脸检测速度提高的一个转折点,在速度极大提高的前提下,检测率达到了当时的最高水平,使得人脸检测从真正意义上走向实用[18]。之后的很多学者在该基础上又提出了一些改进方案[19-20],使得检测率和检测速度进一步得到了提高。如 Rainer Lienhart 在 Viola 的基础上又提出了一些新的旋转 Haar-like 特征[43-44]。Stan Z. Li 提出了一种基于 FloatBoost 的多视角人脸检测算法[45-47]。C. Liu 等人提出了一种基于 Kullback-Leibler Boosting (KLB)算法构造的一个紧凑的分类器[48]。该算法解决了 AdaBoost 算法中存在的问题。P. Viola 本人也在接下来的工作中对 AdaBoost 进行了一些改进[49],提出了一种不对称的 AdaBoost 算法,可以用于快速的图像检索和人脸检测。此外还有 Zhang 提出的基于 Boosting 的多视角的人脸检测[50], Sahbi 提出的综合 SVM 和分级的方法进行人脸检测的系统[51-52]等。

但直到这一转折时期,还未考虑或较少考虑到检测带来的资源和功率消耗,速度的提高往往是以硬件资源或功耗的增加为代价。如法国的 Khattab 等[21]发

现顺序执行算法时，数据处理时间远远不如存储器访问时间。因此，提出将整个处理过程分成三个对存储器访问次数大致相等的部分，以并行执行减少处理时间。但是由此导致了存储器数量的增加。Cyprus 大学 Theocharides[22]一直致力于人脸检测研究，2006 年他提出了一个阵列型布局的人脸检测结构，可以并行对所有人脸特征进行提取和验证计算，达到了较高的检测率和检测速度。但是这种结构也需要消耗大量的硬件资源。

随着人脸检测在各领域的进一步深入发展，以及出于能源节约，系统小型化、便携化发展趋势的考虑，对人脸检测的硬件和功率消耗的要求也渐渐地高了起来。而且有些设计也考虑了可配置性以节约开发成本。人脸检测由此被推入了成熟期阶段。但这方面的人脸检测技术研究还处于起步阶段，研究成果不多。卡耐基梅隆大学的 Lai 提出了一个高速的人脸检测硬件架构[23]，同时考虑了存储器大小的设置和处理的并行性。而 Hiromoto[24]对算法进行了深入分析，研究了多种并行方案、不同图像的缩小法以及定点化对检测速度和资源消耗的影响，提出了一个部分并行的人脸检测架构。加利福尼亚大学的 Junguk[25]提出的人脸检测系统采用寄存器存储人脸检测窗口，加快检测速度，并且采用流水线数据处理方式，比较了不同个数的特征并行带来的速度上的变化。上海交通大学施跃华、赵峰[26-27]提出的细胞阵列结构，采用双向流水结构实现图像中每个检测窗口内的特征的快速检测，并且提出可多窗口并行以提高检测速度。这些方案，虽然没能使检测的功耗和硬件资源得到大幅的下降，但是对人脸检测的速度、功耗、硬件面积有了较为深入的分析，为进一步的发展改进做了贡献。

之后的人脸检测技术继续发展，主要分化为了两个方向，一个方向还是以保证准确度与速度为条件，在功耗、面积方面只有少许改进。偏重这一方面的芯片性能较好，但是成本较高功耗又大。如 Theocharides[28]在 2010 年提出的使用金字塔产生模块与阵列结构相结合，用阵列的高度并行性处理分解后的人脸图像，与其前期研究相比，在减小了资源消耗面积的同时保证了检测率和速度，也具备了一定的可配置性，但是功耗和资源面积消耗还是较大。另一个方向是重点关注功耗和资源面积的降低，而一定程度牺牲了别的检测性能。如南洋理工大学的 Wei 等[29]实现了较少资源下的近实时人脸检测，速度可以达到 15 帧/秒。然而其针对的只是分辨率为 120*120 的较小图像，并且也牺牲了一定的检测率。Yang 将人脸检测用于手持数码相机[30]，为了减少硬件消耗和实现实时检测，简化了算法，并且采用一个复杂的控制机制充分挖掘了算法的并行性，实现了一个低成本检测系统，检测速度也达到了 13 帧/秒，但检测率却由原算法 90%以上降到了 75%。日本的 Yuichi 等提出了一个只有 0.79 平方毫米，29 毫瓦的实时人脸检测芯片[31]，但该设计是基于皮肤边缘检测算法，且只针对彩色图像，光线的变化

对其皮肤色彩的影响会极大的降低其检测率，其使用范围有一定的局限性。日本的 Hanai[32]在 2009 年提出的通用识别处理器可以检测图像、视频、声音多种信号，实现了低功耗、低成本的检测，检测速率可以达到 8 帧/秒，但其只针对 320*240 的图像，检测率也只有 81%。Wong[33]和 Chen[34]在 09 年和 10 年分别在会议和期刊上提出了采用“奇偶场”的方法进行人脸检测窗口的选择，并且采用特征并行，用灰度值的高四位表达像素，大大减小了图像等信息的存储量，在资源消耗上得到了显著地降低。但其处理的也只是 160*120 的小图，使用受到极大限制。

值得指出的是，后期的人脸检测算法基本也都是基于 Viola 和 Jones 提出的 AdaBoost 级联算法，因为其不仅在准确度和速度上占有优势，而且也比较适合在硬件上的实现。

综合以上分析，人脸检测技术目前已经进入了相对成熟的状态，以纯粹提高检测某一指标的研究已失去了实际的意义，不是我们未来的发展方向。在保证准确度与速度的同时，能兼顾成本和功耗是人脸检测今后的一个总的发展趋势。但这方面的研究目前还不够成熟，需要进一步的深入。

1.3 人脸检测面临的主要困难及今后发展方向

通过前期多年的研究发现，人脸检测的几个性能指标之间总存在着一些相互制约的关系。当我们需要进一步提高准确度的时候，发现它的提高往往会带来算法复杂度的大大增加，两者大致是一个成正比的关系，因此会造成速度的降低或功耗、成本的增加。而如果要降低功耗，节约硬件资源，又往往要牺牲准确度或速度为代价。如何达到较优性能，是今后面临的第一个难题。

另外，因为人脸检测应用范围极广，如针对每一应用重新加以设计会浪费大量的人力、物力。已有的少量可配置方面的研究成果，可配置性也还较弱。目前检测主要还是利用高频处理器如高性能 DSP 等多核处理器来处理数据。但受限于其昂贵的成本，及极大的功率消耗，并不利于未来在各领域的大规模推广应用。一方面不符合人类绿色节能的理念，消耗了大量自然界有限的能源，另一方面也因为受制于固定的大功率供给电源，其活动范围大大受限，不能满足移动便携式应用场合的需要。如何在保证一定准确度下，实现低功耗、低成本的实时人脸检测，并使其能够适合多个应用背景是当前我们面临的另一个很大的困难。

仔细分析造成这些难点的原因，发现问题主要在于以下几个方面：

一、图像与音频等其他信号相比，数据量较大，而且随着采集器分辨率的提高，数据量还在增长。要对所有的图像数据进行处理，就需要不断地进行存储器的读写访问，而存储器访问时间直接影响了人脸的实时检测，消耗巨大的功率。

二、人脸检测算法要在复杂的背景下得到较高的检测准确度，需要对人脸目标进行多个特征的验证，计算量非常大。大计算量造成了速度提高和功耗降低上的困难。

三、片内存储器可以提高数据的读取效率，提高对大量数据的处理速度。但片内存储器消耗硬件资源，较多的片内存储空间往往是造成芯片高成本的一大原因。

四、检测计算量大，为加快检测速度，并行或流水线结构是经常采用的手段。但是并行程度过高或者流水线过深会占用较多的硬件面积，而两者过低又不能较好地加快速度。如何权衡也是一个难点。

五、ASIC 设计与 FPGA 等可编程器件相比，在功耗和面积上占有优势，值得采用，但是其可配置性灵活度一般较弱，不能满足很多移动便携设备的要求。

如何克服以上几个方面的难点，是人脸检测技术得以在各领域继续发展的关键，也是今后的发展方向。

1.4 本文研究目标和主要研究内容

鉴于人脸检测还是需要首先保证准确度和实时性，因为不能准确定位人脸或检测时间太长，即使别的性能很好也是毫无意义的。这就意味着我们必须要在同一时间能处理大量的人脸数据，在此基础上进一步降低功耗、成本，增大灵活性，满足移动便携设备要求。因此，本文的第一个研究目标就是设计能在保证一定检测准确度与速度的前提下，使人脸检测消耗的功耗和成本尽可能少的高性能人脸检测芯片。二是使其具有可方便灵活调整检测性能，节约开发成本，实现多领域灵活应用。但正如前所述，人脸检测各指标之间有一定的制约关系，要实现以上目标还存在着不少困难，因此本文拟从检测算法本身、片内存储器的设计、数据处理结构等各个方面深入研究。以这些问题的分析解决为重点而展开。

具体研究内容如下：

- (1) 深入分析人脸检测算法，并做出改进，在保证准确度的同时，合理调整算法，减少数据运算量，加快人脸检测速度。
- (2) 提出高效的数据存储方案，尽可能减少大量数据访问所带来的功耗的提高和速度的减慢。
- (3) 研究处理速度快、功耗低的数据处理结构。数据处理结构大致可分为串行、并行和流水线处理三种。要深挖算法中的并行因素，对无法并行处理的数据，采用流水线的结构，充分利用数据并行、流水带来的速度上的提高。但同时需考虑这些结构对功耗影响。
- (4) 研究灵活方便的可配置结构，使得人脸检测的性能能根据实际的应用需求而

改变，适应将来多领域的综合应用，减少多次开发所带来的开发成本。

本文对人脸检测技术的一系列系统研究能为人脸检测的发展提供更广阔的空间，大大增加其灵活性，具有直接的现实意义。而且也能为其他类型的目标检测提供有意义的参考。

1.5 论文结构安排

本论文共分6章。

本章是第1章，即绪论部分，主要阐述人脸检测技术的研究意义，发展历程，面临的主要困难以及解决方法，今后的发展方向。最后引入本论文研究工作的主要内容。

第2章介绍适合于硬件实现的 AdaBoost 级联检测算法的概念、原理、优势等。详细介绍其进行人脸检测时的训练方法以及检测过程。

第3章对人脸检测算法进行进一步深入分析与改进，减少检测计算量，从算法角度提高芯片的性能。并对用于检测准确度分析的人脸数据库和检测算法的几个性能评测指标进行了介绍，便于对优化后的人脸检测算法进行性能评估。

第4章是本论文的主要工作之一。介绍了高性能人脸检测芯片设计的各个方面。包括架构的选择、并行度的分析、积分图快速更新法和四级流水线处理结构的提出、分类器的选择与存储、高效存储方案的设计。并对设计的高性能人脸检测芯片进行了不同数据库的检测准确度测试，给出了 FPGA 的验证结果以及综合仿真与比较结果。

第5章也是本论文的主要工作之一。提出了多模式可调的人脸检测芯片。介绍了多种模式的实现以及各种不同模式的芯片的性能。最后介绍其综合仿真结果。

第6章结论部分，对本论文工作进行总结，并展望下一步工作。

第 2 章 基于 AdaBoost 的人脸检测算法及其实现分析

本章首先介绍了目前最为高效、应用最为广泛的 AdaBoost 人脸检测算法，该算法采用级联结构，是实时人脸检测算法中能做到较高检测准确度的算法之一。然后对取得关键检测特征的 AdaBoost 训练方法以及检测的整个过程做了深入阐述。

2.1 基于 AdaBoost 的人脸检测算法介绍

正如上一章中所提，在检测的实时性能得到重视以后，人们提出了各种方法来加速检测过程，包括利用肤色、人脸轮廓等等。但是绝大多数会同时引起检测准确度的下降。基于 AdaBoost 的人脸级联检测算法是由 Viola 和 Jones 在 2001 年提出来的一类基于统计理论的方法[25]。该算法不仅在检测速度和准确度上占有优势，而且运算简单、富有规律和具有一定扩展性，也比较适合硬件实现。目前各类实时人脸检测系统基本都是基于该算法。

该算法主要分为分类器训练和模式识别两大部分。这两个部分将分别在 2.2 和 2.3 小节中详细介绍。人脸检测经过训练，可从一个很大的特征集中选择很小的一部分关键特征，产生一个极其有效的分类器。而之后在识别部分就利用这些分类器将人脸与其他背景部分分离开来，最终确定人脸的大小和位置。图 2.1 显示了基于 AdaBoost 的人脸检测算法的整个流程图。

此算法主要有以下几点贡献：

- (1) 第一次引入了“积分图”的概念，使得检测器中的特征计算非常之快。
- (2) 使用 AdaBoost 的学习算法训练人脸检测的分类器。它能从一个很大的特征集中选择很小的一部分关键特征，产生一个极其有效的分类器。
- (3) 提出级联(cascade)检测器的概念。级联检测器类似具有能集中注意力的机构，能很快排除背景区域，从而节约时间用于更像人脸区域的计算。

以下就此几个部分作详细说明。

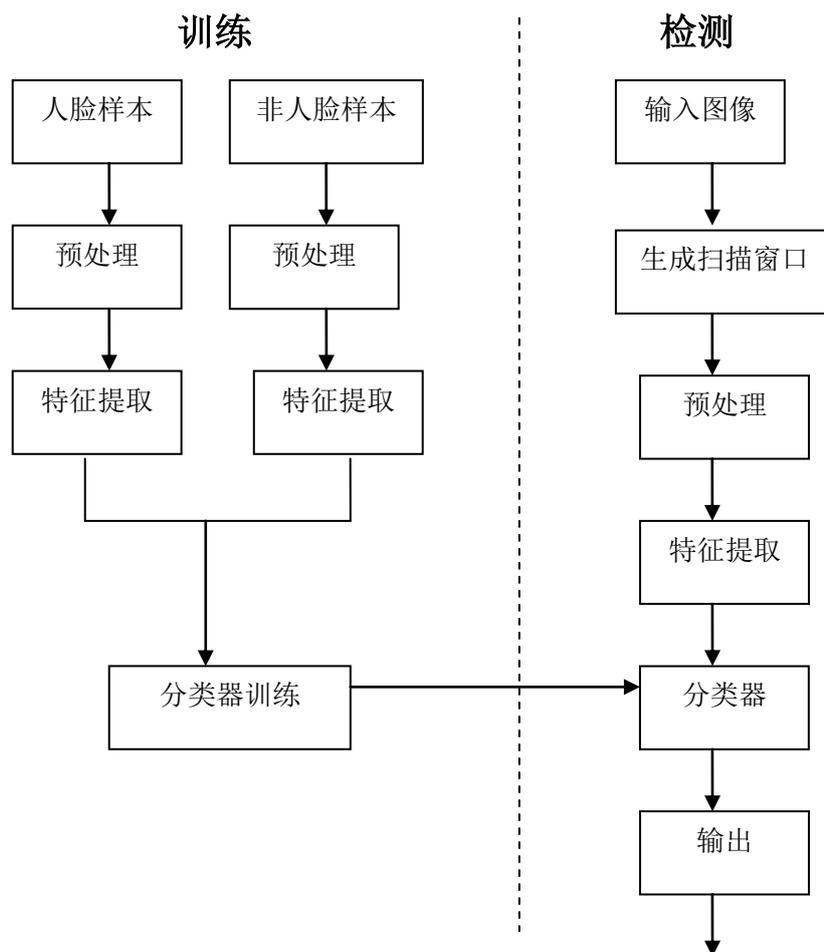


Figure 2.1 Face detection algorithm flow based on AdaBoost

图 2.1 基于 AdaBoost 的人脸检测算法流程

2.1.1 Haar-like 特征

基于 AdaBoost 的人脸检测算法的训练和识别过程都离不开“特征”。特征一般是通过需要对需要分类的对象所拥有的信息进行编码所得到的，对于人脸图像而言，特征的优点在于比像素更有利于人脸的表示，且这些信息从图像的原始像素中很难得到。使用特征对人脸图像进行建模，将有利于减小待分类对象的类内距离，同时还可以增加待分类对象的类间距离，这样就更有利于建立更有效的判别函数，使得最终的分类结果更好。对于利用统计学习方法的分类器，由于训练样本数有限，所以对于复杂的样本进行一次概括，对取得好的训练器效果是很重要的。此外使用特征表示人脸还表现在其更容易形式化，比直接利用图像点灰度值的速度快，更加有利于促进人脸检测速度的提高等，所以在人脸检测中特征的设计始终扮演着重要的角色。

Haar-like 特征是 Paul Viola 提出的一种简单矩形特征，因类似 Haar 小波而得名。它使用 AdaBoost 训练算法根据大量人脸正负样本，从庞大的特征库中选取人脸特征。描述的是相邻的两类矩形区域间的灰度差，反映了图像局部的灰度变化。这种特征表示的形式简单，有利于快速计算，同时形式多样。

从最终选取的结果看，Haar-like 特征非常抽象，实际意义很难与人脸联系在一起。而实际上，基于 Haar-like 特征的 AdaBoost 算法是一种通用的目标检测算法，人脸检测只是它的一个典型特例。图 2.2 是典型的三大 Haar-like 特征，分别为边缘特征、线特征、特定方向特征。其中边缘特征反映了图像边缘的信息，线特征反映了图像在线性方向上的信息，特定方向上的特征反映了图像在特定方向（如对角方向）的信息。

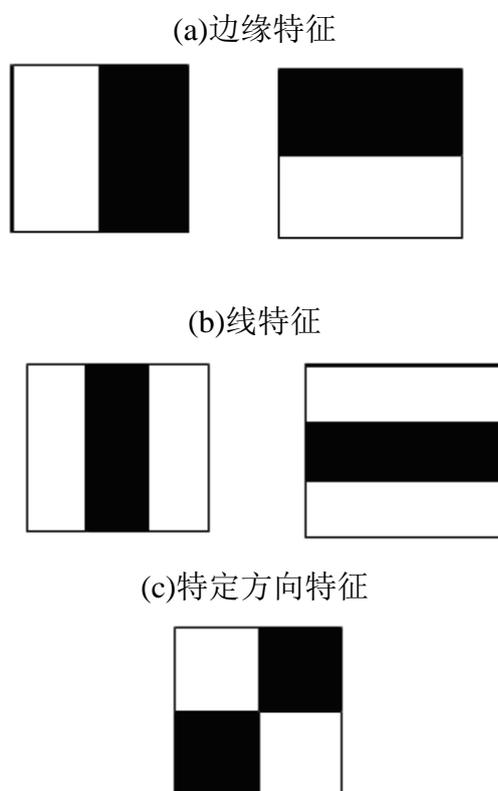


Figure 2.2 Three kinds of Haar-like features

图 2.2 三类 Haar-like 特征

特征值定义为白色矩形区域的像素和减去黑色矩形区域的像素和。Haar-like 所有矩形的权重和面积之间有如下的关系：

$$\sum_{i=1}^n w_i s_i = 0 \quad (2.1)$$

其中 w_i 和 s_i 分别是第 i 个矩形的权重和面积， n 是 Haar-like 特征包含的矩形的总数。实际计算时，白色矩形的权重取正值，黑色矩形的权重取负值。

Haar-like 特征对于一些简单的图像结构，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，比较粗略。如图 2.3 中，脸部一些特征能够由 Haar-like 特征简单的描绘，例如，通常眼睛比脸颊颜色更深，鼻梁两侧比鼻梁颜色更深，眼睛比中间鼻梁颜色更深等等。



Figure2.3 Haar-like feature in human face

图 2.3 人脸中的 Haar-like 特征

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。

在不同大小的子窗口内，三大类 Haar-like 特征的总数量如表 2.1[35]。

表 2.1 不同窗口特征总数量

Table2.1 Feature numbers in windows of different sizes

窗口大小	16*16	20*20	24*24	30*30	36*36
特征数量	32384	78460	162336	394725	816264

实际上，我们可以很容易的扩展出更多形状的 Haar-like 特征，但是这会使特征数目发生爆炸式的增长。太多的特征会使检测器的训练时间无限的延长，而训练得到的分类器性能却不一定大幅提高。另外，扩展出来的 Haar-like 特征极有可能是上述三大类 Haar-like 特征集里某些特征的组合，所以在已有的 Haar-like 特征集上再扩展其他形状的 Haar-like 特征没有太大的意义。

Haar-like 特征数目庞大，如果每次计算特征值都要分别计算各矩形内的像素和，那么计算这些 Haar-like 特征将相当耗时，严重影响训练和检测的速度。但其实，Haar-like 特征值的计算看似复杂，其实又十分简单。它的计算本质上是矩形灰度和的计算，因此 P. Viola 又提出了快速计算矩形灰度和的方法[25]，即积分图(Integral Image)的概念。

2.1.2 积分图和平方积分图

积分图与原始用像素表示的图像相比，仍然是一个数字矩阵，只是原始输入图像的一个变形，积分图中每个位置的值都是其左上角区域所有像素值的总和，其具体定义如图 2.4 所示。

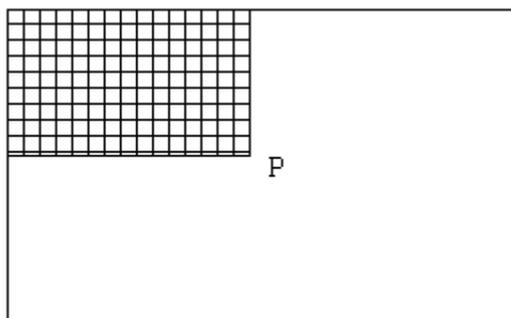


Figure 2.4 Integral Image representation

图 2.4 积分图和矩阵面积计算

设积分图为 Π ，原图为 I ，则位置 $P(x,y)$ 处的积分图值为

$$\Pi(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y') \quad (2.2)$$

其中 $\Pi(x,y)$ 表示像素点 $P(x,y)$ 处的积分图值， $I(x',y')$ 表示图像中 $P(x,y)$ 左上方所有像素点的灰度值。图像中所有像素点的积分图可以通过公式(2.3)和公式(2.4)迭代计算得到。

$$\Pi(x, y) = \Pi(x-1, y) + S(x, y) \quad (2.3)$$

$$S(x, y) = S(x, y-1) + I(x, y) \quad (2.4)$$

其中， $S(x,y)$ 是点 $P(x,y)$ 所在的 y 分量上所有图像像素灰度值的和。按照上述算法计算出图像中所有像素点的积分图值后，就可以在很短时间内计算出某个矩形特征的像素和。利用积分图技术可使得求任意大小的矩形区域像素灰度值之和只需三次加法运算，大大加速了 Haar-like 特征的求解过程。如图 2.5 中的区域 D ，利用积分图，可知其区域内的像素和为 $\text{Sum}(D) = P4 + P1 - P2 - P3$ ， $P1$ ， $P2$ ， $P3$ ， $P4$ 为积分图四点位置上的值。

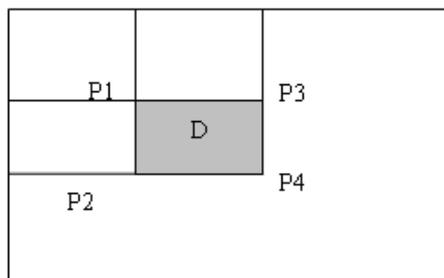


Figure 2.5 Rectangle area computation

图 2.5 矩阵面积计算

而且由此可见，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，而且都只是简单的加减运算。所以，积分图的引入，大大地提高了检测的速度。我们也可以进一步得出，图 2.2 中的边缘特征，其像素和之差可通过六个参考矩形求得；线特征可以通过八个参考矩形求得；而特定方向特征可以通过九个参考矩形求得。

另外，为了消除分类器对光照的敏感，高性能的人脸检测往往还要利用方差进行特征阈值的调整。调整方式如公式 2.5 所示：

$$\theta = \theta_0 * \sqrt{Var} \quad (2.5)$$

$$Var = \sum_{i=0}^{M*N} p_i^2 / M * N - (\bar{E})^2 \quad (2.6)$$

$$\bar{E} = \sum_{i=0}^{M*N} p_i / M * N \quad (2.7)$$

θ_0 为调整前的阈值， θ 为调整后的阈值， Var 为检测窗口的方差。 P_i 是 i 位置的像素值， M ， N 为检测窗口的大小， \bar{E} 为检测窗口的像素平均值。

根据如上介绍的积分图，公式 2.7 的均值是很容易通过检测窗口最右下角的积分值，即窗口内所有像素的和 $\sum_{i=0}^{M*N} p_i$ 得到。同样的道理如果我们也能知道窗口内所有像素平方值之和 $\sum_{i=0}^{M*N} p_i^2$ ，公式 2.6 的方差值也就不难计算了，由此提出了平方积分图的概念。

平方积分图类似积分图，每一点的值保存为该点左上方所有像素灰度值的平方和。其定义如下：

$$SII(x, y) = \sum_{x' \leq x, y' \leq y} [I(x', y')]^2 \quad (2.8)$$

如此，计算一个矩形区域的像素灰度值平方和可以极快的速度完成。像求像素和一样，计算某一矩形区域内的像素灰度值平方和，只需要读取其四角顶点的积分平方和就可以了。

平方积分图的引入可以快速计算方差，从而显著消除光照变化，使特征的分类能力更强，分类器需要的特征数更少。但是在实际计算中也发现平方积分图的引入会增加浮点数乘法运算，消耗了大量的检测时间。因此在对速度要求的苛刻程度远超过准确度要求的场合，可以舍弃平方积分图的使用。但是随着对检测性能要求的不断提高，一般认为平方积分图的使用还是非常必要的。

2.1.3 级联分类器

Haar-like 特征的集合构成分类器，综合分类器中每个 Haar-like 特征的特征值就可以判断一副图像是否是人脸图像。不同的 Haar-like 特征在分类器中的影响是不同的。AdaBoost 训练算法为每个 Haar-like 特征设定一个阈值 θ 和两个正负影响因子。当计算出的 Haar-like 特征值大于阈值 θ 时，取正影响因子，反之则取负影响因子。检测算法累计分类器中所有 Haar-like 特征的影响因子，并与阈值 β 作比较。如果和大于 β ，则判定图像是人脸窗口，反之则图像不是人脸窗口。

人脸检测是通过大量 Haar-like 特征的组合来提高精确度，通常需要数百到数千个特征。这导致人脸检测的计算量十分大。因此 P. Viola 同时提出了使用级联分类器的方法实现快速人脸检测。它的基本思想是将分类器按照先简单后复杂的顺序排列，用分类器级联完成人脸检测。通过分类器中各 Haar-like 特征的影响因子之和与该分类器的阈值相比，判断该窗口是否是人脸，是的话继续进行下一个较复杂分类器的判断，否则就排除是人脸的可能，只有最终通过所有分类器的窗口才被认为是人脸。级联检测过程如图 2.6 所示。排在前面的简单分类器包含较少的 Haar-like 特征，检测准确度也较低，有很高的虚警率，但漏检率很低。后面的复杂分类器包含较多 Haar-like 特征，因此它的准确度较高，虚警率和漏检率都较低。在级联器的前端使用简单的分类器可以用很少的计算量排除大部分非人脸窗口，之后逐级提高分类器的复杂度，可以在保证较快检测速度的前提下提高人脸检测准确度。

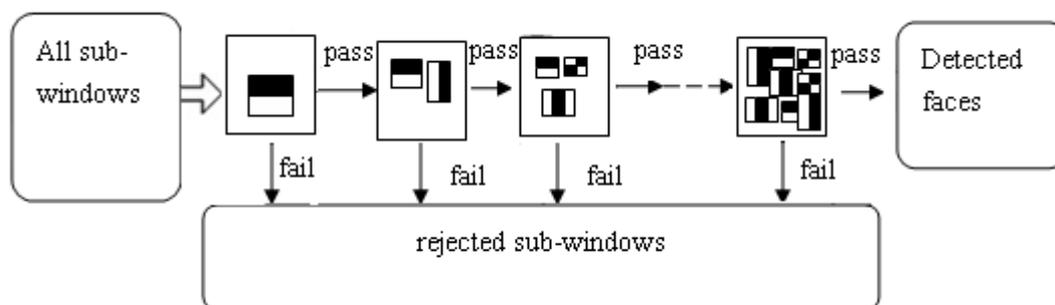


Figure2.6 Cascade structure of classifier

图 2.6 级联结构分类器

级联分类器的这种特殊处理流程其实很好利用了一个基本事实，那就是在一幅图像中，绝大多数的待检测窗口都是负样本，而级联分类器总是试图尽可能地在前面的几级分类器中就拒绝这些负样本。然而对于一个正样本而言，由于级联分类器有非常高的检测率，正样本总是能很大概率通过所有分类器。从整个检测过程来看，正样本总是一个小概率事件。

根据级联结构的特点，我们可以定义一个包含 k 个分类器的级联分类器的最终虚警率是：

$$F = \prod_{i=1}^k f_i \quad (2.9)$$

其中 F 为级联分类器的最终虚警率， f_i 为第 i 个分类器的虚警率。

这个级联分类器的最终检测率是：

$$D = \prod_{i=1}^k d_i \quad (2.10)$$

其中 D 表示级联分类器的最终检测率， d_i 为第 i 个分类器的检测率。如果要训练一个给定检测率 D 和误识率 F 的级联分类器，只需要分别训练 n 级具有检测率 d_i 和 f_i 误识率的分类器。例如，如果希望级联分类器的检测率达到 90%，则可以构造 10 级每级具有检测率 99% 的分类器 ($0.9 \approx 0.99^{10}$)。同样如果希望误识率 0.0006%，虽然看起来很难实现，但是只要每级分类器具有 30% 的误识率就可以实现 ($6 \times 10^{-6} \approx 0.3^{10}$)。所以，构造级联分类器关键是如何使每级分类器的检测率提高和使误识率降低。

通常，经过 AdaBoost 算法训练得到的分类器会保证最小化的错误率，而不是很高的检测率。因为高检测率的代价往往是高误识率，两者是相悖的。例如要提高一个分类器的检测率，一个简单而有效的方法就是降低分类器的阈值；而要降低分类器的误识率，一个简单而有效的方法是提高分类器的阈值。当然如果增加 Haar-like 特征个数，可以在提高分类器检测率的同时降低误识率，但 Haar-like

特征个数增加，又会引起计算时间的增加。所以，构造级联分类器时要同时考虑各个方面，尽力做到平衡。

2.2 AdaBoost 训练算法

AdaBoost 的全称为 Adaptive Boosting，即自适应的 Boosting 算法，是 1995 年 Freund 和 Schapire 提出的一种通过调整权值而运作的机器学习算法，解决了早期 Boosting 算法很多实践上的困难，是 Boosting 家族最具有代表性的算法。

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。随机猜测一个是或者否的问题，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。

原先的 Boosting 算法需要预先知道弱假设的错误率下限，而 AdaBoost 算法根据弱学习的反馈适应性来调整弱假设的错误率。AdaBoost 算法的核心思想是每一次迭代过程在当前的概率分布上找到一个具有最小错误率的弱分类器，然后调整概率分布，增大当前弱分类器分类错误的样本的概率值，降低当前弱分类器分类正确的样本的概率值，以突出分类错误的样本，使下一次迭代更加针对本次的不正确分类，即针对更“难”的样本，使得那些被错分的样本得到进一步重视。最终选取最具有分类意义的若干个弱分类器根据权值合成一个强分类器。Freund 和 Schapire 证明了：随着弱分类器个数的增加，得到的强分类器在训练样本上的分类错误按指数递减。

AdaBoost 训练算法的形象化描述如下：

- (1) 输入 m 个训练样本 $((x_1, y_1), (x_2, y_2), \dots, (x_m, y_m))$, $x_i \in X, y_i \in Y = \{1, -1\}$ 。这里 1 为正样本，-1 为负样本，样本总数为 m 。
- (2) 初始化样本权重：对于 $y_i = -1$ 的样本， $w_i^1 = \frac{1}{2u}$ ，对于 $y_i = 1$ 的样本， $w_i^1 = \frac{1}{2v}$ ，其中， u 和 v 分别为负样本和正样本的个数。 w_i^t 为第 t 次循环中第 i 个样本的误差权重。
- (3) 对于 $t=1 \dots T$ (T 为迭代次数) 规范化权重分布： $w_i^t \leftarrow w_i^t / \sum_{i=1}^m w_i^t$ 对于每个特征 j ，用权重分布 w_i^t 训练弱分类器 h_j ，计算 h_j 的误分类损失，选择误差 $\varepsilon_j = \sum w_i^t$ 最小弱分类器更新样本权值： $w_{t+1,i} \leftarrow w_{t,i} \beta_i^{1-e_i}$ (若 h_j 对第 i 个样本判断正确，则 $e_i = 0$ ，反之，则 $e_i = 1, \beta_i = \frac{\varepsilon_i}{1-\varepsilon_i}$)。

经过 T 次迭代，选出 T 个弱分类器，形成最终的强分类器 $H(x)$ ，在人脸检测过程中 $H(x) = \begin{cases} 1 & \sum_{t=1}^T a_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T a_t \\ 0 & \text{otherwise} \end{cases}$ ，其中 $a_t = \log \frac{1}{\beta_t}$ 。以上算法假设每一个分类器都实际有用，在每一次分类的结果中，正确分类的样本个数始终大于错误分

类的样本个数，即 $\varepsilon_i < 0.5$ 。而因为 $\beta_i = \frac{\varepsilon_i}{1 - \varepsilon_i}$ ，所以 $\beta_i < 1$ ，当在本次迭代中找出的弱分类器 h_j 将本样本 x_i 分类错误时，该样本在下一次迭代的权值 $w_{t+1,i} \leftarrow w_{t,i} \beta_i^{1 - e_i}$ 不变；如果 h_j 将本样本 x_i 分类正确，则该样本在下一次迭代的权值 $w_{t+1,i}$ 将减小，这等于减小了已分类正确样本的重要性，使下一个分类器更加重视在前面的迭代中分类错误的样本，满足了提升的思想。

从以上分析可以看出，AdaBoost具有速度快、运算简单、易于编程等优点，不需要弱学习器的先验知识，可以和任意方法灵活地结合去寻找弱假设。当给定足够的训练样本和一个能够提供中等精度的弱学习器后，它就能够提供学习的理论保证。

AdaBoost 的诞生颠覆了传统的学习系统设计思想，人们不再是试图设计一个能在整个样本空间上都精确的学习算法，而是集中于寻找仅比随机猜测略好的弱学习算法，这就容易办到的多。图 2.7 是 AdaBoost 训练算法的流程图。

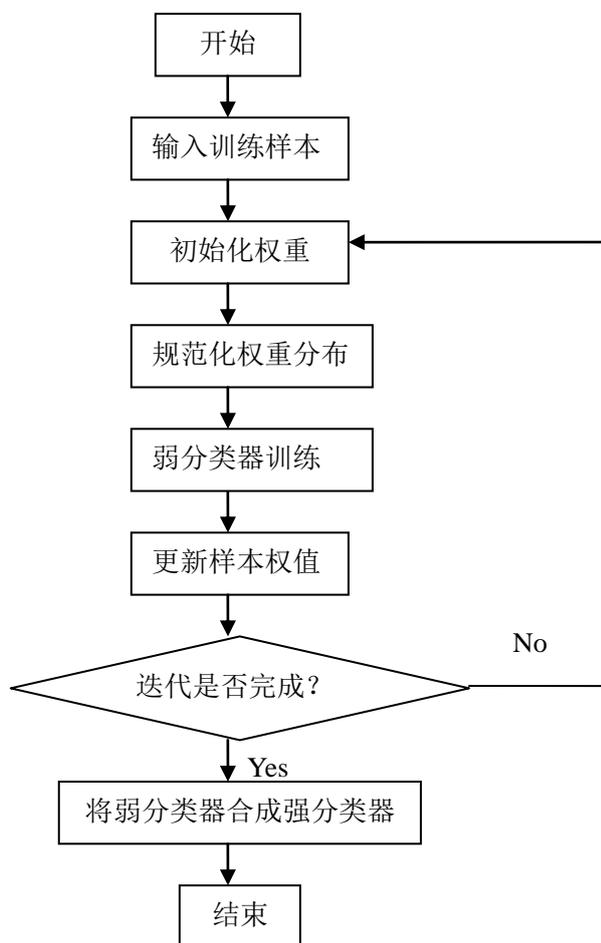


Figure2.7 The flowchart of AdaBoost train algorithm

图 2.7 AdaBoost 训练算法基本流程图

2001 年 Paul Viola 和 Michael Jones 首先将 AdaBoost 算法应用于人脸检测问

题，用简单的矩形特征，即 Haar-like 特征作为人脸特征。在 AdaBoost 人脸检测的训练机制中，就将弱分类器和特征挂起勾来，一一对应，即让每一个 Haar-like 特征相当于一个弱分类器。而 Haar-like 特征的组合，即之前所提到的级联分类器的每一级分类器就是强分类器，每一级强分类器都是用 AdaBoost 方法训练得到的。

训练之前必须先确定整个级联分类器要达到的最高虚警率 F_{\max} 、最低检测率 D_{\min} 、每个强分类器的最高虚警率 f_{\max} 、最低检测率 d_{\min} 。在确定目标的前提下，整个级联分类器至少需要由 $(n = \log f_{\max} F_{\max})$ 个强分类器组成。

整个人脸训练过程可以描述为：

- (1) 确定系统目标虚警率 F_{\max} ，各强分类器的最高虚警率 f_{\max} 、最低检测率 d_{\min} ，推断系统至少需要 $(n = \log f_{\max} F_{\max})$ 个强分类器组成；
- (2) 根据训练系统提供的人脸样本和非人脸样本库，假设人脸样本 p 个，非人脸样本需要 q 个；
- (3) 获取初始人脸样本集与非人脸样本集；
- (4) 假设 t 为强分类器序号， $t=1:n$
 - a) 训练一个强分类器 $H_t(x)$ ；
 - b) 组合前 t 个强分类器 H_1, H_2, \dots, H_t ，对人脸样本集进行验证，淘汰被错误判断的人脸样本(漏判)，并修改人脸样本的数量 p 的值；
 - c) 组合前 t 个强分类器 H_1, H_2, \dots, H_t ，对当前非人脸样本集进行验证，淘汰被正确判断的非人脸样本，并重新获取非人脸样本，以补充非人脸样本集。使其数量重新达到 q 个。在重新获取非人脸样本过程中，也是组合前 t 个强分类器对候选非人脸样本进行验证，只有被误判的候选非人脸样本才能被加入到非人脸样本集中；
 - d) 保存训练结果。

以上过程中，所谓“验证”就是利用前 t 个强分类器组成的临时级联分类器对样本进行检测。在训练过程中，要不断更新非人脸样本集。因为在验证时有一部分非人脸样本会逐渐被筛选掉，并不会通过所有层，被筛选掉的非人脸样本对于下一层的训练已经没有价值了。即用这一层之前的已经训练好的强分类器对非人脸样本进行检测，然后删除掉被正确检测的非人脸样本，留下被虚警为人脸的非人脸样本再用这些图像作为下一层强分类器的非人脸训练样本。

这种方法使得训练更具有针对性和代表性，不过在最后几层的训练中非人脸样本的收集变得很慢训练过程也变得非常困难。这是由于通过前面那些层的强分类器排除的后面的非人脸样本越来越和人脸样本相似。所以非人脸样本集是需要更新的。在验证时，也会有很小一部分人脸样本被误判为非人脸，这种情况叫做

误判。这些被误判的人脸样本会逐渐被错误地筛选掉，并不会通过所有层，它们对下一层的训练也没有价值了，可以从人脸样本集中淘汰掉。与非人脸样本集不同的是，不必补充新的人脸样本到人脸样本集中，因为最小检测率 d_{min} 通常定得很高，被漏判的人脸样本个数非常少，淘汰很小一部分影响不大。通过训练得到的级联分类器的所有参数存放在文本文件中，可在检测人脸前调入内存。

通过将 AdaBoost 算法与级联算法结合，实现的人脸检测系统成为人脸检测速度提高的一个转折点，是人脸检测从真正意义上走向实用的里程碑。AdaBoost 整个训练过程就是一个利用 AdaBoost 方法构造强分类器的过程。关键就是弱分类器——Haar-like 特征的选择训练和强分类器的组合训练。

本文中，采用了自采集的 24*24 的正面人脸图像和非人脸图像进行了 AdaBoost 算法的训练，得到了 19 个阶段(stage)共 1596 个 Haar-like 特征组成的级联分类器，表 2.2 中对每个阶段的弱分类器数做了统计。可以观察到，前几级的弱分类器较少，后面几级较多。

表 2.2 每级所对应的 Haar-like 分类器数

Table 2.2 Number of Haar-like classifiers in each stage

阶段	分类器数	阶段	分类器数	阶段	分类器数
0	1	7	28	14	107
1	2	8	42	15	88
2	3	9	64	16	88
3	9	10	37	17	301
4	14	11	53	18	579
5	23	12	57		
6	19	13	81	total	1596

2.3 基于 AdaBoost 的人脸检测

整个 AdaBoost 人脸检测算法的流程如图 2.8 所示。首先加载级联分类器，然后加载当前的图像，计算待测图像的积分图。接下来扫描其中每个 $N*N$ (N 为检测窗口长或宽) 的窗口。对所有窗口，按顺序遍历每个分类器，计算这个分类器中的每个 Haar-like 特征的特征值，并和特征阈值 α 比较，从而选择得到影响因子。每个强分类器中，所有的特征影响因子的和表示该窗口和人脸的相似度，最终把相似度与该强分类器阈值 β 比较。如果相似度大于 β ，则认为该窗口是人脸窗口，可以继续进行下一个强分类器检测。如果相似度小于 β ，则该窗口被排除，该窗口将不进行下一强分类器的检测。能通过所有强分类器的窗口就是人脸

窗口。整个过程中，每个强分类器选择由简单到复杂。首先选择人脸特征比较明显的分类特性，去掉大部分没有人脸的窗口，然后通过每一级分类器筛选，最终判断是否整幅图像含有人脸。当所有的 $N*N$ 的窗口都检测完毕后，第一轮检测就结束了。

为了适应不同大小的人脸，在完成 $N*N$ 大小窗口的检测后，需要放大窗口，并重复以上步骤，从而检测出大于 $N*N$ 的人脸窗口。这个步骤将一直被重复，直到窗口不能再被放大为止(再放大尺寸就将大于图像的长或宽)。由于 Haar-like 特征的参数是以 $N*N$ 的窗口为参照的，因此放大窗口必须调整 Haar-like 特征参数，如调整矩形顶点坐标，放大矩形尺寸和阈值等。

最后，算法需要将所有检测到的人脸窗口做归并。把位置和大小相近的人脸窗口合并为最后输出的人脸窗口。

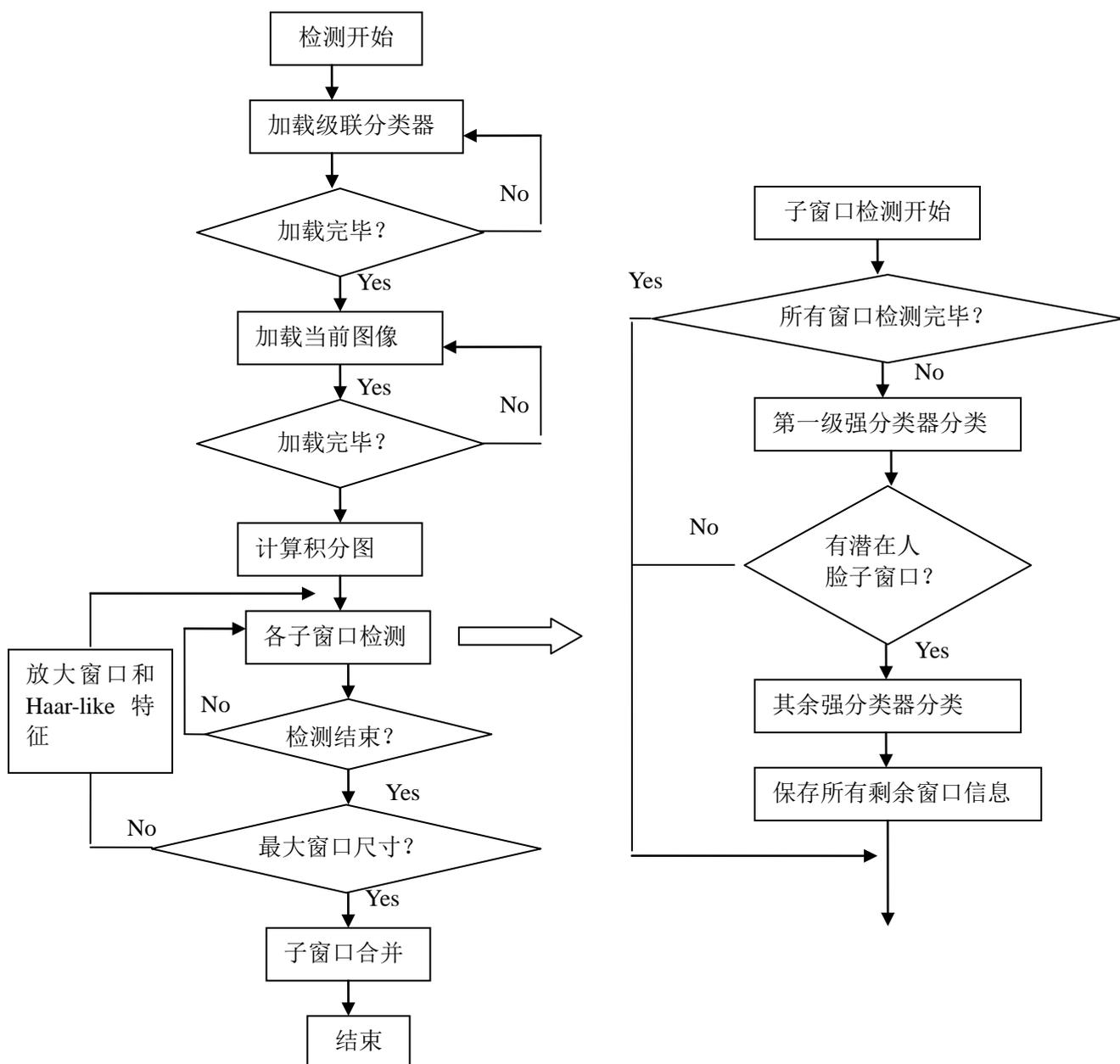


Figure 2.8 Face detection flow based on AdaBoost

图 2.8 AdaBoost 人脸检测算法检测流程

2.3.1 检测方法

如上所述，Viola 和 Jones 所提出的这种基于 AdaBoost 的人脸检测算法采用的是多尺度方法，即把待检测图像的大小保持不变，逐层等比放大检测窗口，然后再用各不同大小的窗口分别扫描整张待检测图像。窗口放大采用直接乘以一定比例系数的方法，检测时根据检测窗口的位置和所要检测窗口的大小截取要检测的子图像。

随着检测窗口的放大，弱分类器也就是 Haar-like 矩形特征也被等比例放大

以检测不同大小的人脸。对于任意大小的矩形特征,特征值的计算代价是一样的,所以这也是采用矩形特征作为弱分类器的一个缘故。这里需要指出的是,如果设置的放大系数越大,检测窗口越能快速放大到整幅图像大小,因此需要放大的次数就越少,被检测图像中被检测到的子窗口数就越少,系统的检测率就会降低,但检测速度将提高;反之亦然。

Viola 的实验表明窗口放大系数为 1.25 或 1.2 时带来的漏检较少,同时可以获得较好的检测速度。此外由于矩形特征对于位移并不是很敏感,所以没有必要对每个像素位置都进行检测,当前检测窗口的位移步长加上当前检测窗口的像素位置就是下一次检测的位置。另外需要注意的是,虽然检测窗口放大了,但是各个弱分类器的阈值是仍然保持不变的。

因为当我们将矩形特征随着检测窗口的放大而进行一定比例放大之后,理论上说,我们需要重新训练样本得到新的级联分类器,然后使用新生成的级联分类器对放大后的检测子图像进行检测。但是,其实这种重新训练是不必要的,我们只需保持各弱分类器的阈值和原来一样,那么强分类器的阈值也不会变。另外仔细分析可发现,弱分类器的阈值其实是某个被选中的矩形特征在某一样本图像上的特征值,简言之,弱分类器的阈值就是一个特征值。如果样本图像和矩形特征都放大 n 倍,这时的矩形特征值:

$$F1 = \frac{w_1}{n * n * S} * II(r_1) + \frac{w_2}{n * n * S} * II(r_2) \quad (2.11)$$

其中 w_1, w_2 是初始权值,是根据原矩形特征的权值比确定的,在训练和检测过程中保持不变。 S 为原样本图像的面积,样本放大 n 倍后其面积为原来的 n^2 倍, r_1, r_2 是放大后的矩形特征内的两个矩形,分别对应于原矩形特征内的两个矩形 r_1', r_2' , r_1, r_2 的面积分别是 r_1', r_2' 的 $n*n$ 倍。 $II(r_1), II(r_2)$ 分别是 r_1, r_2 内所有像素灰度值之和。由于 r_1, r_2 和 r_1', r_2' 在样本图像上的相对位置是一样的,对应样本的区域也是相同的,即 $II(r_1) = n*n * II(r_1'), II(r_2) = n*n * II(r_2')$ 。因此,

$$F1' = \frac{w_1}{S} * II(r_1') + \frac{w_2}{S} * II(r_2') = F1 \quad (2.12)$$

可以发现,矩形特征值不变,弱分类器的阈值也不变,因此放大样本和矩形特征不必重新训练生成新的级联分类器。前面所述的把权值除以样本图像的面积 S 也就是为了实现在放大检测窗口的同时保持分类器的阈值不变的目的。

2.3.2 检测流程

多级级联分类器前面几级通常仅有少数几个弱分类器构成,但是由于人脸图像绝大多数是背景,检测的早期阶段通常也能达到接近 99% 的检测率。然后用后面包含较多弱分类器的强分类器,对剩余的只占人脸图像少部分面积的潜在人脸窗口进行进一步的精确确认,降低虚警率。整个图像的检测过程如下:

- (1) 设级联分类器的层数为 n , 首先使用第一级强分类器进行检测; 然后通过的窗口再采用其他强分类器依次检测。
- (2) 设 i 级强分类器所包含的弱分类器个数为 m , 利用积分图快速计算该强分类器所包含的所有弱分类器的特征值, 然后权值相乘加得出 $\text{sum}(i)$ 。比较 $\text{sum}(i)$ 和第 i 级强分类器的阈值, 大于则是通过, 否则, 退出。
- (3) 通过所有强分类器的子窗口被判断为人脸; 否则, 不是人脸。

从检测过程可知, 整个检测过程中主要需要计算的只有特征值。而矩形特征值可以用积分图快速计算。当图像较大时, 放大检测窗口也只需要线性改变窗口和矩形特征的尺寸。所以, 虽然训练过程比较复杂而且费时, 但得到的级联分类器却是简单而有效的, 能取得比较快的检测速度。

2.3.3 区域合并

采用按一定比例放大检测窗口后, 通常会在同一人脸的周围检测出多个位置和大小都差别不大的矩形框。所以最后还需要一个合并过程将所检测出来的相关矩形进行合并, 以便得到唯一的人脸图像位置和人脸图像尺度。合并既相当于是对原始检测框的一个平均, 可以消除噪声使得检测结果更加精确; 同时还有消除误检的作用。合并通常要遵从两个原则[36], 一是被合并的矩形框必须要有足够大的重叠面积; 二是被合并的矩形框大小应该比较相近。在进行归并处理的时候, 上面这种重叠包含关系比较容易判断和处理, 但实验中发现两个挨得很近的人脸也可能出现交错关系, 应选择好的归并法避免将这种情况误判。

区域合并通常分两种: 某一尺度下检测到的人脸图像区域的合并和不同尺度下检测到的人脸图像区域的合并。这两种合并方法都采用求平均的方法, 即将所有检测到的人脸图像区域的坐标位置的平均值作为最后合并得到的人脸图像的坐标位置, 所有检测到的人脸图像区域的尺度的平均值作为最后合并得到的人脸图像的尺度。

某一尺度下检测到的人脸图像区域的合并是指, 当在某一尺度下进行人脸图像扫描检测时, 在某一位置附近可能会检测到多个同一人脸的图像, 因此, 就必须将这些人脸图像进行合并。因为同一尺度下的检测结果对应的窗口大小相同, 只需进行位置的合并。定义邻域半径 R 和阈值 T , 若对于某个窗口中心位置

(X_0, Y_0) , 在其半径为 R 的邻域 $(X - X_0)^2 + (Y - Y_0)^2 \leq R^2$ 内人脸窗口中心的个数为 M , 当 $M < T$ 时, 认为该窗口对应的是误检的人脸, 否则, 若 $M \geq T$, 则认为在 (X_0, Y_0) 的 R 邻域中检测到了一个人脸, 对该位置的所有人脸用求平均的方法进行合并得到一个人脸。

不同尺度下检测到的人脸图像区域的合并是指, 将各个尺度下人脸的大小和位置恢复到原始尺度的图像, 若两个人脸的重叠面积大于某一设定的阈值, 则认为这两个人脸代表同一个人脸, 对它们进行合并, 并将它们位置和大小的平均值作为合并后得到的人脸位置和大小; 如果两个人脸的重叠面积小于某一设定的阈值, 就认为它们代表两个不同的人脸区域。

在实际操作时, 通常两种合并方式是一起执行的。即在检测得到所有尺度的人脸图像之后, 根据窗口的位置和尺度, 找到所有可归属同一个人脸的窗口, 然后对这些窗口进行位置和尺度的均值合并。

2.4 本章小结

本章对目前最为高效的基于 AdaBoost 训练的人脸检测算法进行了全面的介绍, 对人脸检测过程做了详细分析。

第 3 章 人脸检测算法改进及定点化分析

本章在前述的基于 AdaBoost 人脸检测算法的基础上，将提出要实现高性能人脸检测芯片可在算法方面进行改进的几个方面。从改进算法的提出，到原理的分析，以及效果的评测几个方面进行分析阐述。为了评测的需要，本章将同时对采用的人脸数据库以及检测算法性能评测标准进行介绍。

3.1 人脸数据库和检测算法性能评测

3.1.1 人脸数据库介绍

本文主要研究高性能的人脸检测芯片的实现，具体以正面人脸检测为着手点进行。文中的正面人脸包括无任何旋转的正向以及可向左右至多各旋转 45 度的侧面人脸。因此选择了如下几个国际公认的人脸数据库作为进行检测性能评估的标准。

CMU+MIT 数据库[37]：是用来评估人脸检测算法性能，应用最为普遍的数据库之一。主要用来评估对正脸图像的检测效果，由卡耐基梅隆大学(CMU)和美国麻省理工学院(MIT)共同采集，最初是服务于 CMU 的基于神经网络的人脸检测项目。其共有四个子数据库，包含除一个专门采集旋转人脸的子数据库外，其余三个子数据库中包含 130 幅图片 507 张人脸。其中常用的 Test 子数据库有 42 幅图片，NewTest 有 65 幅图片。数据库中均为灰度图片，包括不含人脸的各类图片、绘制的人脸图片和一般照片拍摄的人脸图，背景都较复杂，个别图像质量很差。

图 3.1, 3.2, 3.3 分别是 CMU+MIT 数据库中典型的典型人脸图像，其中包括一些绘制的特殊人脸图片和非人脸图片。该数据库可用于测试算法在较差的图像质量、较复杂的背景和光线变化下对不同表情人脸进行检测的准确性。



Figure3.1 Typical images including faces in CMU+MIT database

图 3.1 CMU+MIT 中典型的包含人脸的图像



Figure3.2 Fake face images in CMU+MIT database

图 3.2 CMU+MIT 中绘制的假人脸图片



Figure3.3 Non-face images in CMU+MIT database

图 3.3 CMU+MIT 中不包含人脸的干扰图片

Bao 数据库[38]: 是一个大型人脸数据库，人脸多采自亚洲人种，对亚洲人脸的检测具有较大的参考意义。其共有 370 幅均包含人脸的图片，其中 221 幅是多人脸图片，149 幅单人脸图片，均为彩色图像。主要测试算法对复杂背景下不同表情，光线的稳定性。



Figure3.4 Typical images in Bao database

图 3.4 Bao 数据库中的典型的单人脸和多人脸图像

BioID 人脸数据库[39]: 该数据库曾服务于欧洲人脸和姿势识别工作组的 FGnet 项目。是第一个公开能真正将人脸检测结果以科学方式计算其准确度的数据库, 提供给所有致力于人脸检测领域的研究者评估其算法质量的标准。数据库采集了在自然条件下的 1521 幅分辨率为 384×286 像素的灰度正面人脸图片, 每幅图像包含一张人脸, 具有各种不同的光线变化和复杂的背景。共包含 23 个不同测试者的人脸, 同一人脸被采集多次, 每次具有不同或相似的背景, 以及不同的表情。为便于比较, 数据库中的所有图片都配有手工定位的精确人眼位置数据。与前两个数据相比, BioID 数据库中的背景较为简单, 人脸大小相对比较接近。主要测试算法对不同表情、背景的稳定性的。



Figure3.5 Typical images in BioID database

图 3.5 BioID 人脸数据库中典型人脸图像

Yale 数据库[40]: 由耶鲁(Yale)大学采集, 共 165 幅 GIF 格式灰度图组成, 内有 15 个测试者的头像, 每个人有 11 张不同表情和拍摄条件下的人脸。这 11 个条件分别为中间光、戴眼镜、表情高兴、左边光、不戴眼镜、正常情况、右边光、表情悲伤、犯困闭眼、表情惊讶、眨眼。此数据库彻底排除了背景, 主要测试人脸检测算法对不同表情和光线的兼容能力。

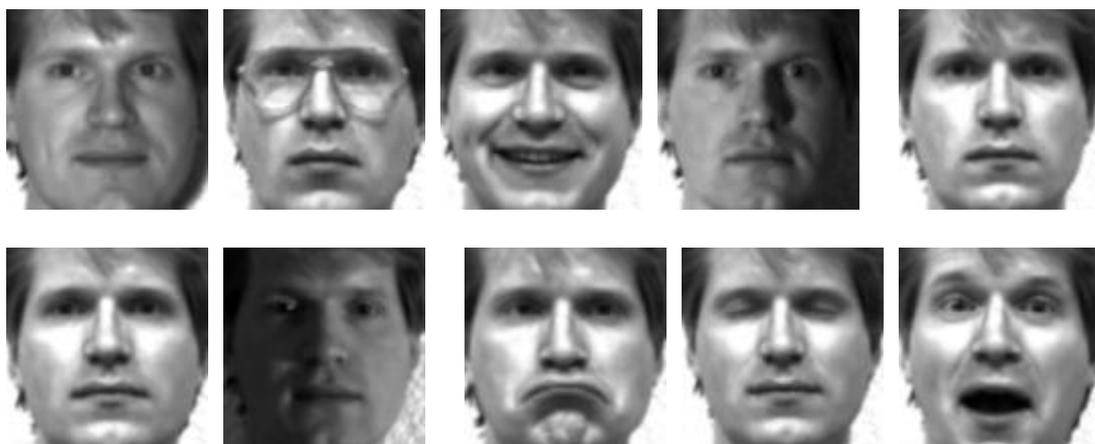


Figure3.6 Face images of different facial expression or configuration in Yale database

图 3.6 Yale 人脸数据库的不同表情与条件下的人脸

ESSEX 数据库[41]: 用于计算机视觉科学研究项目。所有的图像分在四个目录中 faces94, faces95, faces96 和 grimace, 检测难度逐步加强, faces96 和 grimace 的难度最高。其难度主要体现在两个方面, 一是背景和尺度的变化, 还有就是伴随着表情的极端变化。数据库采集 394 个人的人脸, 每个人共 20 幅不同表情与背景尺度变化下的图片, 所以一共 7880 幅图片。394 个人中有男有女, 有不同种族的人, 主要采自一年级的的大学生, 因此大部分人年龄在 18-20 岁之间, 也有一些年龄稍大。图像是 24 位彩色 JPEG 格式, 使用 S-VHS 可携式摄像机拍摄, 拍摄的光线均为人工合成的钨荧光。主要测试算法对不同表情, 装饰, 背景的稳定性的。



Figure3.7 Fce images in ESSCX database

图 3.7 ESSCX 数据库中的人脸

3.1.2 检测算法性能评测

人脸检测的算法现在尚无统一的性能评测标准, 一般评价标准主要看以下几个指标: 正确检出率、漏检率、错误报警率、ROC 曲线、检测速度及鲁棒性等。

(1) 正确检出率(detection rate)也称人脸检测率是正确检测到的人脸数与实际存在的人脸总数之比, 检测率越高, 说明检测系统对人脸的接受能力越强; 反之, 检测率越低, 接受能力越弱。可用公式表示为:

$$DR = \frac{\text{correct detected faces}}{\text{total faces}} \quad (3.1)$$

(2) 漏检率(False rejected rate:FRR)为漏检的人脸数与实际包含的人脸数之比。

$$FRR = \frac{\text{error rejected faces}}{\text{total faces}} \quad (3.2)$$

- (3) 错误报警率, 也称虚警率, 是指检测时将非人脸的区域错误地输出为人脸, 因此错误报警率也称错误检测率(False Accepted Rate: FAR), 是错误报警数与实际包含的人脸数之比, 或错误报警数与总检测窗口数之比, 本文中指前者。由于正确检出率无法反映系统对非人脸的排除能力, 有可能出现这种情况: 所有人脸都被检测到, 同时很多非人脸区域也被误认为是人脸。因此, 有必要引入虚警率来衡量系统对非人脸样本的排除能力; 虚警率越低, 说明检测系统对非人脸的排除能力越强。实践表明虚警率随着检测率的提高而提高, 检测率随着虚警率的降低而降低。

$$FAR = \frac{\text{error accepted faces}}{\text{total faces}} \quad (3.3)$$

- (4) ROC 曲线(Receiver Operation Characteristic Curve)是指通过调整检测分类器的阈值, 得到不同的正确检出率和错误报警率, 以横坐标为错误报警率、纵坐标为正确检出率绘制的曲线。ROC 曲线与横轴间所围成的面积大小表示分类系统的平均性能优劣。面积越大, 表示系统的预测性能越好, 本质上反映两类样本的可分性越大或概率密度函数之间的交叠部分越少。
- (5) 检测速度(detecting speed)也是评价一个人脸检测系统的重要方面, 可以将不同的算法在同一实验平台上进行速度比较, 从而衡量算法的优劣。由于大部分应用领域需要实时的检测人脸, 所以, 在检测率和虚警率达到满意的前提下, 检测速度越快越好。
- (6) 鲁棒性(robustness)指的是在各种条件下检测系统的适应能力。比如, 基于肤色模型的方法无法检测灰度图像; 大部分检测系统无法检测任意角度旋转的人脸。一般都要把旋转角度限制在一定范围之内; 有些检测方法受复杂背景的干扰, 只有在背景简单时效果好等。

本章中将用以上指标对人脸检测算法及其改进算法做分析。

3.2 方差预处理

人脸检测采用了由简单到复杂的级联检测结构, 利用一或两个弱分类器排除大部分的非人脸区域, 大大节约了整体检测时间。但是这一、两个弱分类器的特征计算仍然需要一定的时间。虽然流水线结构可以在连续检测多个特征时, 使每个特征的结果在一个时钟周期内得到, 但是对于不同的检测窗口, 填满流水线的初始化时间是不可避免的。因此, 出现了很多如将肤色、边缘轮廓等简单特征与级联检测算法相结合的方法[42-43], 用于进一步加快检测的速度。但肤色、边缘轮廓等的处理方法往往需要对整幅图像进行读取, 在减少了级联检测需要进一步

进行的检测区域的同时，却也增加了对图像的读取次数，即存储器的访问次数增多了，这对于减少功率消耗与处理时间不利。如何在减少总体计算量的同时，而又不增加读取图像的次数，是最佳人脸检测预处理方法要遵循的原则。

本文改变以往别的文献从肤色、边缘、背景等特征着手的方法，而是从图像最基本的特性入手，力图以最小的计算量对人脸图像进行有效的预处理。经过对各基本特性分析比较后发现，方差——这一反映图像统计特征的最基本的参数，能从总体上反映图像的基本信息。人脸方差具有一定的特点，其变化具有一定的区间性。图 3.8 中为大小是 20*20 像素的不同方差的人脸图像。Var 代表方差。

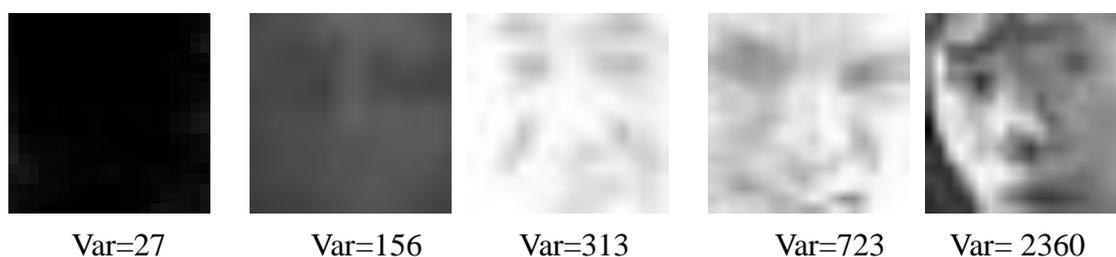


Figure 3.8 20*20 picture of different variances

图 3.8 20*20 的不同方差的人脸图像

从图 3.8 的图像中可以发现，方差大的人脸图像从视觉上比较棱角分明些。方差小于一定数值的人脸图像由于变化较小，界限模糊，基本肉眼无法分辨是否为人脸。

为了解人脸方差的确切分布规律，将 CMU+MIT、Bao 这两个数据库作为测试集，对其中 1401 个人脸方差进行了统计，统计结果如下：

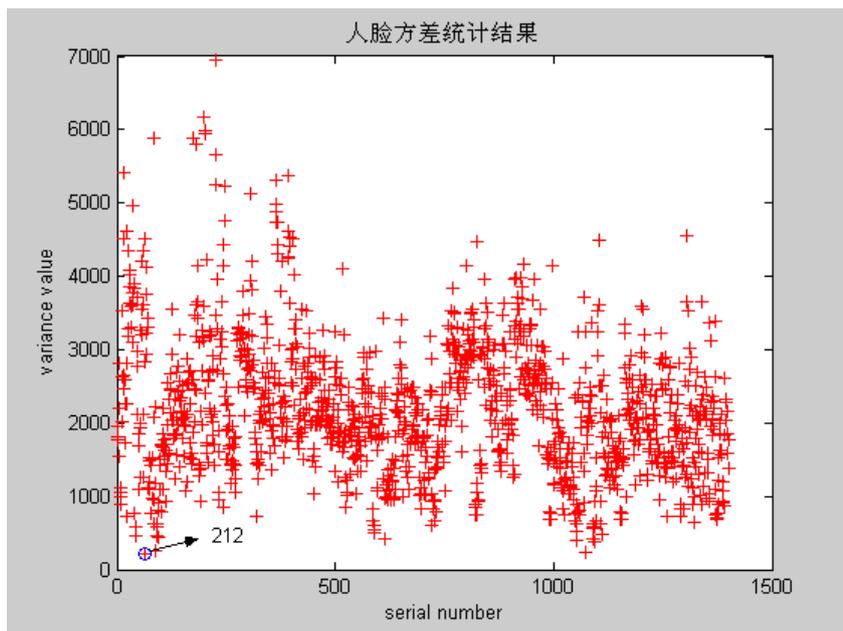


Figure3.9 Statistic result of variance value of the test faces

图 3.9 人脸方差统计结果（一）

从图 3.9 中的统计数据可以看出，方差值主要集中分布于 500-3000 之间。为进一步对数据进行直观分析，以 200 方差值为分段单位进行进一步归类统计，归类后的方差分布规律如图 3.10，从图中可以更加明显的看出人脸方差的集中分布规律。

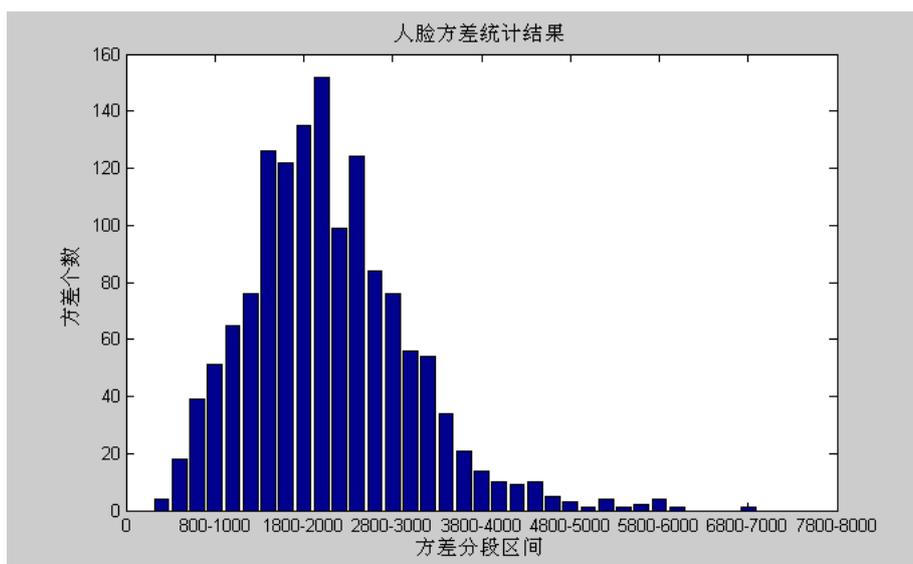


Figure 3.10 Statistic result of variance value of the test faces

图 3.10 人脸方差统计结果（二）

根据这些统计数据可以得出，采用方差预处理的方法就可以直接排除方差值不在合理人脸方差分布区域内的大部分背景，这比采用首级 Haar-like 特征计算更加快速。而且正如前所述，因为方差是调整人脸区域亮度变化必然需要的参数，对其的计算是不可避免的，可通过平方积分图快速取得，因此该预处理几乎没有消耗额外的计算时间，却可以排除一部分背景区域的特征计算，对减少计算量，缩短检测时间极为有利。特别对于背景结构单一或者背景很复杂的图片来说效果更为显著，甚至避免了前几级计算较为简单的分类器分类。

增加了方差预处理后每个检测区域的级联检测过程如图 3.11。Var 代表方差，TH 代表是否能判定为人脸的方差门限阈值。

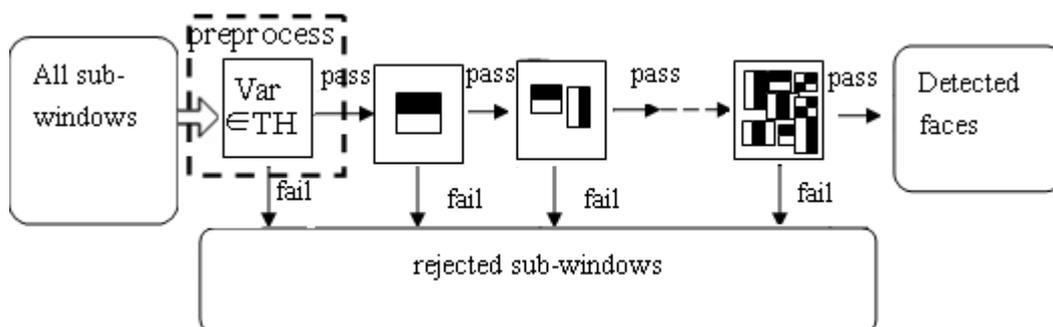


Figure 3.11 Cascade structure of classifier with preprocessing

图 3.11 增加预处理后的级联结构分类器

这里值得注意的一点是方差的门限阈值设定。如果预处理时判定为人脸的方差范围越小，显然可以使检测速度越快，排除越多的非人脸区域，但这也会同时带来将部分人脸区域排除在外的风险，引起正确检出率的下降和漏检率的上升。反之，如果方差范围选的较大，虽然保证了正确检出率，但是预处理对检测速度的提高影响就会较小。综合考虑两者，以及图 3.9 和 3.10 中所反映的人脸方差实际分布规律，选择方差的上下门限分别为 $TH_H=4000$ ， $TH_L=200$ 。将改进后人脸检测的各性能指标与原先各指标用不同的数据库数据分别进行了比较，对 CMU+MIT, Bao, BioID 数据库的比较结果如表 3.1, 3.2, 3.3 所示。

表 3.1 对 CMU+MIT 数据库进行测试的比较结果

Table 3.1 Test result of CMU+MIT database

	with preprocess	without preprocess
Detection Rate	86.7%	89%
False Reject Rate	13.3%	11%
False Accept Rate	6.9%	9.2%

表 3.2 对 Bao 数据库进行测试的比较结果

Table 3.2 Test result of Bao database

	with preprocess	without preprocess
Detection Rate	94.6%	94.6%
False Reject Rate	5.4%	5.4%
False Accept Rate	10.5%	18.3%

表 3.3 对 BioID 数据库进行测试的比较结果

Table 3.3 Test result of BioID database

	with preprocess	without preprocess
Detection Rate	93.8%	95.1%
False Reject Rate	6.2%	4.9%
False Accept Rate	2.0%	14.5%

CMU+MIT, Bao, BioID 三个数据库都属于有较复杂背景和光线变化的人脸图像集合, 人脸所占的图片面积也大小不一, 方差预处理对其的影响大致相同。对这一类型的数据库中人脸图像, 方差预处理可以使需要进行级联检测的窗口减少大约 10%-55%。背景较为单一且所占面积大的图片排除的背景窗口较多, 能达到 55%左右, 这对于检测速度的提高是很有利的, 而背景所占面积较小或其方差值在人脸方差区间内的图像, 预处理所能排除的背景窗口就较少, 最少只能排除 10%左右。除检测速度以外, 从三个表中也可以看到, 200 到 4000 的方差预处理范围可以进一步减小错误报警率, 但同时也会略微引起检测率的下降和漏检率的上升。

Yale 数据库中所有图片都为单人大头照，无背景且背景空间极少，为进一步证实方差预处理的作用，对其也进行了预处理前后检测率、虚警率、漏检率的比较。并另增加检测速度一项以观察预处理对检测速度的影响。结果如表 3.4。检测速度用级联检测所需要检测的子窗口数表示。100%表示所有的子窗口均需检测。

表 3.4 对 Yale 数据库进行测试的比较结果

Table 3.4 Test result of Yale database

	with preprocess	without preprocess
Detection Rate	95.8%	97.6%
False Reject Rate	4.2%	2.4%
False Accept Rate	0%	0%
Detection Speed	69%-99%	100%

从表中数据可以看出，200 至 4000 的方差预处理范围对 Yale 数据库中的人脸图像可以直接排除 1%-31%的子窗口，但也会引起正确检出率的下降和漏检绝率的上升。检测效果总体来说不如前三个数据库。分析原因，可以发现，这是因为该数据库中图片几乎没有背景，所以通过排除背景而提高速度的效果不明显，而且错把背景当人脸的虚警率也较小。此类数据库可适当提高方差预处理的上下门限，对于速度和检出率的提高都会有所裨益。

ESSEX 数据库是一个既包含单一背景的人脸图像也包含复杂背景的人脸图像的数据库。其测试结果(见表 3.5, 3.6)也体现了如上分析结果。

表 3.5 对 ESSEX 数据库中单一背景的人脸进行测试的比较结果

Table 3.5 Test result of faces with single background in ESSEX database

	with preprocess	without preprocess
Detection Rate	88.1%	98.4%
False Reject Rate	11.9%	1.6%
False Accept Rate	0.8%	1.3%
Detection Speed	45%-85%	100%

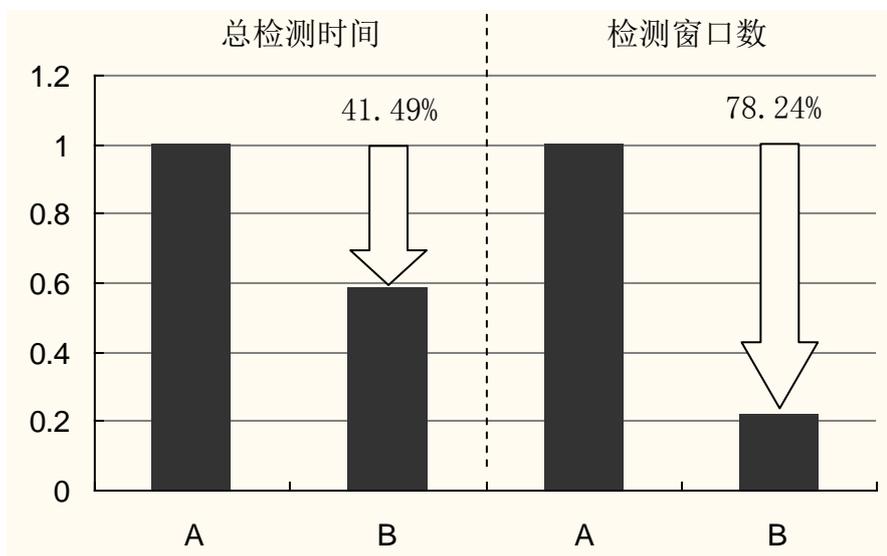
表 3.6 对 ESSEX 数据库中复杂背景的人脸进行测试的比较结果

Table 3.6 Test result of faces with complex background in ESSEX database

	with preprocess	without preprocess
Detection Rate	98.4%	99.0%
False Reject Rate	1.6%	1.0%
False Accept Rate	6.0%	8.1%
Detection Speed	61.8%-86.3%	100%

综合以上分析和几个数据库的实际测试结果，可以知道：方差预处理法确实可以在不增加计算量的前提下，快速排除背景，提高人脸检测速度，降低虚警率。尤其对于背景单一且背景所占面积较大的人脸图像，效果更为明显。且由于其依赖的是图像本身的基本特性，对各数据库中人脸的检测均有效，适应于各种情况下人脸的检测。

图 3.12 进一步给出了除检测窗口数之外，体现速度变化的总检测时间的比较。更进一步准确测出了预处理引起的检测度的变化。从图 3.12 中可以看出，方差预处理可以使需要进行级联检测的窗口平均减少 78%，总体检测时间减少 41% 左右。



A: 没有使用方差预处理的人脸检测

B: 使用方差预处理的人脸检测

Figure 3.12 face detection speed comparison between the algorithm with or without variance preprocess

图 3.12 方差预处理使用前后的人脸检测速度对比

3.3 图像缩小法人脸检测

3.3.1 图像缩小法与窗口扩大法的比较

人脸检测时，为了检测到大于原始检测窗口的人脸图像，可以采用扩大检测窗口或者缩小图像的方法来实现。

扩大检测窗口法将窗口放大一定的比例去检测较大的人脸。上一章介绍的 Viola 和 Jones 提出的基于 AdaBoost 的人脸检测算法，采用的就是扩大检测窗口的方法，即把待检测图像的大小保持不变，按一定的比例逐步放大检测窗口，然后再用各种不同大小的窗口分别扫描整张待检测图像以确定人脸的位置。因为无需对原始图像进行操作，所以该方法在软件实现上被广泛采用[44]。

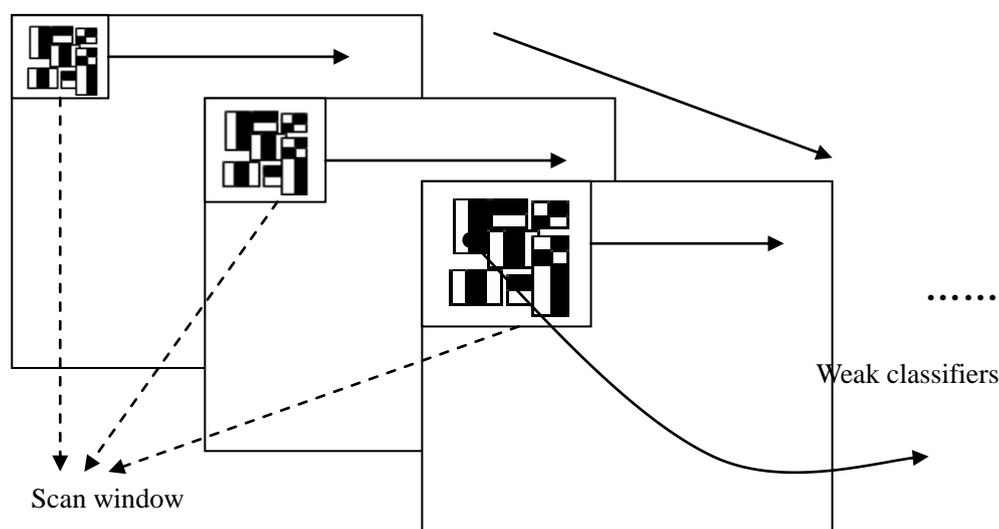


Figure 3.13 Detection faces by enlarging scan window

图 3.13 检测窗口扩大法

具体说来主要具有如下三点优势：

- (1) 容易实现。窗口扩大只是修改检测窗口的长和宽，在放大时乘以一个比例系数即可。而检测需要的弱分类器——Haar-like 特征因为是由一些简单的矩形特征组成，因此也比较容易放大，各特征的阈值也不需修改就可以实现更大人脸的检测。分类器无需重新训练，简洁方便。
- (2) 积分图可多次利用。由于改变的是特征值，原始图像并没有发生变化，因此只需要在开始检测以前计算一次积分图，就可以满足后面各个窗口计算的需要。
- (3) 不会影响检测率。因为包括 Haar-like 特征在内的各参数都是同比例线性扩

大，对检测率没有影响。

但应同时注意到，窗口扩大法也同时带来了两个问题。而且这两个问题对硬件的有效实现都是至关重要的。

- (1) 随着弱分类器特征数值的放大，存放弱分类器各值的寄存器宽度也得不断加大。在硬件电路中，由于寄存器的位数必须确定，即在设计电路时要使得寄存器具备可能出现的最大特征的位数，这往往需要占用大量的存储空间，而事实上用到这些数值较大的特征次数是很少的，这样造成了资源的浪费，但又无法改善。
- (2) 随着检测窗口的扩大，特征矩阵变大，对积分图的访问会越来越分散，这很有可能导致缓存中没有所需要的数据，或者需要很大的存储阵列来存储图像的积分图。以分辨率为 800 像素*600 像素的人脸灰度图像为例，由于积分图的单个单元的最大值为 $255 \times 800 \times 600$ ，因此需要 27bit 来存放积分图中的一个数，而整个积分图则需要 $800 \text{bit} \times 600 \text{bit} \times 27 \text{ bit}$ ，即 13 Mbit 的存储单元。用 FPGA 或者 ASIC 实现的时候，采用这么大的片内存储单元显然是不明智的。另外，如果是对于更大分辨率的图像，其积分和所需要的位数和图像尺度更大，积分图存储空间将成倍上升，极其不利于硬件实现。

图像缩小法的人脸检测采取固定扫描窗口，按一定比例逐步缩小被检测图像，然后再用扫描窗口对缩小后的人脸进行检测，以测得原先面积比扫描窗口大的人脸区域。最后在检测完成后，按缩小的比例再将测得的人脸区域面积放大，得到真实的人脸大小。此法可以解决窗口扩大法的两个问题，避免计算大幅图像的积分和。

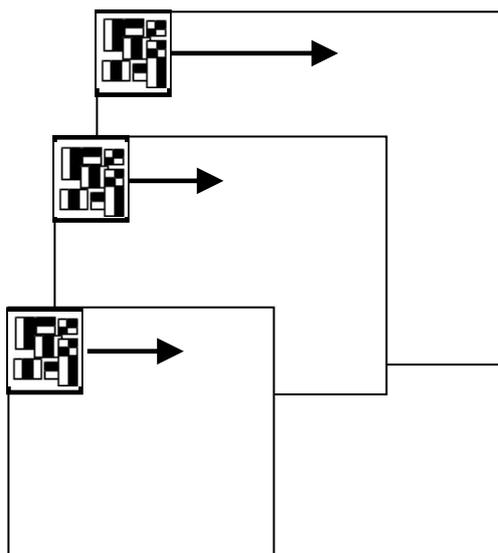


Figure 3.14 Detection faces by down scale the image

图 3.14 图像缩小法

图像缩小法能解决上述两问题的原因如下:

- (1) 由于采用的是改变图像重检测的方法, 所以所有的弱分类器是不需要做任何改动的, 检测窗口的大小也恒定不变。所以避免了弱分类器特征扩大所带来的存储空间的浪费, 始终只需要存储最小尺度检测框的弱分类器数据。这既可以节约存储空间, 也可以节省弱分类器计算的时间。
- (2) 由于图像大小的改变, 积分图在每个尺度下都需要重新计算。随着图像的变小, 积分和值会越来越小, 积分图的存储空间也会越来越少。而且, 鉴于此方法中, 积分图只使用一次, 因此无需保留整幅图像的积分图, 只需随着检测顺序, 边计算边存储当前待检测窗口的积分图即可。以使用最为广泛的 24×24 像素的检测窗口为例, 积分和最大为 $24 \times 24 \times 255$, 需要 18bits 来存储。那么整个积分图只需要 $24 \times 24 \times 18 \text{bit} = 10 \text{kbit}$, 这比存储一副图像的积分图要小得多, 而且也不随被检测图像大小的改变而变化, 有利于硬件实现。

虽然图像缩小法需要付出的代价是每检测一个 24×24 的窗口就需要重新计算一次积分图, 另外也要考虑到改变待检图像的大小所需要花费的时间。但采用流水和并行执行的方式可以克服图像缩小所带来的时间消耗。这在第 4 章中将详细介绍。也因为以上原因, 图像缩小技术已出现在现存的硬件实现中[61-62]。

3.3.2 两类主要的图像缩小法

图像缩小主要有两种算法实现: 近邻插值法和双线性插值法[45]。

在近邻插值法(nearest neighbor interpolation algorithm)中, 新图像的某像素的值取自原始图像中与该像素距离最为接近的像素的值。这种插值算法计算量较小。而双线性插值法(bilinear interpolation algorithm)中, 新图像的像素值通过原图像中与该像素距离最接近的四个像素(上, 下, 左, 右)计算得到, 它的计算量较大, 但是可以产生比近邻插值法显示效果好的缩小后的图像。

表 3.7, 3.8, 3.9, 3.10, 3.11 中显示的是分别采用近邻插值法, 双线性插值法和窗口扩大法对不同人脸图像数据库进行检测得到的这个正确检出率和虚警率。三者均采用 1.25 的缩小比例因子和 Yale 数据库, 在 vc++6.0 的环境下测得。

表 3.7 图像缩小法与窗口扩大法对 CMU+MIT 人脸数据库进行测试的比较结果

Table 3.7 Test result of CMU+MIT databases with image scaling and sub-window enlargement methods

	Detection Rate	False Accept Rate
近邻插值法	86.7%	8.3%
双线性插值法	89.9%	10.6%
窗口扩大法	89%	9.2%

表 3.8 图像缩小法与窗口扩大法对 Bao 人脸数据库进行测试的比较结果

Table 3.8 Test result of Baodatabases with image scaling and sub-window enlargement methods

	Detection Rate	False Accept Rate
近邻插值法	89.6%	20.6%
双线性插值法	93.5%	22.1%
窗口扩大法	94.6%	18.3%

表 3.9 图像缩小法与窗口扩大法对 BioID 人脸数据库进行测试的比较结果

Table 3.9 Test result of BioID databases with image scaling and sub-window enlargement methods

	Detection Rate	False Accept Rate
近邻插值法	93.2%	14.5%
双线性插值法	94.7%	16.7%
窗口扩大法	95.1%	14.5%

表 3.10 图像缩小法与窗口扩大法对 Yale 人脸数据库进行测试的比较结果

Table 3.10 Test result of Yale databases with image scaling and sub-window enlargement methods

	Detection Rate	False Accept Rate
近邻插值法	97%	0%
双线性插值法	97.6%	0%
窗口扩大法	97.6%	0%

表 3.11 图像缩小法与窗口扩大法对 ESSEX 人脸数据库进行测试的比较结果

Table 3.11 Test result of ESSEX databases with image scaling and sub-window enlargement methods

	Detection Rate	False Accept Rate
近邻插值法	98.6%	3.1%
双线性插值法	98.7%	3.7%
窗口扩大法	98.8%	4.5%

从这 5 个数据库的测试结果可以看出：近邻插值法，和双线性插值法以及窗口扩大法比较，在检测率上会有下降，但是虚警率也同时降低了。而两种图像缩小法比较而言，双线性插值法得到的检测率稍高，但是在虚警率上不如近邻插值法好。综合考虑三者，以及硬件的实现特点，本文最终采用了计算量较小，实现较为容易的近邻插值图像缩小法。而检测窗口的大小根据训练时的设定，为 24*24 像素。

3.4 高性能人脸检测算法定点化分析

人脸检测的硬件实现常常需要涉及到将浮点数转换为定点整数，以便在计算中提高速度，占用较小的硬件面积，因此人脸检测算法的定点化是非常重要的一个环节，关系到整个系统的最终性能。

定点数与浮点数相比，定点数表示的精度较低，但表示的数值范围较大；而浮点数恰恰相反。定点数的运算在计算机中实现起来比较简单，效率较高；而浮点数的运算在计算机中实现起来比较复杂，效率相对较低。分析本文所用的人脸检测算法，其中涉及大量的矩阵特征值计算。如果采用浮点计算，虽然精度高，但速度势必会低很多。相反，定点数据处理速度快，硬件功耗低，虽然定点过程会引起一定程度检测精度的下降，但为了能达到设计所需要的速度，浮点转定点是必要的一项工作。保留足够的定点数据宽度，是可以满足设计所需的检测精度的。

通常，浮点转定点有以下几种方法：

- (1) Q 格式的浮点数，比如 Q13 格式。这多用于 DSP 嵌入式系统开发中使用。其过程是将浮点数放大若干倍(2 的幂次)，在解释运算结果时再把这个放大的倍数考虑进来，得到最终结果。
- (2) 利用 Matlab 的 rat 和 rats 函数。这两个函数可以将输入的浮点数表示成最近似的有理数，有理数的精度可以通过函数中的参数来控制。将所有的浮点数

都变为一定精度的有理数后，再通过一个有理化操作就将所有的浮点数变成了定点整数。在这种方法中，一个确定的字长表达的数的范围和精度是相互制约的，要表达的数字范围越大，能达到的精度越低，反之精度越高。

本次设计结合上述两者优点，在一定的精度要求下，直接对数据位进行操作，提取指数和尾数，然后对尾数执行移位操作。例如浮点数 y 定点化为 n 位整数 x 。如果 y 的范围是可以确定的，则 n 也就确定了。对于无法确定范围的变量， n 的确定采用统计分析方法，即用足够多的输入信号样值来确定程序中变量的动态范围。这里输入信号要有一定的数量，只有这样，统计出来的结果才能具有典型性。当然，统计分析毕竟不可能涉及所有可能发生的情况，因此，统计得出的结果在程序设计时可采取一些保护措施，如适当牺牲一些精度。

整个人脸检测中，需要进行定点化的数据有：

- (1) 每个检测子窗口的均值和方差值。
- (2) 每个弱分类器的门限和权重。
- (3) 级联强分类器每一级的门限。

这些数据范围均可以确定，所以各数据定点化的位数是确定的。设定各数据精度为 8-15，即将浮点的各数放大 2^8 - 2^{15} 倍，所得的人脸检测率变化情况如图 3.15 中所示。图中数据统计自 CMU+MIT 数据库。

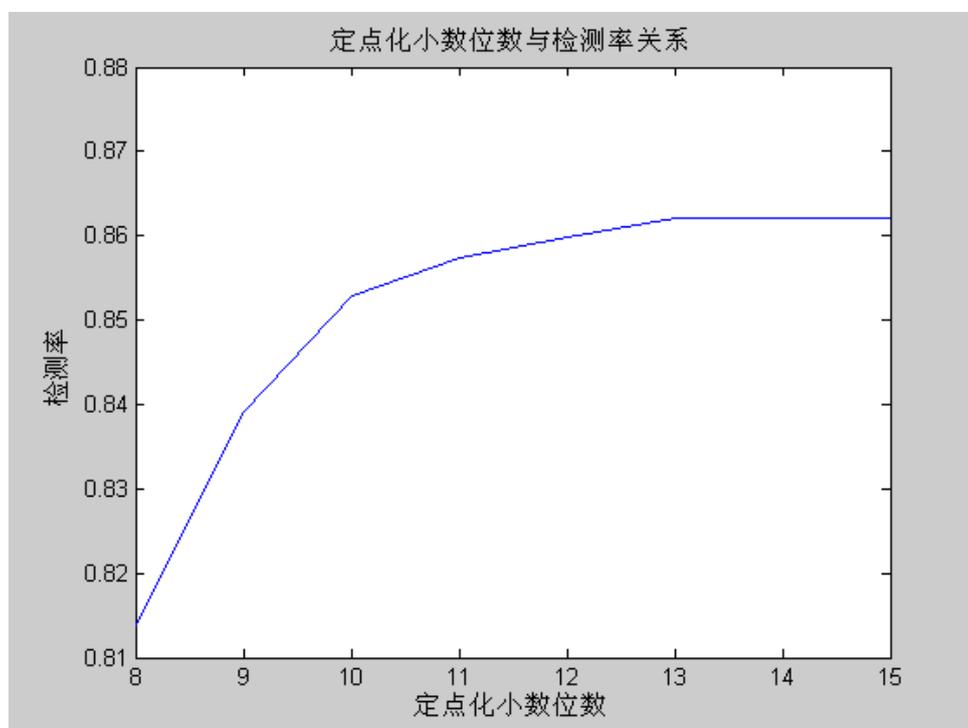


Figure 3.15 Detection rate against decimal bit width of fixed-point data

图 3.15 定点化的小数位数与检测率之间关系图

可以发现，在定点化小数位数改变的过程中，检测率呈上升趋势，保留的小数位越多，检测率越高。但当小数位数定为 14 位之后，检测率基本稳定在 86.2%(浮点时检测率为 86.7%)。因此我们最终选择定点化后各数保留小数位数为 14 位。

采用 DSP DM642 平台，对定点化之后的检测速度进行测试。测试结果发现， 352×288 规格的图像中人脸检测大约需要 70ms。而未进行定点化时，浮点数执行的速度为 1045ms，检测速度获得了大幅提升。

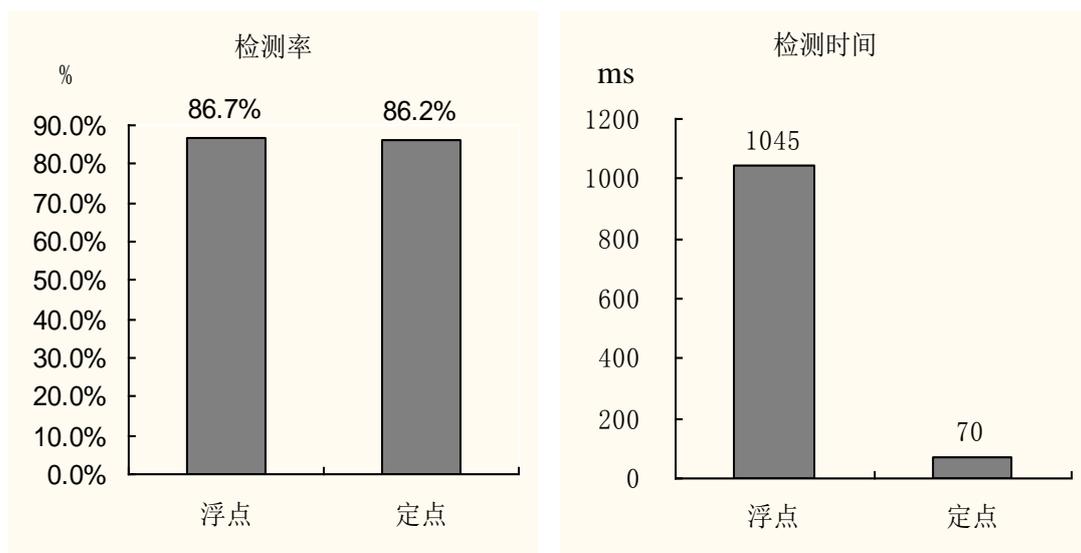


Figure 3.16 Comparison of detection accuracy and speed between fixed-point and float data

图 3.16 定点和浮点情况下检测率、检测速度比较

图 3.16 是定点化后检测率、速度与浮点相比的柱状图，由图可见，定点化是十分必要的。

由于对于灰度图，每个检测子窗口的均值和方差值整数部分可以确定为 8 位和 16 位，弱分类器门限的整数部分为 10 位，弱分类器权重的整数部分可用 2 位表示，强分类器门限的整数部分为 6 位，所以最终确定检测子窗口的均值和方差值、弱分类器的门限和权重、强分类器每一级的门限分别为 22, 30, 24, 16, 20 位。

3.5 本章小结

本章首先介绍了五类特点各异、应用较为广泛的人脸数据库，并且提出了检测算法性能评测的几个指标，便于从各个方面评估本文中人脸检测算法的有效性。并提出了用方差预处理检测法对人脸检测算法进行改进，大大减小了计算量，取得了较好的检测效果。另外对窗口扩大法和图像缩小法进行了全面分析，并对

两类图像缩小法进行了比较,最终选择了计算量较小而又比较适合硬件实现的近邻插值法作为检测大于检测框的人脸区域的方法。最后,对算法进行了定点化分析,选用了检测性能较好,定点化位宽较小的定点化方案。

第4章 高性能人脸检测芯片研究

从人脸检测在各领域的应用中可以发现,检测的准确度是必不可少的一个方面。而随着图像采集传感器的分辨率的不断提升,要得到较高准确度的检测所需处理的数据量也就更大。本章提出的高性能人脸检测 IP 核,针对大数据量的计算,结合前一章中对人脸检测算法的优化,保证了较高的准确度要求,符合当今人脸检测需要。

但对这样大的输入图像数据量,要实现芯片的高速、低功耗和小面积,难度也就更大。本章全面剖析阐述高性能的人脸检测芯片的实现,从架构的确定、并行度的分析、积分图快速更新法和四级流水线处理结构的提出、分类器的选择与存储、高效存储方案等各个方面实现人脸检测的实时、低成本、低功耗的需求。并对设计的高性能人脸检测芯片进行了不同数据库的检测准确度测试,给出了 FPGA 的验证结果以及综合仿真测试与比较结果。

4.1 人脸检测芯片硬件架构

常用的基于 AdaBoost 算法的人脸检测架构可大致分为两大类,细胞阵列(cell array)架构[54][58][59][60]和帧存储器型阵列架构 [55][56][64][46][47]。这两个架构各有特点,设计时就这两个架构进行了深入分析比较,最终选择了存储空间优先型帧存储器架构。

4.1.1 细胞阵列架构

所谓细胞阵列架构,是因为架构由许多小“细胞”(cell)组成,每个小细胞都存储一个数据和特征检测时对该数据的计算电路。由于这样的每个细胞具有较为简单的电路结构,使得整个架构可实现类似脉动阵列的工作方式,使关键路径缩短,便于提高数据处理频率,所以受到一部分研究者的欢迎。此种架构根据细胞中存储的数据类型的不同又可分为积分图细胞存储阵列架构、检测窗口细胞存储阵列架构。

(1) 积分图细胞存储阵列架构

积分图存储阵列架构的架构框图如图 4.1:

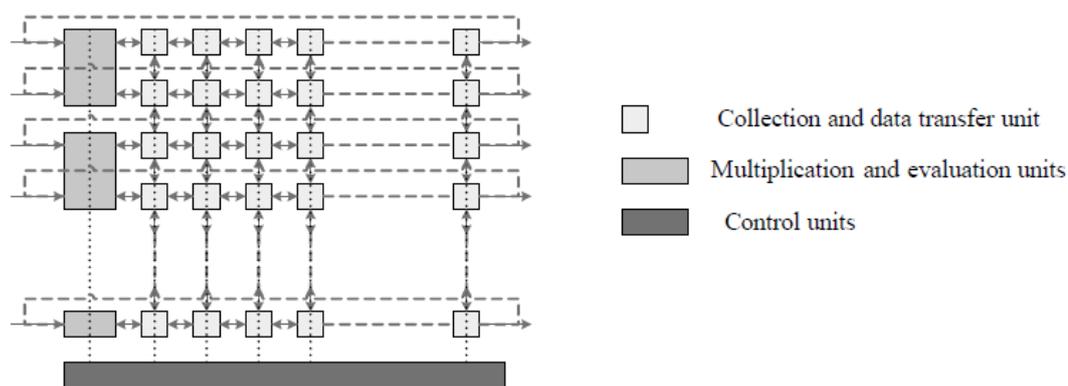


Figure 4.1 array architecture of integral image [54]

图 4.1 积分图存储阵列架构

图 4.1 中，为数众多的小方块为数据存储与传输单元，即为细胞单元，占据整个架构的绝大部分。左侧的大方块为乘法器和检测单元，计算各细胞单元中数据之间的乘法和每级强分类器的特征值估计等。下方的长条形单元为控制模块，控制数据的流动和协调运算。

整个阵列架构的工作过程如下：

首先，从阵列的左侧输入图像数据，每行每个时钟周期输入一个像素的数据。数据存储与传输单元接受数据并向右侧传递。与此同时，控制单元控制数据存储与传输单元的数据向下、向右流动，每一个数据存储与传输单元对前一级的数据进行累加，产生积分图。

然后，控制单元根据每个弱分类器特征向量的值，控制该特征向量的左下，右下和右上三个顶点的灰度图的值向左上角移动。当每一个顶点移动至左上角时，和左上角的像素值进行加减运算。当最右下角的点移动到最左上角时，特征向量值的计算过程就完成了。

依次对所有的特征向量都重复以上计算步骤。所有的特征向量的值就都存储在相应特征向量的左上角的数据存储与传输单元中了。

接下来，控制单元控制所有的数据存储与传输单元中的特征向量值向左移动至最左侧的乘法与检测单元。由乘法与检测单元计算出特征向量值乘后的值，并将结果保存起来。这就完成了一个弱分类器的检测。当一级中所有的弱分类器都计算完毕后，整个强分类器的最终值也就得到了。这时就可以根据强分类器的结果决定是否计算下一个强分类器的值。

当原图像及其所有缩小后的图像都进行完以上所有步骤的操作后，整个图像的检测也就最终完成了。

该架构通过增加访问图像数据的带宽，进而带动整个图像检测速度的上升。

其优势在于图像只需读取一次，以后每次都可以并行访问该图像灰度图的任意多个像素的值。这加速了每一个分类器中特征向量的计算。从而提高了整个检测系统的性能。文献[54]采用了这种方法，在 500MHz 主频下，系统达到 53fps 的速度。但是，该架构的一个较大的缺点是资源占用量极大。对于一个 $m*n$ 大小的图片来说，仅数据存储与传输单元就需要 $m*n$ 个。而且由于要存储图像的灰度积分图，越靠近右下角的数据存储与传输单元的位宽越大，一个以 8bit 位宽表示一个像素的 $1024*1024$ 的图片中，最右下角像素的灰度位宽将达到 28bits。此外，为了计算特征向量的值，每一个数据存储与传输单元都需要有另一个用于存储特征向量值的空间。这样整个系统的资源消耗量是极大的。文献[54]采用这种方法，在 90nm 工艺下，核心面积也达到 115mm^2 ，相当于 1000 万门的规模。成本巨大，不适合实际应用。

文献[54]的作者也意识到了这个问题，在文献[60]中做了改进，结合图像缩小法和特征扩大法，大幅度减少数据存储与传输单元。但其仍需 210 万门的晶体管，面积仍然较大。

(2) 检测窗口细胞存储阵列架构

检测窗口存储阵列架构的架构图如下：

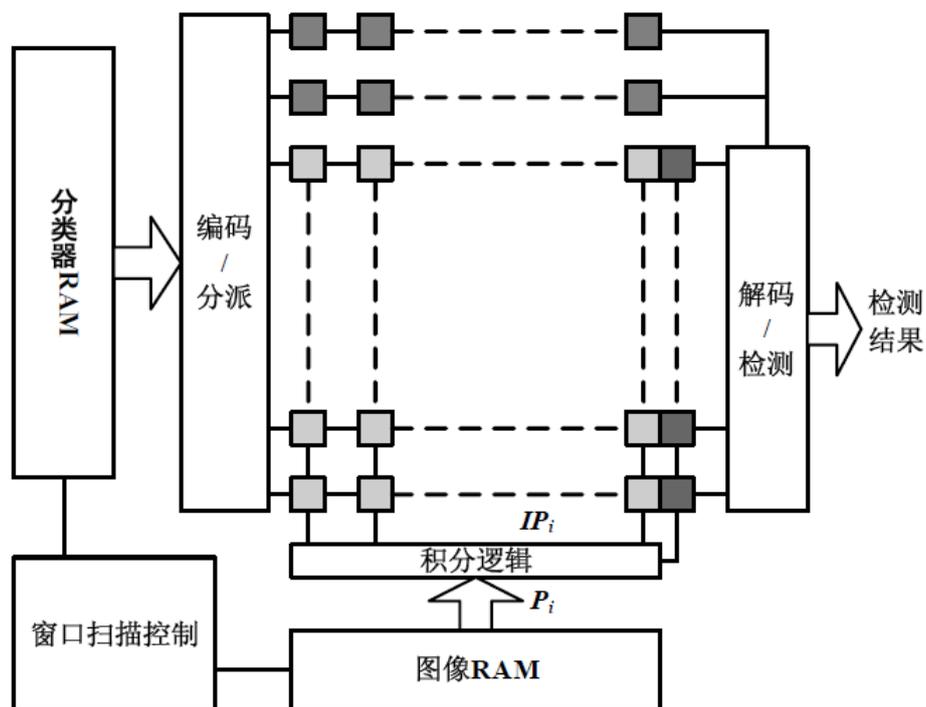


Figure 4.2 Array architecture of scan window[58-59]

图 4.2 检测窗口存储阵列架构

其中，窗口扫描控制指导整个架构的工作；图像 RAM 存储图像像素值；分

类器 RAM 存储所有的分类器数据；积分逻辑负责产生行积分；编码/分派模块负责将分类器特征向量传送至相应的处理流水线中；浅色方块单元为检测窗口积分图存储阵列；横线单元为分类器阈值与参数流水线；竖线单元为平方和积分单元，负责计算检测窗口的平方积分和；解码/检测模块负责对流水线的计算结果进行处理，判断扫描窗口是否含有人脸。

检测过程分为装载阶段和检测阶段。

在装载阶段，每个窗口的像素会一行行从图像 RAM 中被读出，在积分逻辑中进行横向积分，然后装入阵列，完成积分图的计算。阵列底部的单元累计输入的行数据，后续的单元则把数据依次往后移，这样就完成了纵向积分。当所有行都读取后，阵列中就保存了窗口的积分图。

在检测阶段，分类器 RAM 中的数据从左侧不断进入阵列。其中“编码/分派”模块将特征矩形的参数编码为有序对。然后把编码放入矩形顶点坐标对应的行的横向流水线中。理论上每个周期都可以从左侧进入一个矩形参数。阵列中的单元根据输入数据，选择处理或传输两种动作。从而计算出特征向量的特征值。之后，解码/检测模块负责对特征值进行处理，判断检测窗口是否含有人脸。

在该架构中，图像的存储单元个数被减少至检测窗口中像素点的个数。通过简化不必要的存储器，部分解决了以图像为中心的架构中存储单元占用资源过多的问题。但此优化方法有一定的代价，那就是降低了图像检测的并行度。在积分图存储阵列架构的结构中，由于整个图像的灰度积分图都已经存储在阵列单元中，分类器的特征向量可以流水地在各个扫描窗口中进行并行计算，并且该流水不会中断。而在该结构中，由于阵列存储单元只存储了一个检测窗口的灰度图。检测窗口装载完毕后，流水线中只能进入该检测窗口所需要的特征向量。这就导致每检测一个检测窗口的一个强分类器特征向量，就必须先填满整个流水线，最后还要排空流水线。这样，该结构在检测图片部分就需要比之前的结构多 $((\text{图像检测窗口个数} * \text{每个检测窗口需检测的强分类器个数的平均数} - 1) * \text{扫描窗口像素宽度})$ 个时钟周期。

4.1.2 帧存储器型阵列架构

帧存储器型阵列架构一个很明显的特点是包含一个独立的帧缓冲器(frame buffer)。帧存储器型阵列架构示意图如下：

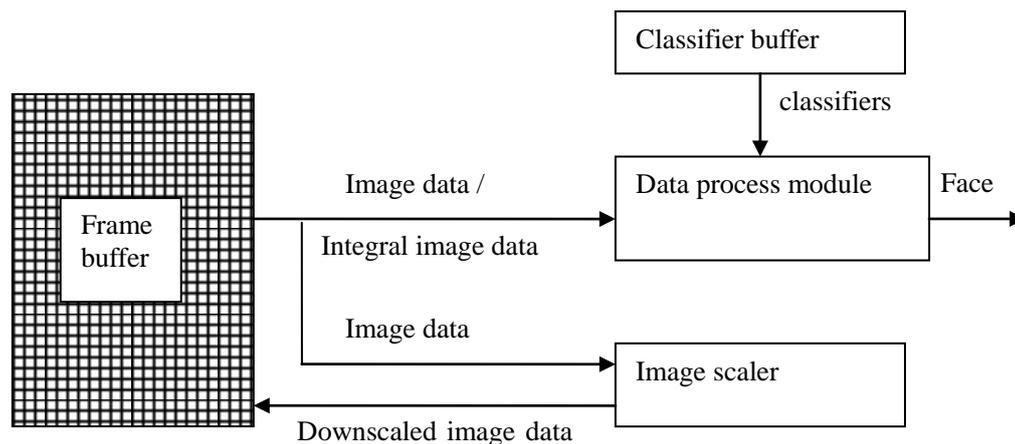


Figure 4.3 Frame buffer architecture

图 4.3 帧存储器型阵列架构

从图 4.3 中可以看出，帧存储器阵列架构由帧缓冲器(Frame buffer)提供给数据分类处理模块(Data process module)原图像的数据或者积分图数据，结合分类器缓冲器(Classifier buffer)提供的分类器数据进行人脸检测。

按照帧缓冲器中存储数据的量可以分为全存储，部分存储。按照存储的不同类型的数据，又可分为图像数据存储和积分图数据存储。因此帧存储器阵列架构根据不同的类型和数据量的组合可分为四种。帧存储器存储所有原图像数据架构、帧存储器存储部分原图像数据架构、帧存储器存储全部积分图数据架构、帧存储器存储部分积分图数据架构。

(1) 帧存储器存储所有原图像数据的架构

这种架构将传感器所得的原图数据全部存储在存储器中。分类计算前，由积分图产生器从该缓冲器中取得图像数据以计算检测窗口的积分图，然后再由分类计算模块根据积分图中数据计算每个特征的特征值进行检测。文献[56]就属于此类。该架构的优势在于整个图像的数据只需读取一次就可检测完毕。而其缺点也是明显的。由于要存储的数据众多，如果将该帧缓冲器存于片内，则会占据芯片的极大面积，造成芯片成本的上升。如果将其存于片外，则受总线宽度影响，原图数据从片外读取进来的速度比较慢，极大影响芯片的数据吞吐率。

(2) 帧存储器存储部分原图像数据的架构

此架构只存储一部分原图像的数据在片内存储器中，通过不断更新存储的数据，完成数据的处理。文献[55][64][79][48]均属于此类。

这里值得指出的是，随着部分存储数据量的不同，图像数据的读取次数也会受影响。存储的部分数据越多，一般图像数据需要重复读取的次数会越少。

(3) 帧存储器存储全部积分图数据的架构

此种架构将原图的积分图数据计算完毕后存储在帧存储器中，优点是积分图只需计算一次，无需不断更新。但由于积分图数据会随着图片大小迅速增加，该架构存储所需要的空间是四种架构中最大的。

(4) 帧存储器存储部分积分图数据架构

在这种架构中帧存储器存储一部分原图产生的积分图数据，以供检测所需。随着检测的进行，更新积分图。此架构也存在着积分图数据随图像增大迅速增加的问题。

4.1.3 架构比较与确定

为便于比较，将以上两种细胞阵列架构与四种帧存储器架构的优缺点列于表 4.1 中。表中，cell 代表细胞阵列架构，frame 代表帧存储器架构。

表 4.1 人脸检测各架构的优缺点比较

Table 4.1 The comparison between each array

	优点	缺点
积分图 cell	图像只读一次，可并行访问任意多个像素	资源占用量极大，造成硬件成本很高。
检测窗口 cell	只占用检测窗口大小的存储器资源	检测的并行度被降低，流水线不能连贯
全图 frame	图像只读一次，可随时访问	资源占用量极大，造成硬件成本很高。
部分图 frame	图像读取次数与部分存储的空间有关	存储的部分数据如选择不当，则需要多次更新，增加功耗。
全积分 frame	积分图只需计算一次，并可随时同时访问	所需的存储空间巨大，且积分值随图像增大会迅速增加
部分积分 frame	与全积分 frame 相比，减少了存储空间	积分图仍然需要更新，而且积分值随图像增大会迅速增加

为进一步了解每种架构的工作能力，我们将几种架构所需具体的存储空间、积分图空间以及计算所需的操作次数列于表 4.2 中。假设所需要处理的图像大小为 1024*1024，检测窗口大小为 24*24，总线宽度为 8 位。其中的部分存储结构

列举了两种极端，一种偏向于操作次数最少，一种偏重存储空间最少。

表 4.2 几种架构比较

Table 4.2 Comparison between several architectures

	存储空间 (bits)	积分图空间 (bits)	操作次数
积分图 cell	1024*1024*108		1024*1024
检测窗口 cell	24*24*46+24*44+21*24		1024*1024*(1024-24+1) + (1024-24)*24*(1024-24 +1)
全图 frame	1024*1024*8	24*24*18	1024*1024
部分图 frame (操作优 先)	1024*24*8	24*24*18	1024*1024
部分图 frame (存储空 间优先)	24*24*8	24*24*18	1024*1024*24
全积分 frame	1024*1024*28	0	1024*1024
部分积分 frame (操作 优先)	1024*24*28	0	1024*1024
部分积分 frame (存储 空间优先)	24*24*28	0	1024*1024*24

通过以上优缺点列举和具体指标的比较发现，采用帧存储器存储部分原图像数据的架构在存储空间和操作次数的综合性能上占相对优势。在本文中，将此种结构加以改进，其中的帧缓冲器采用了双口 SRAM，使数据写入和读取能同步进行，便于在数据处理的同时，还可以从片外取数据。这样虽然也受总线宽度的限制，但是由于数据处理时间一般大于从片外读取图像数据的时间，所以有效利用了数据读取与写入的并行，减少了整体检测时间。另外再结合合理的控制电路，使这种部分存储原图数据的结构也达到使原图像数据只需读取一次的效果。帧缓冲器的具体设计在下节中描述。最终确定的架构框图如图 4.4 所示。

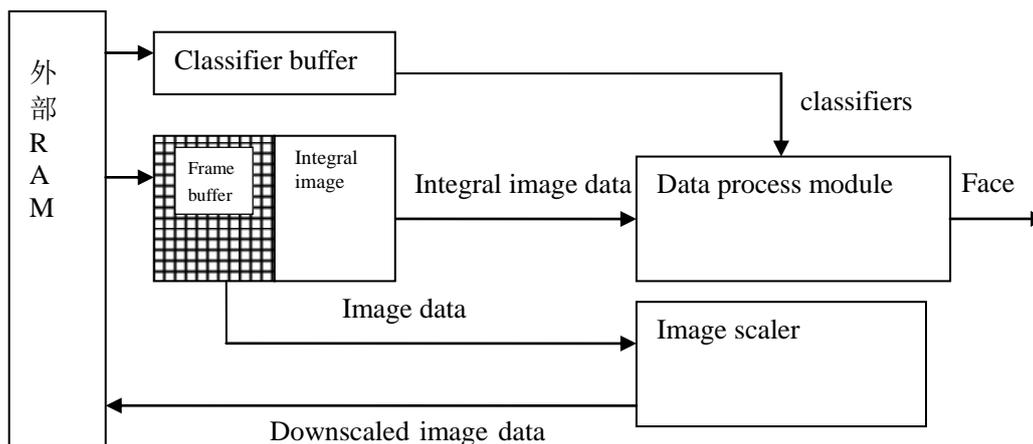


Figure 4.4 Top architecture of face detection

图 4.4 人脸检测芯片硬件架构

为加快数据传输速度，本文采用的实际总线宽度为 32 位，这样每次可以传输 4 个像素的值，大大提高了传输速度。

4.2 高效存储方案

在本文采用的帧存储器阵列架构中，帧存储器起着图像数据存储和传输的重要作用，其结构直接影响整个系统的检测时间和功耗等。

4.2.1 存储器结构与效能比较

本文研究的帧缓冲器存储部分的图像数据，其内部结构如图 4.5 所示。

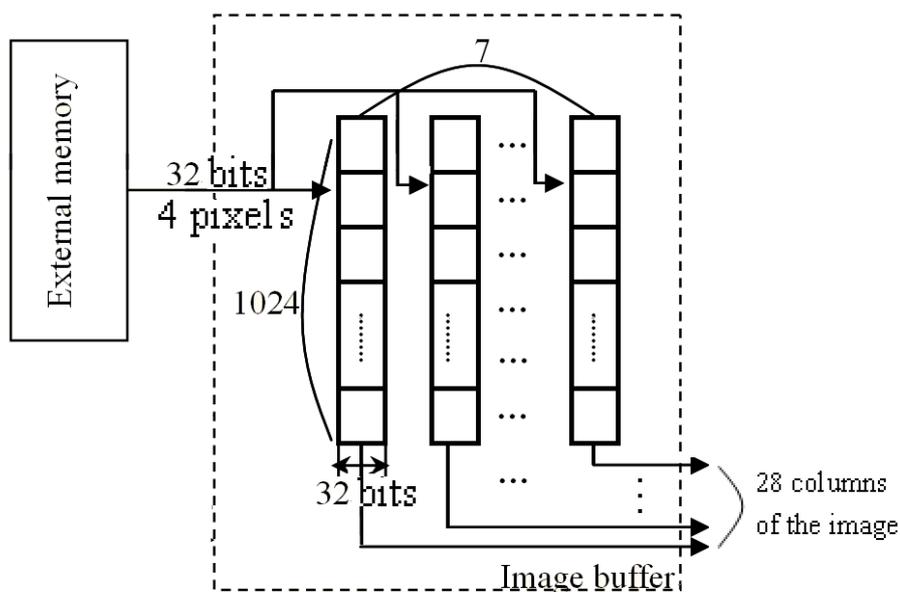


Figure 4.5 Architecture of image buffer

图 4.5 图像缓冲器结构图

该缓冲器采用的是 7 条 32*1024 的双口 SRAM。其优势在于以下几个方面：

(1) 采用双口 SRAM，使图像读取可以与数据处理同步进行。

由于数据处理时间一般是图像读取时间的 4-5 倍，因此完全可以覆盖图像读取的时间，使整体时间节约 20%。

(2) 最大效率利用了总线的传输能力。

其 32 位的宽度与总线宽度保持一致，可以同时传递 4 个像素值，比单像素传递能力提高 4 倍。

(3) 将 SRAM 分成 7 条独立的部分，大大提高了数据吞吐能力。

因为每条 SRAM 都可以提供 4 个像素的值，所以该存储器最多能同时提供 28 个像素数据。实际操作时，考虑到在同一单元内的 4 个像素如果进行局部更新会增加数据读取次数，造成浪费。因此 SRAM 中只选其中 6 条同时提供 24 个像素值，剩余的一条作为替换，及时提供新的像素值。此结构比同样容量的单块 SRAM 能提供 6 倍的图像读取速度。

(4) 数据读取按窗口产生顺序，大大节约初始等待时间。

人脸在外部 SDRAM 的存储是以行为单位，如每行有 1024 个像素，则第二行第一个像素的位置是 1025 个。为提高效率，在第一时间内得到第一个检测窗口内的所有像素值进行后续检测操作，在第一次对 SRAM 进行填充时，并没有按照外部 SDRAM 的存储顺序进行数据的读取。数据仍然以行的方式读入，但每行只读取 6 个 32 位，即 24 个像素后就转而读取其下一行的 24 个像素。这样在读取 24 行的 24 个像素之后，就可以得到第一个检测窗口内的所有像素。而因为同一列的连续两个窗口只有一行不同，只需读取一行新的 24 个像素就可以更新窗口。窗口的扫描顺序是纵向扫描，即 y 坐标优先。两者结合，之后的人脸检测就可以和新像素的读取时间完全并行，大大提高了整体运行速度。

(5) 与数据处理同步更新，将从外部 SDRAM 读取次数降到最低。

当 6 条 SRAM 完全填充完毕后，之后的填充改为纵向，即读取第 7 列 32 位数据。当第一列所有的检测框检测完毕后，第二列的检测框与同行第一列的区别只有右边(一般是 2 个像素)的列的不同，此时要更新的数据就是取自新填充的第 7 列数据。同时还可利用之前的已读取进来的前几列数据，避免了从外界的重复读取。只有当检测框取的 24 个像素是来自第 2-7 条 SRAM，完全不用第一条 SRAM 数据之后，第一条 SRAM 就读入第 8 列 32 位数据供更新。这样，其实备用预先填充的 SRAM 是不断交替的，执行完整幅图像的检测，所有图像数据只从外部读取了一遍，最大限度减少了从外部 SDRAM 读取图像的次数，节约了大量的时间与功耗。

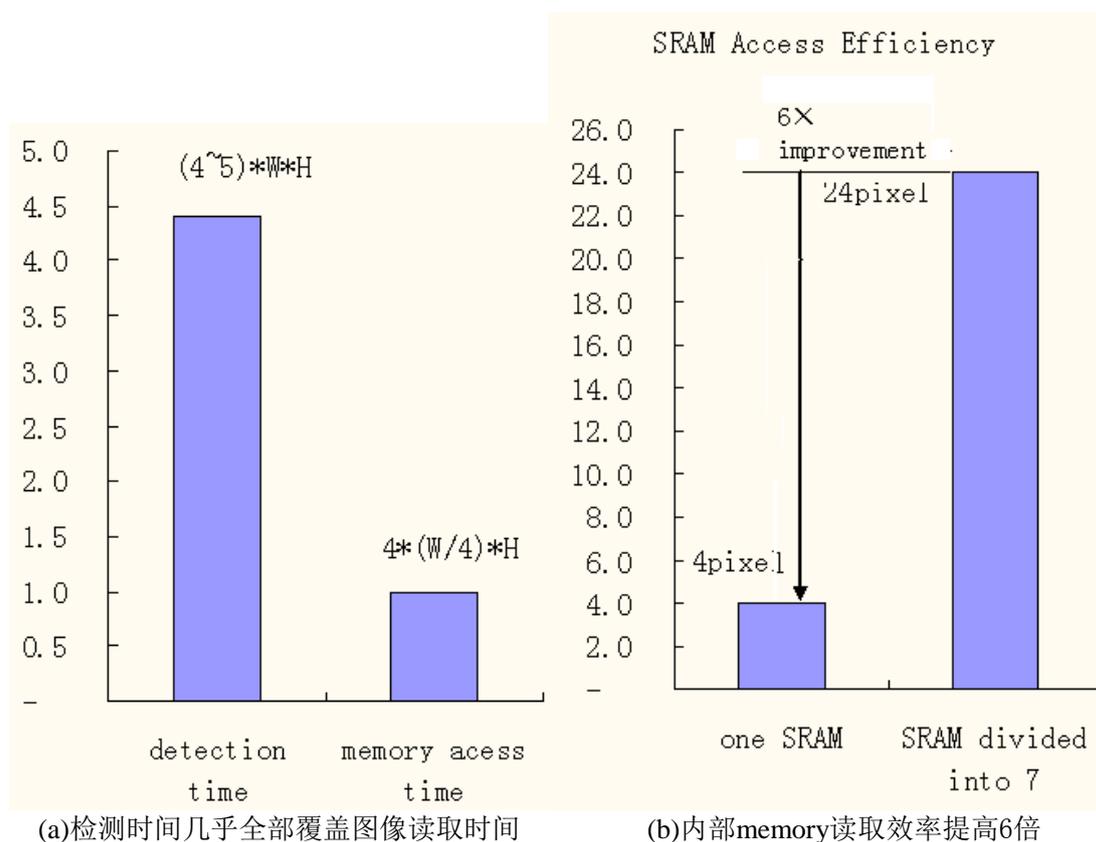


Figure 4.6 Efficiency comparison of memory

图 4.6 存储方案的高效性比较

图 4.6(a)显示了检测时间和存储器访问时间的比较，由于前者一般是后者的 4-5 倍，所以采用双口 SRAM 和特殊的数据读取顺序可以使图像读取时间几乎忽略，大大提高整体检测速度。图 4.6(b)显示了整体一个 SRAM 与将其分成 7 条后的效率比较。

4.2.2 图像存储与检测扫描策略

根据以上所述，将图像读入内部 SRAM 要进行数据装载，主要分为全装载和部分装载。所谓全装载就是读入完整的待检窗口的所有数据到阵列单元。在本设计中，它只发生在系统刚开始运行时，对第一个左上角检测窗口的装载。部分装载发生在第一个检测窗口之后对所有其他检测窗口数据的装载。由于同一列中，相邻的两个待检窗口有重叠关系，在窗口步进为 1 个像素的情况下，这两个窗口的其余 23 个行数据是完全相同的，可以共用。因此只需读取 1 行新数据即可。而同一行相邻的两个窗口之间也有类似的重叠关系。本设计中的前列数据是一直保留到该列数据无需再用为止，因此也可复用窗口共同部分数据。这两种情况都是执行部分装载。

检测时，整个图像的装载过程如下图，其与检测扫描方向是一致的。

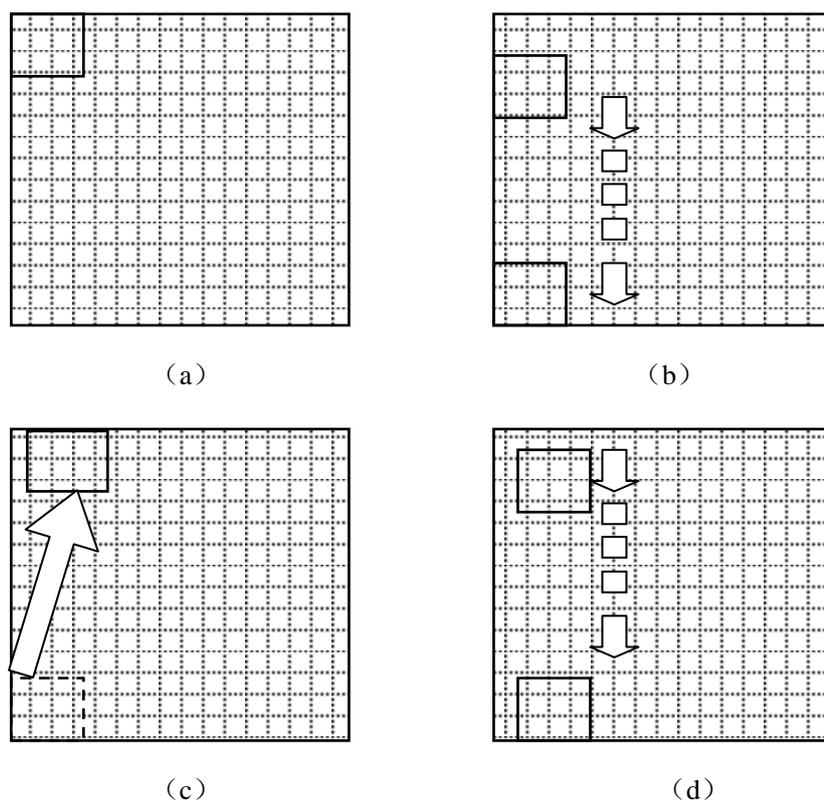


Figure 4.7 Image load and detection direction

图 4.7 图像装载、检测方向

当当前被检窗口未能通过当前这级分类器或者当前被检窗口通过所有分类器时，检测窗口需要读入一行新的数据。当检测窗口对当前图像扫描完毕以后，检测窗口重新对缩小一定尺度后的图像进行装载检测。本设计中的图像缩小模块(在 4.6 中将详细介绍)是与人脸检测模块同步工作的，当当前尺度的人脸图像检测完毕后，缩小后的人脸图像也得到了，立即可以进行下一个尺度人脸的检测。当被检图像被缩小到和检测窗口差不多大时，即完成一幅图像的检测。

为保证图像缩小与检测的同步性，而又要使图像缩小模块不与图像缓冲器模块读取图像数据相冲突，图像缩小模块的原图数据被取自图像缓冲器。这种处理还有一个好处就是数据获取快速，比从外部 SDRAM 读取数据要快得多。但由于图像缓冲器只是缓冲部分数据，并且在不停的更新，同时其又在进行图像数据的写入与提供给积分图窗口数据的工作，如何实现从外部读取图像数据到缓冲器，并将其提供给积分图窗口以及图像缩小模块这三者不冲突，不出现图像缩小模块读不了图像数据或没有相应的图像数据可读的问题，是一个难点。

4.2.3 图像数据读写策略

为了解决图像数据读取冲突问题，本文充分利用图像缓冲器双口 SRAM 的读取可同步的特点，并将该问题采用两个原则来解决。

一、图像数据的写入与提供给图像缩小模块的数据的读出，这两者可对同行(同一偏移地址)或不同行(不同偏移地址)的不同 SRAM 数据单元同时操作，但两个单元必须处于非同条 SRAM。

二、因为积分图窗口需要同时读取同行(同偏移地址)的 6 条 SRAM 的数值，其必须和图像数据的写入具有不同的 SRAM 偏移地址。

前者是为了避免两者对同一 SRAM 上同一单元的读写冲突，也为了避免出现两者在对同一 SRAM 进行读写操作时，积分图窗口也要对该 SRAM 进行读操作的情况。而后者是为了避免对同一单元的读写操作。

图 4.8 用两条 SRAM(竖)，每条 3 个单元的简化模式表示了读写操作的几种正确和错误模式。

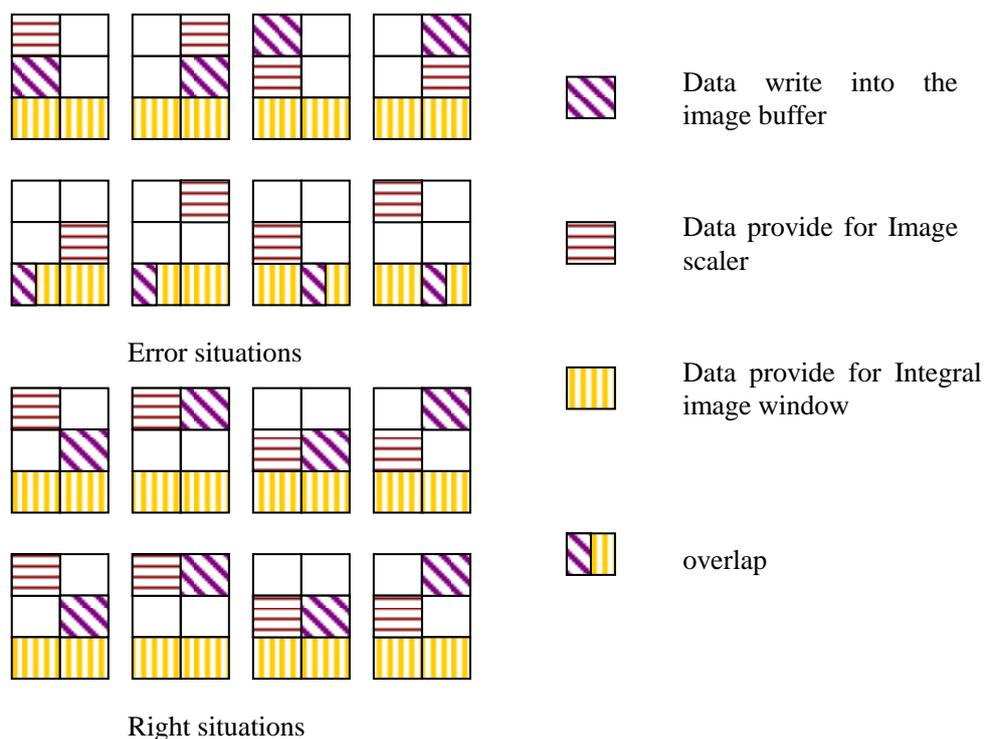


Figure 4.8 Error and right situations of data access

图 4.8 数据访存的几种正确和错误情况

图 4.8 中斜线格子代表提供给图像缩小模块图像的数据位置，横线格子代表写入图像缓冲器的数据位置，竖线格子代表提供给积分图窗口的图像数据的位置。一半斜线一半竖线的格子代表图像写入与读取同时。因为积分图窗口读取数

据时读取的是所有 SRAM 上相同偏离地址上的数据，即同行数据，所以图中每幅图中都有一行被其占用。其位置也可以是第一、二行，图中以其出现在第三行为例。图 4.8 中，第一排四种错误的情况是因为违背了第一条原则，从同一 SRAM 中进行了数据写入与图像缩小模块读取数据，当然如果两者还是对同一单元的操作，那更是错误的。第二排错误的原因是数据写入与积分图窗口读取数据同步进行，违背了第二条原则。第三、四排是正确的情況，其实另外还包括图像缩小模块和积分图窗口同时读取同一单元数据的情况，这里就不一一列举了。

除了高效的存储方案设计外，数据处理模块中积分图的快速更新和分类计算的四级流水线结构也使得处理速度大大提高。数据处理模块(Data process module)的结构如图 4.9。

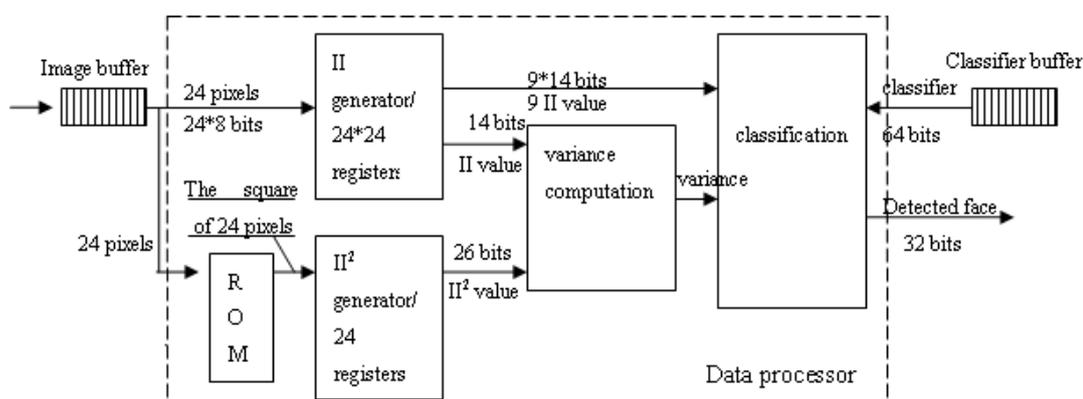


Figure 4.9 The block diagram of the data processor module

图 4.9 数据处理模块图

4.3 和 4.4 节将对积分图产生器(II generator)/平方积分图(II² generator)产生器的积分图快速更新法，分类模块(classification)的四级流水处理结构进行详细介绍。

4.3 积分图快速更新

因为积分图对特征的快速计算具有比像素大得多的优势，因此读取检测窗口对应的像素点后都需要进行积分图计算。积分图中的数据是特征计算的输入，其结构和更新速度因此直接影响了检测速度。本文中采用了寄存器的方式存储一个检测框(24*24)的积分图数据，并通过快速更新法，在一个时钟内更新相邻窗口的积分图，大大提高检测速度。

4.3.1 积分图寄存器的组成与结构

本文的积分图检测窗口由 24*24 个寄存器组成，每个寄存器存放检测窗口内对应每个像素点位置的积分值。寄存器的宽度最小为 8 位，最大为 18 位

$(\log_2(255*24*24))$ ，分别是积分图左上和右下角的积分值。之所以采用寄存器而不是 RAM 或 ROM，是为了可以使检测窗口内任意位置的积分图数据都可以同时被访问更新，提高检测效率。而且因为其只存储当前检测窗口的积分图，所以硬件的消耗并不大。

积分图寄存器结构及其与帧存储器，积分图数据访问之间的关系如下图所示：

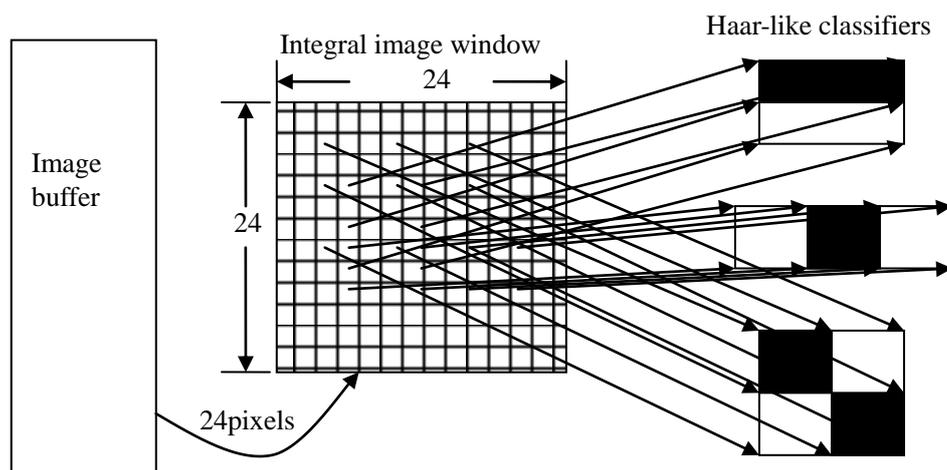


Figure 4.10 Integral image window and its simultaneous access

图 4.10 积分图寄存器窗口与其同步访问

4.3.2 积分图快速更新法

本文中所提出的快速更新法充分利用了相邻窗口之间的关联性、寄存器的随读随取特性，是一种基于流水线的单时钟积分图快速更新法。只有第一个检测窗口需要 29 个时钟的延迟，以填满原本全空的积分图窗口。之后只需一个时钟就可得到相邻窗口的积分图，数据更新极其快速。

整个积分图的更新主要可以分为两个步骤：

- (1) 得到检测窗口的行积分值。
- (2) 由行积分值得到整个窗口的积分值。

为简化图示，以 3*3 窗口为例，积分图的更新过程如下图所示：

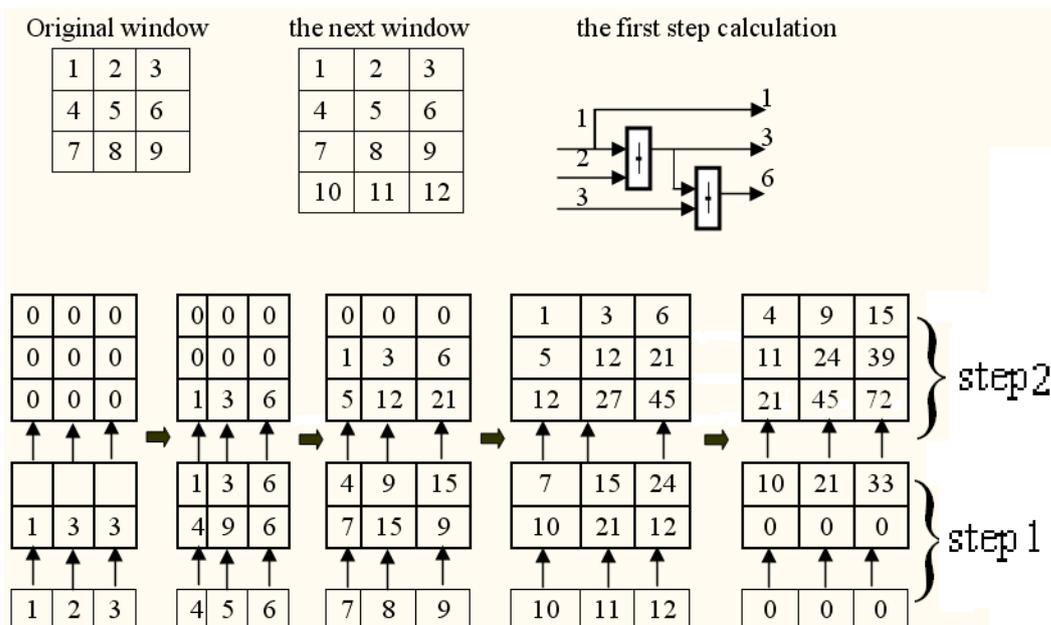


Figure 4.11 Generation and refreshment of integral image

图 4.11 积分图产生与更新

首先, 利用加法器计算行积分值。正如前所述, 图像缓冲器可同时提供检测窗口中一行的 24 个像素。为了得到这一行的积分图值, 越靠右边的积分值需要越复杂的加法器来实现, 最右边的积分值需要 24 个输入加法器才可以得到。如果要在一个时钟周期内由这 24 输入加法器计算得到结果, 加法器需要占用较大的面积。为避免这种超多输入加法器的使用, 且不影响行积分值的产生, 本文中采用了流水线的方式, 将行积分值通过 5 个阶段的流水线得到。这样虽在第一次计算时有 5 个时钟的延迟, 之后每个时钟都可以同时得到一行的积分值。控制了加法器面积的同时, 保证了积分图产生的速度。

为简化这一过程, 图 4.11 中用一行 3 个数(1, 2, 3)来表示。在第一阶段计算 $1+2$ 的值, 得到 3。保留 1 和 3, 在第二阶段将计算得到的 3 与第三个输入值 3 相加得到 6, 是第三个积分值。因此经过两级流水, 利用两个两输入加法器, 得到了该行的积分值(1, 3, 6)。而且在计算第二阶段 $3+3$ 的同时, 第二行数据(4, 5, 6)的 $4+5$ 运算也在进行, 那么在下一个时刻该行的积分值(4, 9, 15)也可以得到了。同样的道理, 24 个像素值的行积分值也可以通过 5 级流水, 12×5 个加法器得到。

得到行积分值后, 整个积分图寄存器组执行由下自上的脉动操作, 在每个时钟, 除最下一行外, 其余每一行寄存器的值都上移到上一行中, 并减去将要移出寄存器组的第一行数据。而最下一行执行以上操作外, 还需将得到的积分值与积分图寄存器值该行的值进行相加。在这样的工作方式下, 纵向相邻两个检测窗口

的积分图可以在一个时钟内得到快速的更新。更新过程详见图 4.11。(4, 9, 15)正是(5-1, 12-3, 21-6)的结果, 而(21, 45, 72)是(10+12-1, 21+27-3, 33+45-6)的结果。寄存器组初始值均为 0。

这种积分图快速更新法, 比传统 24 个数据的更新方式快了 24 倍, 对存储器的访问率也降低了 95.8%。

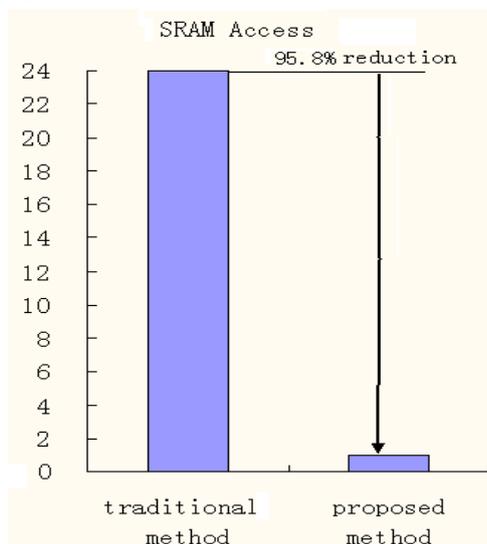


Figure 4.12 SRAM access comparison between traditional method and the proposed method

图 4.12 传统方法与提出的快速更新法在 SRAM 访问次数上的比较

4.3.3 平方积分图的产生与更新

如第二章中所述, 为了计算每个检测窗口的方差值, 以调整光线对人脸检测的影响, 在计算积分图的同时, 还需计算平方积分图。

在计算平方积分图时, 因为平方积分图的每个位置上的数值是其左上方所有像素的平方和, 所以牵涉到大量的平方运算。如果采用乘法器来实现这种平方运算, 那么平方积分图的产生速度将与乘法器的个数和结构直接相关。以每次产生 24 个像素的平方值为例, 需要 24 个 8*8 的乘法器同时工作, 而且如果需要在—个时钟周期就得到乘积结果的话, 乘法器需要的硬件资源极大。而较少的乘法器会使平方积分图的产生速度变慢, 影响后续特征计算的进行。所以本文中采用 ROM 代替 24 个乘法器, 采用查找表的方式实现平方值的计算。为了进一步减小 ROM, 查找表只设置了 0-127 共 128 个数字的平方值, 而 128-255 数值的平方值可以通过公式 4.1 获得。

$$s = (x+128)^2 = x^2 + 256x + 128^2 = x^2 + 2^8 x + 2^{14} \quad (4.1)$$

$x+128$ 为大于 127 的任意灰度值, x 即为该值的低 7 位。S 为该值平方值。从公式中可知, 其平方值为 x 的平方值、 x 与 2^8 的乘积值以及常数 2^{14} 三者的和。

由于 x 的平方值可以从 ROM 查找表中找到，而 2^8 与 2^{14} 均通过简单的移位操作就可以求得，所以其平方值也就不难求得了。

其电路结构如下：

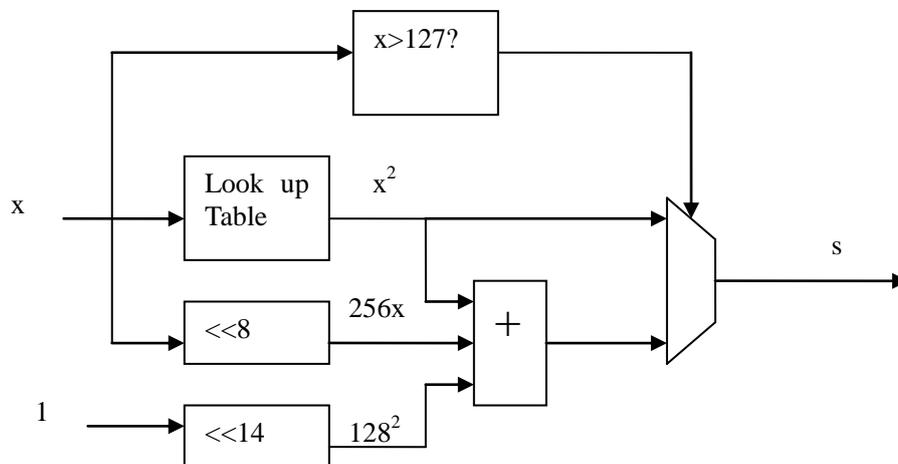


Figure 4.13 Square value circuit

图 4.13 求平方值电路

从该电路中可看出， x 小于等于 127 的值可以直接通过查表求得，而大于 127 的值可通过简单的两个移位操作和一个加法器操作得到，大大简化了原先需要多个乘法器的电路。

但是如果采用和积分图一样的存储方式，用 24×24 个寄存器存储平方积分图，由于平方积分数据较大，所需要的寄存器位数就更宽，此寄存器组面积也比积分图寄存器组要大的多。所幸当考虑到用于方差计算的只是每个检测窗口最右下角的平方和值时，可以发现并不是检测窗口中所有的平方积分图数据都需要存储下来，只需存储最右边的 24 个数据，并且像积分图一样更新就可以满足运算的需求。因此计算每行的平方图时，也只需保留最右边的平方和数据即可。平方积分图计算过程如下：

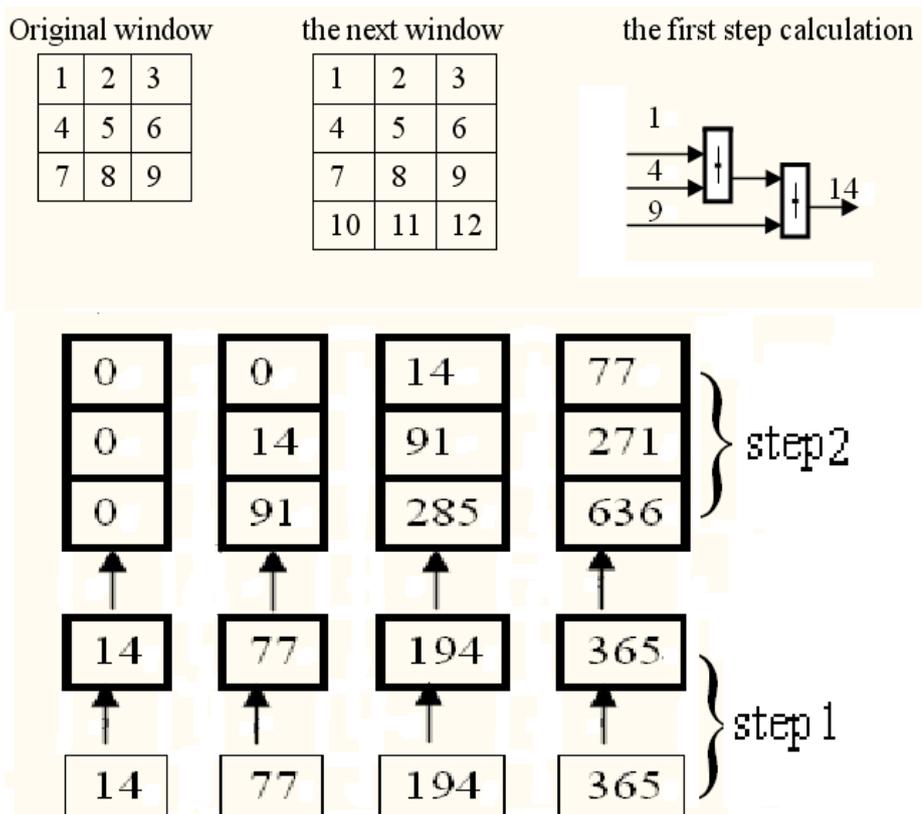


Figure 4.14 Generation and refreshment of square integral image

图 4.14 平方积分图产生与更新

更新过程与积分图一致，因此我们可以同时得到检测窗口的积分和平方积分值。

4.4 四级流水特征计算结构

4.4.1 常用特征值计算结构分析比较

在人脸检测中，特征值的计算是贯穿始终的一个重要步骤，其计算速度与功耗对芯片的性能影响很大。为提高其计算效率，充分挖掘算法中的并行度是很有必要的。文献中一般采用以下几种方式：从粗到精依次为不同尺度的检测窗口之间的并行、图像的不同检测区域之间的并行、同尺度的检测窗口之间的并行、弱分类器之间的并行。

(1) 不同尺度的检测窗口之间的并行

由于人脸检测需要经历不同尺度的检测窗口，因此可将这些检测窗口中对特征的计算并行处理。但是此种做法，会导致两个后果：1)需要同时存储不同尺度的检测窗口数据；2)由于不同尺度的检测窗口数是不一样的，各窗口不能完全并行，总计算时间取决于窗口数最多的那个尺度。

(2) 图像的不同检测区域之间的并行

此做法是将输入图像分割成几个部分，各部分并行进行特征值计算，即检测操作。为保证分割地带的区域也被检测到，分割时往往要使相邻区域之间重叠与检测窗口一样的宽度。但这等于增加了所要处理的区域面积，增加了计算时间。特别当所需要处理的图像较小时，重叠部分的区域甚至占了所需要处理区域的绝大部分，造成巨大的计算浪费。

以上两种并行方式中并行运算的几个部分划分得都比较“粗”，比较适合在多核处理器中的应用，但对于嵌入式应用来说，并不适合。所以本文不予采用。

(3) 同尺度的检测窗口之间的并行

它根据级联检测中每个强分类器来设置各级流水，设分类得到的强分类器总数为 n ，则当流水线填满时，能实现 n 个检测窗口的并行，其并行结构如下：

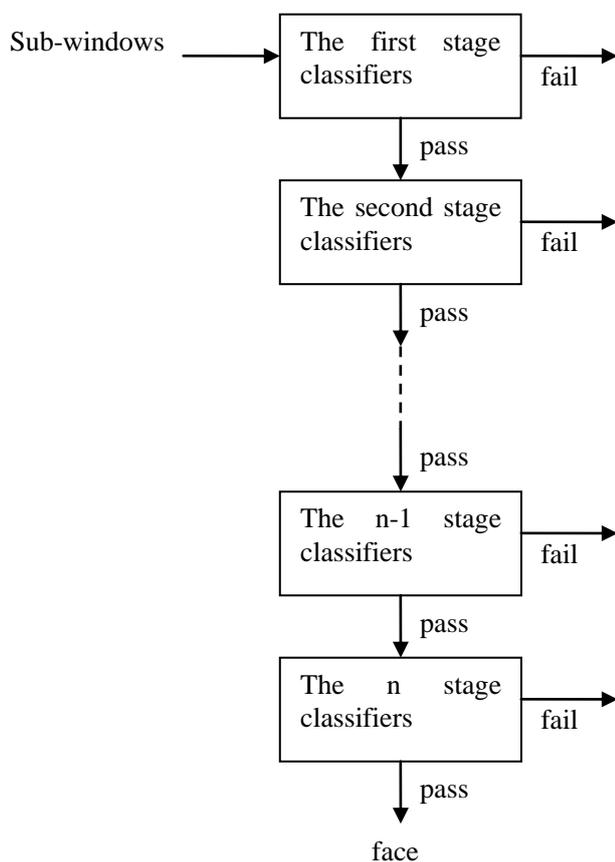


Figure 4.15 Pipeline architecture of strong classifiers

图 4.15 强分类器流水线并行结构

图 4.15 中输入的是每个检测窗口的积分图数据。当第一个检测窗口完成第一个强分类器处理，开始进行第二个强分类器的处理之时，第二个扫描窗口开始进

行第一个强分类器的处理。之后当第一个检测窗口进行第三个强分类器检测时，第三个检测窗口就可以进入流水线进行第一个强分类器的检测了。在检测过程中，如果检测窗口没能通过某级强分类器，即该级强分类器判定该检测窗口为非人脸，则检测窗口不再进行下一个强分类器的检测，但这一过程不影响别的检测窗口的检测过程。最终能通过所有强分类器检测的窗口即被认为是人脸区域。

这种流水线架构提高了数据的并行处理能力。但这种结构的缺点也是明显的。AdaBoost 算法训练得到的每一个强分类器能够让几乎所有输入这一级的包含人脸的检测窗口通过，而拒绝绝大部分非人脸的检测窗口。这就使得越是后面几级的强分类器，其所需要处理的检测窗口越少。而根据级联结构的特点，随着级数的增大，包含的弱分类器个数会急剧增多。这样导致的结果就是，层数较高的强分类器消耗了大量的硬件资源，但其所需要处理的检测窗口却很少，长期处于闲置状态，导致了大量硬件资源的浪费。而且，同时要提供 n 个窗口的积分图数据，这在软件实现上比较方便，而对于硬件结构来说，是不适用的。

(4) 弱分类器之间的并行

弱分类器是特征计算的基本单位，这种并行方式属于较为“精细”的并行，它并行计算的数据取自同一检测窗口，比较符合硬件特点。根据分类器的不同并行程度，此种并行结构又可细分为：全并行、部分并行。

a) 全并行结构

弱分类器全并行结构，对同一窗口取得的 Haar-like 特征数据进行所有弱分类器的并行计算。由于人脸检测一般需要几千个弱分类器，因此这样的设计可以达到很高的并行度。理论上，如果 P 个弱分类器模块同时工作，处理时间可以减少到原先的 $1/P$ 。然而由于人脸检测的级联检测特点，前几级的弱分类器执行频率较高，而后几级分类器经常处于闲置状态，所以全并行结构并不能达到理想的减少时间的目的。而且这么多的弱分类器也需要消耗比较大的硬件面积。

b) 部分并行结构

鉴于分类器全并行结构的以上缺点，文献[55][56][49][50]对其进行了改进。文献[55]使用五种不同个数(1, 2, 4, 6, 8)弱分类器的并行，减少了硬件面积。但它的弱分类器并行个数的划分并不能和每级中弱分类器个数或其倍数保持一致，所以在检测时仍存在着硬件的浪费。文献[82]建立了一个通用的可配置强分类器，该分类器内包含若干个能并行工作的通用可配置弱分类器，通过仅并行计算强分类器内的弱分类器加速系统的检测速度。文献[83]通过分析每一个强分类器的通过情况提出了一种更激进的架构，让前两级强分类器并行工作，而以后的分类器采用同文献[82]相同的方式工作。但这两种方式也都只是部分解决了资源浪费问题(通用可配置强分类器中弱分类器需要按照最大需求设置，仍会造成资

源浪费)。文献[56]考虑到前几级被执行的频率较高,因此只将前几级的弱分类器并行,而后面几级采用顺序执行方式,实行部分并行操作。但是新的问题也随之产生:到底前面多少级的分类器并行才能达到最佳的性能?文献[56]因此对每一级的执行频率和每级的执行时间进行了统计,指出使用该方法时,并行级个数的选定应在电路面积和处理性能之间折中考虑。但是该结构的并行划分还是个很模糊的问题,部分并行方案仍存在着缺憾。

基于以上分析,本文提出了符合弱分类器执行特点的可配置的弱分类器内四级流水处理结构。其在执行效率和硬件面积上都占有较大的优势。

4.4.2 四级流水处理结构

如第二章所述,弱分类器的特征计算过程包括对 Haar-like 特征里的矩形差的计算和方差调整后与设定门限的比较过程。其计算可分四个阶段完成。根据这一特点,本文提出了按照这四个阶段划分的四级流水处理结构。它只占了一个弱分类器特征值计算所需的硬件资源,但当流水线充满时,能同时执行四个弱分类器计算。每个时钟周期都能完成一个特征值的计算。

另外由于人脸检测主要的 Haar-like 特征类型有四种,输入的特征顶点 4-9 个不等,考虑到计算过程的类似性,如果对每种特征都使用不同输入个数的弱分类器会造成硬件资源的浪费,因此采用了如图 4.16 所示的输入可配电路结构。

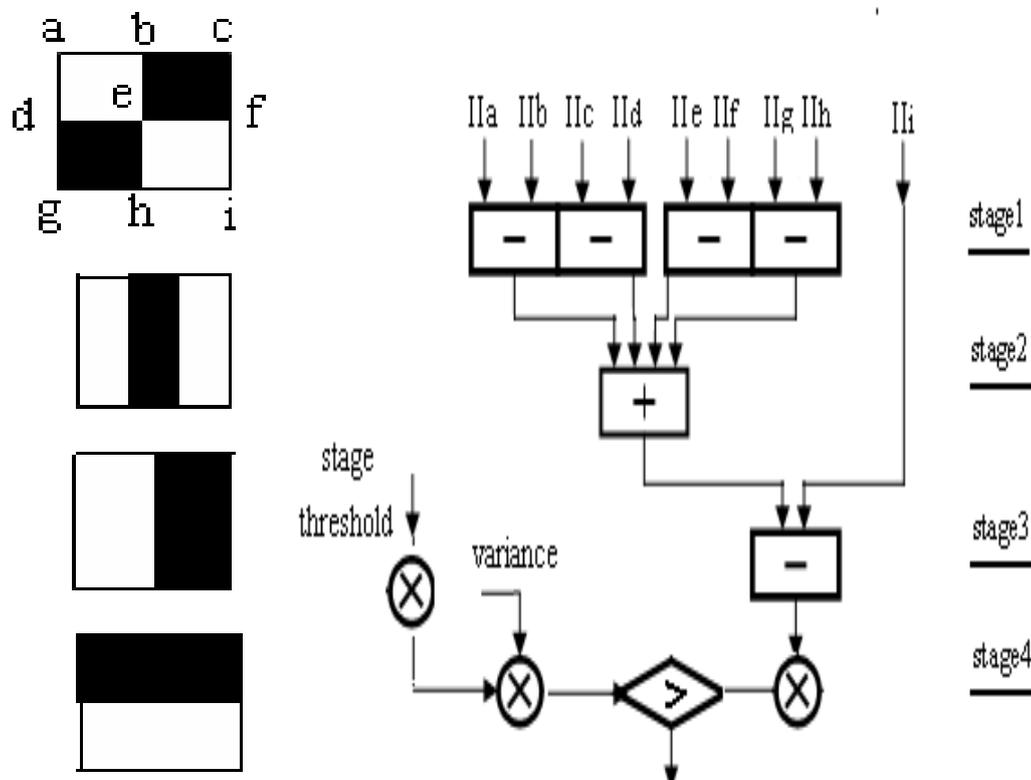


Figure 4.16 four-stage pipeline structure of weak classifier calculation

图 4.16 弱分类器特征值计算的四级流水结构

输入统一为9个，当特征顶点数小于9时，将剩余端口闲置即可。这样，任何特征都可采用该结构处理了。

表 4.3 中将这种四级流水线结构与常用的三分类器并行的结构进行了比较。为保证比较的前提一致，两种结构使用相同的人脸检测算法，这样级联结构中每个强分类器包含的弱分类器数均相同。

表 4.3 三分类器并行结构与四级流水处理结构的比较
Table 4.3 the time comparison between parallel and pipeline architecture

Stage	Number of classifiers	The windows left for further detecting	Cycle numbers in a stage (3-classifier parallel)	Total cycle numbers in a stage(parallel)
1	2	7967	4	31868
2	3	4058	4	16232
3	9	2451	12	29412
			Total cycle numbers of stage1-3(parallel)	77512
Stage	Number of classifiers	The windows left for further detecting	Cycle numbers by pipeline architecture	Total cycle numbers in a stage(pipeline)
1	2	7967	5	39835
2	3	4058	3	12174
3	9	2451	9	22095
			Total cycle numbers of stage1-3(pipeline)	74068

表中，Stage 代表的是级数，表中列出了前 3 级，每级中包含的弱分类器数分别是 2, 3, 9 个。为了比较整体处理时间，表中还列出了在前三级各自需要处理的窗口数（数据统计自 CMU+MIT 数据库中包含人脸的图像第一尺度检测的窗口）。那么，就可以用该级窗口数乘以该级处理需要的时间得到所有窗口通过该级所需的检测时间。

表中数据证明三分类器并行处理结构，处理第一、二级均为 4 个时钟周期，即弱分类器处理一个特征的时间。处理第三级 9 个特征分三次进行，共需 12 个时钟周期。而四级流水结构处理每级特征时，只第一个特征需要 4 个时钟周期，以后每个特征都可以在一个时钟周期内得到。因此三级所需时间分别为 5, 3, 9 个时钟周期。将前 3 级的检测时间相加，发现三分类器并行处理结构需要 77512 个时钟周期，而流水线结构是 74068 个时钟周期，少于前者。显然，随着级数的增加，流水线结构在这种速度上的优势会更加明显。图 4.17 是三分类器并行结

构和四级流水结构所需要耗用的电路门数比较，后者是前者的三分之一，电路面积下降了 66%。

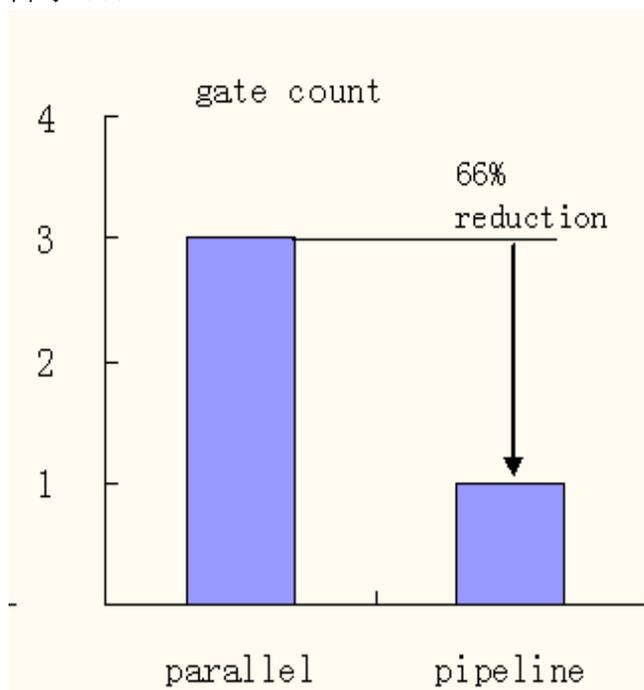


Figure 4.17 Gate count comparison between two architectures

图 4.17 两种结构电路逻辑门比较

当然，如果采用更多的分类器并行，其处理速度很有可能高于本文所提的四级流水结构，但是这是不值得的，因为其硬件处理面积会大幅增加。

4.5 分类器选择与存储

用于人脸检测的弱分类器是采用 AdaBoost 训练得到的，整个级联检测的弱分类器分布已在表 2.2 中表示。共有 19 级 1596 个特征。在检测过程中，这些特征要及时提供给检测模块用于特征值的计算，因此设立了分类器缓冲器预先缓冲分类器数据，以便以第一速度提供。

而且为了节约存储空间，进一步加快从缓冲器读取特征的速度。本文中将每个弱分类器的所有信息都精简成 64 位，并放在缓冲器的一个单元中，这样在一个时钟周期内就可以得到一个弱分类器的数据。

Haar-like 特征的信息包括特征的类别、出现的位置、面积大小、门限和权重。好在之前选择了固定检测窗口的方法，因此用 5bits 就可以表示特征出现的(x, y)位置，以及特征的长和宽。因为共有 4 种 Haar-like 特征，且将特征中不同方向的减法(例如当要相减的两个区域位置互换时)算做两个特征，则共有 8 种，因此

特征的类别可用 3bits 表示。另外，权重和门限分别需要 16bits 和 24bits 去表示。所以总共需要 63bits 去保存所有的特征信息。因为本设计采用 32bits 宽的总线，每个特征需要 2 个时钟周期去获得，获得后并成一个 64 位数据，存于分类器缓冲器的一个单元中以提高从缓冲器取数据的速度。分类器缓冲器结构如图 4.18。

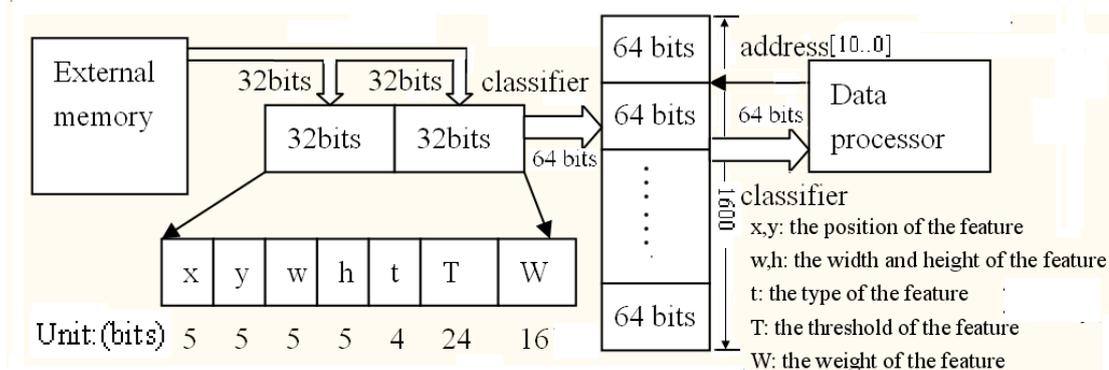


Figure 4.18 The architecture of classifier buffer

图 4.18 分类器缓冲器结构

缓冲器由 64bits 宽 1600 字的单口 RAM 组成，在系统初始化时，所有的特征数据都将存储进来，然后在特征计算阶段提供给检测计算模块。

4.6 图像缩小模块

如第三章所述，本文采用近邻插值图像缩小法实现不同尺度人脸的检测，图像缩小模块便是执行该缩小工作的电路，它与人脸检测同步执行。为了避免随着缩小比例的增加，读取的原图数据越为分散而导致读取速度变慢的问题，也为了配合缩小模块不从外部 SRAM 读取数据，只从内部 RAM 读取以加快读取速度的特点，图像缩小模块只进行一个尺度的缩小操作。即当原图正在进行处理的同时，产生比原图小一个比例的第一个尺度图像，并保存在外部 SRAM。当原图处理完毕，从外界读取第一尺度图像时，图像缩小模块又产生比第一尺度图像小一个比例的第二尺度图像，再将其存在外部 SRAM，如此循环往复。

图像缩小模块结构如图 4.19。

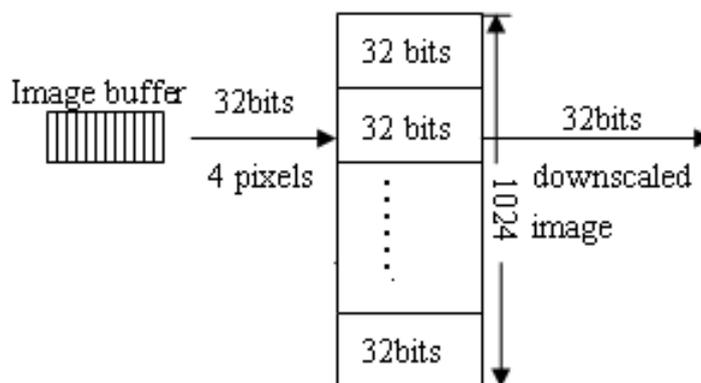


Figure 4.19 The architecture of image scaler

图 4.19 图像缩小模块结构

如图所示，该模块主要由一个单口 32 位宽 1024 字的 RAM 构成，用来存储计算所得的缩小图像的数据。因为图像像素为 8 位，因此可同时存储图像的 4 列。这样的结构设计是为了使其更新与图像缓冲器一致，与图像缓冲器的纵向更新方式同步。该缓冲器的存在，也使得计算所得的数据不需时刻传送到外部 SRAM，影响图像缓冲器从外部读取数据，当该缓冲器数据填满时，才连续进行向外传递数据的操作，而此时图像缓冲器一般已经缓存了足够的数据供给积分图窗口使用，所以不耽误整体的计算时间。

设计中默认采用的缩小因子是 1.25。该缩小因子被证明有比较好的检测结果 [25]。当图像被缩小到等于小于检测窗口的大小(24*24)时，缩小模块停止工作。

4.7 人脸检测芯片的仿真、综合结果与比较分析

将基于以上分析设计的人脸检测模块进行 RTL 代码编写之后，在 modsim6.5 的软件平台上实现了其功能仿真。仿真得到的算法检测率、虚警率等结果与定点化后的人脸检测结果一致。以图 4.20(a)的人脸图片为例，图 4.20(b)为其在 Visual C++6.0 开发环境下得到的检测结果，图 4.21 为其 RTL 级仿真结果。

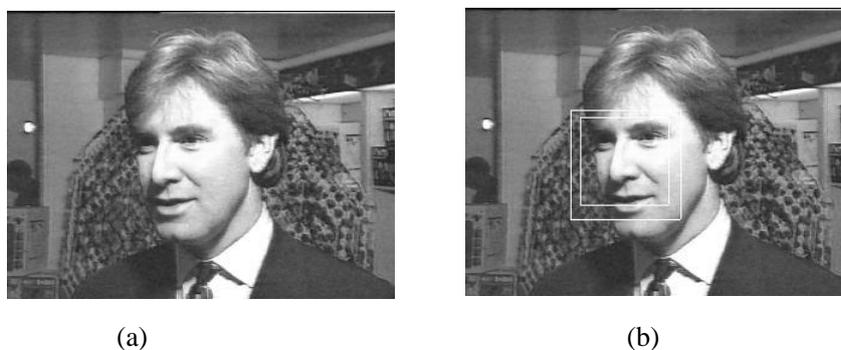


Figure 4.20 Face image and its detection result with software simulation tool

图 4.20 人脸图片及其软件检测结果

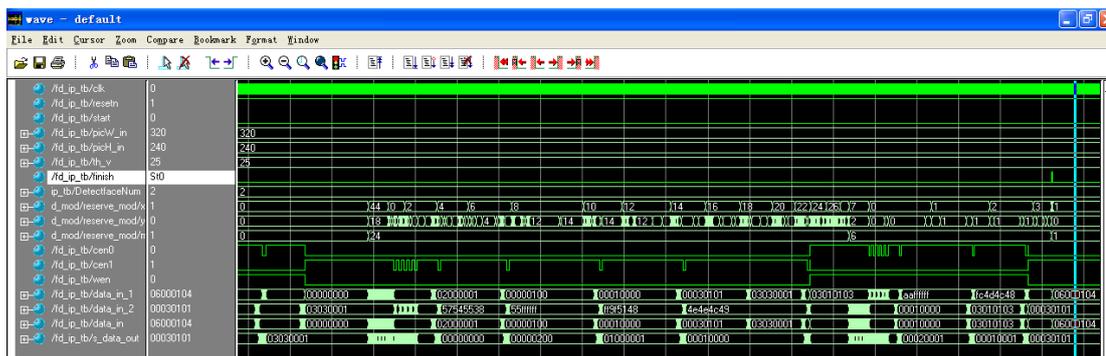


Figure 4.21 RTL simulation result of the face image

图 4.21 人脸图片的 RTL 级仿真结果

图 4.21 证明，RTL 级仿真中，最终检测得到的人脸个数、位置及人脸大小与软件仿真结果完全一致。

然后，进一步使用了 Synopsys 公司 TSMC65nm CMOS LP 工艺库进行了该人脸检测模块的综合仿真。该模块占用了 1.2mm² 面积，可以在 1.2v 的电压下跑 100Mhz 的时钟频率，功率消耗为 1.7mw/fps。具体结果见表 4.4。文中还选取了目前典型的几类发表于国际前沿杂志上的人脸检测芯片进行了性能的比较。

表 4.4 综合结果比较

Table 4.4 Comparison of synthesized results

	Theocharis[51]	Hanai[63]	Hanai[64]	Chih-Rung Chen[66]	Theocharis[60]	Presented work
Technology (nm)	160	130	90	90	65	65
Clock frequency(MHz)	73.75	100	54	167	800	100
Area (mm ²)	30.4	0.79	0.89	0.64	37	1.2
Speed(fps)	424	30	8	390	133	80
Frame size	300*300	160*120	320*240	160*120	320*240	320*240
Power(mw/fps)	17.3	0.97	0.47	0.36	2.45	1.7
Accuracy	79.57%	92%	81%	81.57%	95%	95%
Database	CMU	unknow	CMU+MIT	CMU	unknow	CMU+MIT
Algorithm	NN	skin	AdaBoost	AdaBoost	AdaBoost	AdaBoost
Year	2004	2007	2009	2010	2011	2011

比较结果可以显示出以下几个方面：

- (1) 随着检测技术的进步, AdaBoost 算法逐渐被接受, 原先的神经网络和皮肤检测法等由于受条件限制较多, 已渐渐淡出了历史舞台。
- (2) 面积、速度、功耗、检测率之间经常是互相制约, 此消彼长。从表中可看出 Hanai 的两种结构设计比较重视低功耗、小面积, 但是其速度与检测率都不高。而 Theocharis 的设计速度上占有明显的优势, 但其面积占有率和功率消耗却都很大。chen 的设计具有了较低的面积和功耗, 且速度也不低, 但是其检测率偏低的同时, 针对的是 $160*120$ 的小图像, 适用范围受到严重限制。
- (3) 几者相比较, 本文提出的人脸检测芯片在达到较高检测率的同时, 也满足了实时要求, 而且占用的面积、功耗都较小。具有较明显的优势。另外, 本文所提结构最大能处理 $1024*1024$ 大小的人脸图像, 这是其他结构都不能达到的。
- (4) 结合人脸检测在视频通信、数码相机、人机交互、目标监控、智能机器人等多种场合的实际应用需求, 本文的性能显然最为满足。适应了当今对高分辨率图像的高检测率要求, 又能在保证实时性的同时, 达到较低的功耗与较小的面积。

4.8 人脸检测系统的 FPGA 验证

在 ASIC 设计流程中, 验证和调试所花的时间约占项目总周期的 70%。为了缩短验证周期, 在传统的仿真验证的基础上, 涌现了许多新的验证手段, 如断言验证、覆盖率驱动的验证, 以及广泛应用的基于现场可编程器件(FPGA)的原型验证技术。

FPGA 验证是指在完成 RTL 设计以后, 将设计综合成网表实现在 FPGA 上, 通过分析系统输入输出, 或者配合外围接口电路, 完成对系统功能的验证。FPGA 验证并不是 ASIC 设计中必要的一环, 而是与 ASIC 设计流程并行的一个过程。理论上讲 FPGA 能够验证的, 其他软件仿真器都可以验证, 但是, FPGA 验证具有以下优点[52]:

- (1) 速度快。对于大规模的设计比一般仿真器仿真速度可能快数十倍, 接到逻辑分析仪上看还是很快的;
- (2) 把设计变成了一个物理实际的东西, 很多仿真或者静态时序分析看不出来的问题都能及时发现, 而且也能一定程度上代替综合工具及时解决代码的可综合性问题;
- (3) 加快开发进度。如果模拟部分模块未完成, 可先对数字模块进行验证, 加快软件开发进度。

FPGA 原型验证和其他验证方法不同的是, 任何一种其他验证方法都是 ASIC

验证中的一个环节，而 FPGA 验证却是一个过程。由于 FPGA 与 ASIC 在结构、性能上各不相同，ASIC 是基于标准单元库，FPGA 用的是厂商提供的宏单元模块，因此首先要进行寄存器传输级(RTL)代码的修改。然后进行 FPGA 器件映射，映射工具根据设置的约束条件对 RTL 代码进行逻辑优化，并针对选定的 FPGA 器件的基本单元映射生成网表。接着进行布局布线，生成配置文件和时序报告等信息。当时序能满足约束条件时，就可以利用配置文件进行下载。如果时序不能满足约束，可通过软件报告时序文件来确认关键路径，进行时序优化。可以通过修改约束条件，或者修改 RTL 代码来满足要求。

4.8.1 人脸检测 SOC 系统介绍

本文在 FPGA 上验证了所设计的人脸检测芯片及其外围的整个 SOC 系统。整个 SOC 系统的结构如图 4.22 所示，各模块之间通过 AMBA 总线进行通讯。AHB 担当高性能系统的中枢总线，支持处理器，片上存储器，片外存储器以及低功耗外设宏功能单元之间的有效连接。AHB 也通过使用综合和自动测试技术的有效设计流来确保减轻使用负担。APB 优化了最小功率消耗并且降低了接口复杂度以支持外设功能。

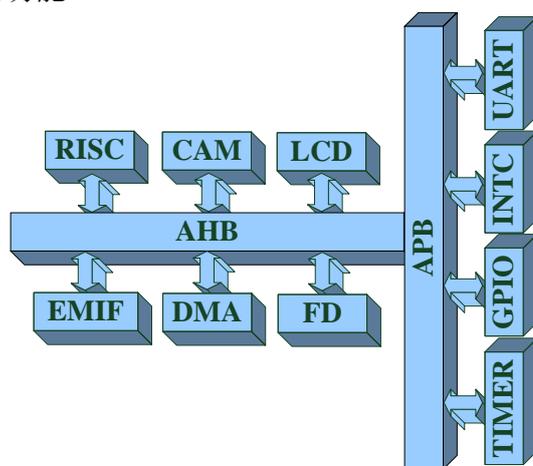


Figure 4.22 SOC architecture of face detection

图 4.22 人脸检测 SOC 结构框图

主要模块功能介绍如下：

(1) MIPS32 处理器(RISC)

MIPS32 处理器核主要包括一个高性能、低功耗的 32 位 RISC 处理器，系统控制器 CP0，片内存储 Cache，内存管理单元 MMU 以及总线接口 BIU，如图 4.23 所示，通过软件编程和硬件配置，MIPS32 处理器可以方便地应用在数字消费及网络市场。

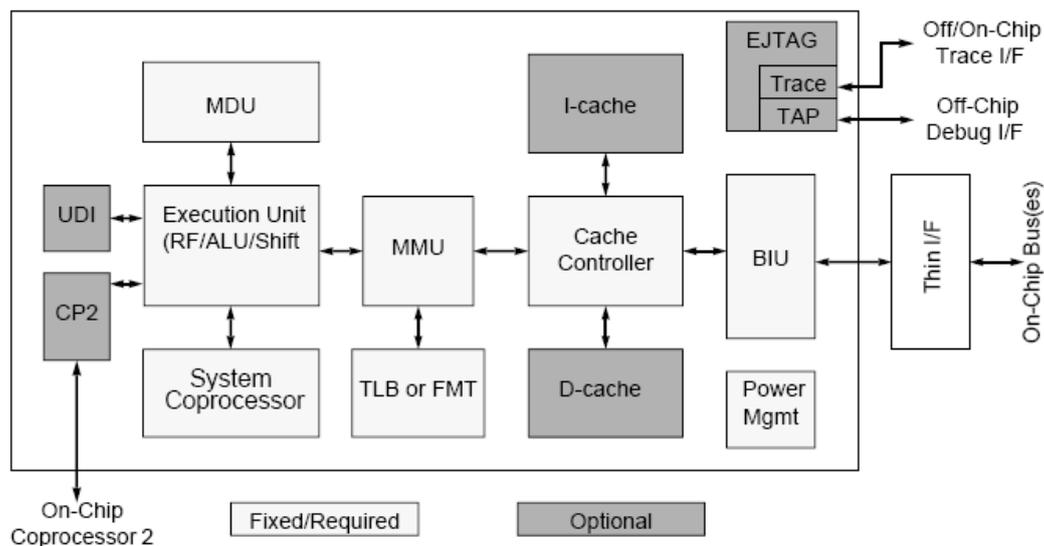


Figure 4.23 The MIPS32 architecture

图 4.23 MIPS32 结构示意图

本文的 SOC 系统中所有模块的工作模式通过 MIPS32 处理器来配置实现, 包括人脸模块工作模式的配置, LCD 模块工作模式的配置, CAMERA 工作模式的配置, UART 工作模式的设置, GPIO 工作模式的设置, 中断模块工作模式的配置。本文采用的 MIPS 处理器, 其软件开发采用 Linux 下的 sde-gcc 编译器, MIPS 中运行的软件代码通过读文件的方式, 写到 MIPS 的片上 ROM 中。

(2) Camera 控制器(CAM)

Camera 控制器是视频输入采集转换模块。该模块接收来自通用视频输出模块的视频信号, 本文采用 OV9650 摄像头模块, 经过视频处理和转换后, 通过 AHB 总线, 将 Camera 捕捉到的数据存储到 SDRAM 中。该 IP 支持 ITU-R BT 601/656 8-bit 模式, 支持 YCbCr, RGB 格式输入。支持 YCbCr, 24bits RGB 输出, 支持 16bits RGB5:6:5 格式输出。

(3) LCD 控制器(LCD)

LCD 控制器通过 AHB 总线, 从 SDRAM 中读取数据, 经过处理后送到 LCD 上显示。该 IP 兼容 AMBA 协议, 包括 AHB SLAVE 和 AHB MASTER 接口, 通过 AHB SLAVE 接口配置控制器内部寄存器, 确定其工作模式, 通过 AHB MASTER 接口读取 SDRAM 中的显示数据。该 IP 支持 YCbCr, 24bits RGB, 16 bits RGB5:6:5 格式输入。

(4) 人脸检测模块(FD)

人脸检测模块通过 AMBA 总线从 SDRAM 中读取图像数据, 存储在内部 RAM 中, 检测完成后, 把检测结果存储在内部 RAM 中, 同时通过中断的方式

通知处理器检测结束，需要新的检测数据。

(5) 通用输入输出(GPIO)

本文使用OV9650摄像头采集图像数据，由于工作模式和图像输出质量等，需要通过配置内部寄存器实现，而寄存器配置是通过IIC协议实现；同时，LCD屏的工作模式也需要配置内部寄存器实现，其寄存器的配置通过SPI协议实现。为了避免同时引入IIC和SPI模块，本文采用GPIO模拟IIC和SPI协议，在一定程度上降低了系统复杂度。

(6) 串口(UART)

本系统提供一个RS232通讯接口，主要用于系统调试和部分数据收发。上位机和MIPS的通讯都是通过RS232实现，通过串口查看摄像头和LCD的配置结果，通过串口发送MIPS的boot代码等。

(7) 其他可配置模块

系统中其他的可配置模块，包括中断管理模块和定时器模块。

4.8.2 FPGA 硬件平台

本文采用Altera公司的Stratix2开发板，作为人脸检测的FPGA硬件验证平台，如图4.24所示，使用到的硬件资源如表4.5中所示。

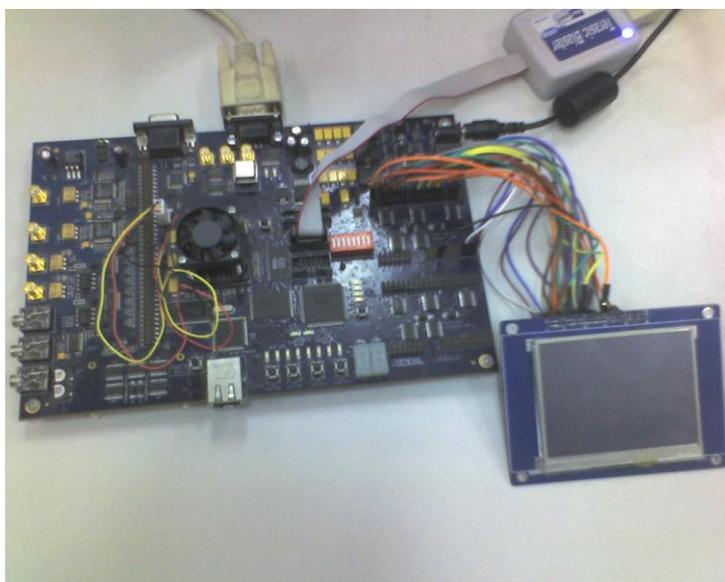


Figure 4.24 FPGA development platform

图 4.24 FPGA 开发平台

表 4.5 开发板硬件资源
Table 4.5 Develop board hardware resources

名称	参数	数量
FPGA	Altera, Stratix2, EP2S180F1020	1
SDRAM	32 Mbytes	1
RS-232	9-pin母头	1
晶振	100-MHz	1
电源	16-V DC	1
下载器	USB-blaster	1

Altera 公司的 EP2S180F1020 型 FPGA 采用的是 90nm 工艺，内核工作电压 1.2V。其包括的主要资源如表 4.6 所示。

表 4.6 EP2S180F1020 主要资源
Table 4.6 EP2S180F1020 main resources

类型	数量
ALUTs	143520
LEs	179400
RAM bits	9383040
PLLs	12
I/O	1170

4.8.3 FPGA 软件平台

本文采用的主要软件开发环境见表 4.7 所示。

表 4.7 主要软件平台
Table 4.7 Main software platform

软件名称	功能	运行平台
ModelSim	系统功能时序仿真	Windows
Quartus	FPGA综合、布局布线	Windows
Signal Tap	信号采集、设计装载	Windows
Sde-gcc	MIPS软件开发	Linux

ModelSim 工具用来对设计进行时序仿真。Quartus 是 Altera 公司的集成开发环境，主要完成 RTL 的综合，布局布线等，其中提供了一些宏单元供用户使用。本文主要利用该功能进行各种存储器的定制，而其集成的 Signal Tap 可以用来采集 FPGA 内部信号，对 FPGA 设计内部的信号进行观察。同时，通过 USB-blaster 把配置数据下载到 FPGA 板。MIPS 中运行的代码，是通过 Linux 系统下的 sde-gcc 编译器生成的。

4.8.4 系统测试

FPGA 综合频率采用了 100MHz。系统工作频率为 54MHz 时，整个系统的资源使用率见表 4.8 所示。摄像头采集、LCD 显示、人脸检测、处理器画框均可实时显示检测结果，达到了系统设计和应用的需求。

表 4.8 FPGA 资源使用情况表
Table 4.8 Resource allocation of FPGA

模块	资源类型	使用数	使用率
RISC	逻辑单元	7479	5%
	存储器(bits)	524288	6%
CAM	逻辑单元	2658	1%
	存储器(bits)	4096	<1%
LCD	逻辑单元	6813	4%
	存储器(bits)	0	0%
FD	逻辑单元	67532	38%
	存储器(bits)	167936	1%
总和	逻辑单元	87628	49%
	存储器(bits)	694720	7%
	Pin	103	14

4.9 人脸检测 SOC 测试

人脸检测 SOC 芯片集成了人脸检测模块、MIPS 处理器、RS232 接口、LCD 控制器、Camera 控制器等，采用 SMIC 0.13 μ m CMOS 标准工艺制造。图 4.25 是该 SOC 芯片的裸片照片，裸片面积 5mm*3mm，采用 160 脚 CQFP 封装，如图 4.26 所示，封装面积 28mm*28mm。

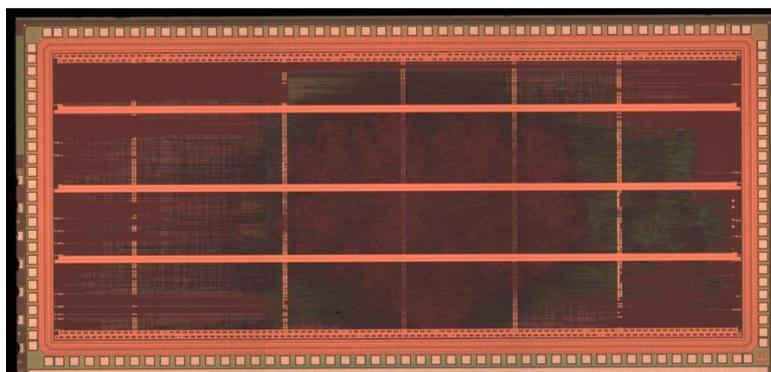


Figure 4.25 Face Detection chip photo

图 4.25 人脸检测芯片

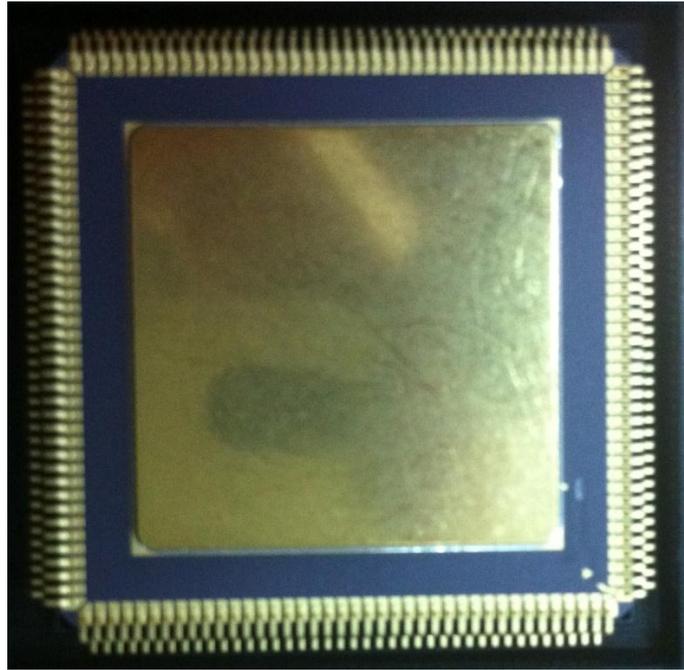


Figure4.26 the packaged chip photo of CQFP-160

图 4.26 CQFP-160 封装

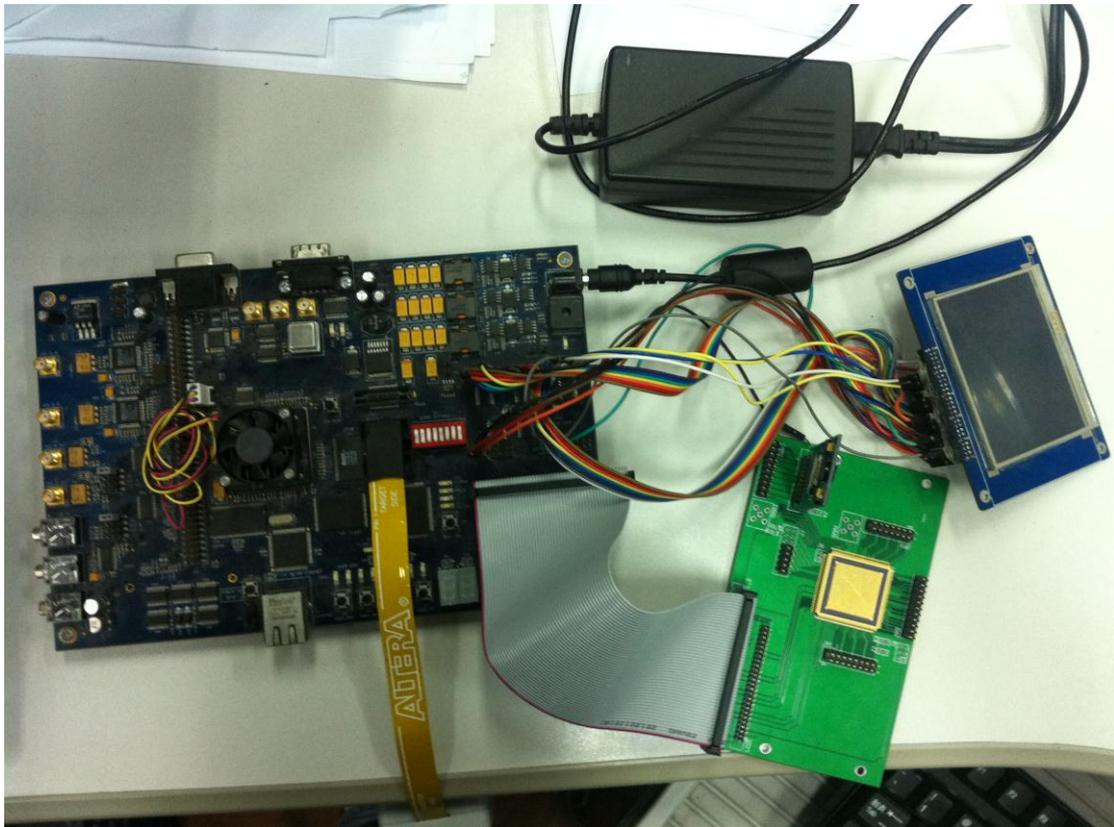


Figure4.27 testing platform of face detection chip(1)

图 4.27 人脸检测芯片测试平台（一）

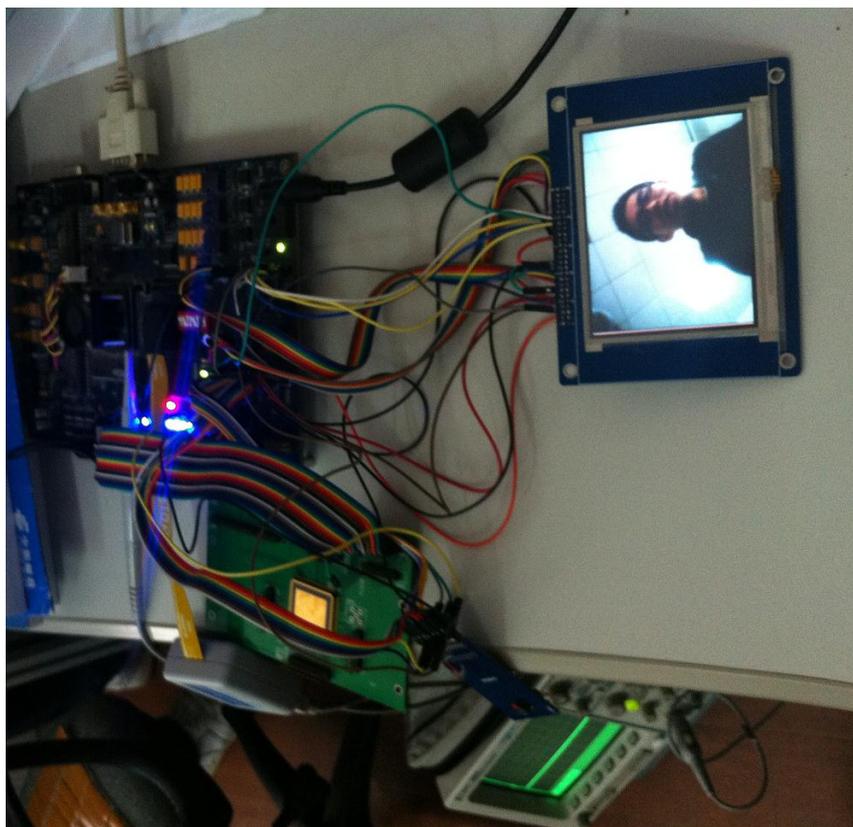


Figure 4.28 Testing platform of face detection chip(2)

图 4.28 人脸检测芯片测试平台（二）

SOC 测试电路平台如图 4.27, 4.28 所示, 主要包括人脸检测芯片, 电源电路和 SDRAM。电源电路提供 1.2V、3.3V 和 5V 电压, 1.2V 为芯片内核供电电压, 3.3V 为芯片 PAD 和 LCD 供电电压, 5V 为 Camera 供电电压。由 SDRAM 存储视频数据, 人脸检测特征数据。芯片的工作频率通过 GPIO 配置, 测试结果目前尚在统计中。

4.10 本章小结

本章主要介绍了高性能的人脸检测芯片的整体设计与各模块的工作特点, 并简要介绍了包含人脸检测芯片与外围模块的整个 SOC 系统的结构。介绍了各部分功能模块的作用、测试过程中使用的软硬件平台。给出了人脸检测模块在 FPGA 上的综合结果, 包括资源利用率, 系统运行性能等。

第5章 多模式可调人脸检测

本章首先就多模式可调人脸检测芯片设计的重要性，面临的难点进行介绍，然后根据人脸检测过程中的关键因素，把握多模式可调点，详细描述了芯片的结构与性能。该功能的实现可使人脸检测的性能随着应用场合的不同而灵活调节，适应多种场合应用的需要，节约多次开发的成本。

5.1 人脸检测芯片的可配置性研究

正如第一章中所述，随着社会的发展，人脸的应用场合飞速增长，但不同的应用场合往往对人脸检测的精确度、资源消耗等指标有不同程度的要求。在一种应用场合中工作良好的检测系统，并不一定能符合另一种应用的需求。比如用于人流统计的人脸检测系统在小区安防应用中就不能发挥最佳作用。而目前的人脸检测系统都是只偏重于某一个或几个指标。如表 4.4 中所提的几种人脸检测系统，Theocharis 的研究成果主要偏重于速度和精确度，对功耗和面积消耗要求不高。而 Hainai 和 Chih-Rung Chen 主要关注面积和功率消耗，针对的只是小分辨率输入图像。如果像这样针对每一个应用环境的特殊情况开发符合各项指标的人脸检测系统，固然能很好的满足其需要，但是由此带来的人力和物力的消耗也是很大的。尤其对于一些移动性设备，随着外界应用环境的改变需要能灵活快速改变检测性能指标。于是，人脸检测的未来发展又面临了一个新的挑战，那就是如何同时满足各种不同场合对检测的不同性能要求？

有些学者已经就该问题做了一些探讨。文献[60]中就提出了一个脉动阵列的结构，该结构采用了较多相似的子结构组成，非常便于进一步裁减和扩展以满足不同大小图像和不同速度要求的人脸检测。这是从设计改变便利性的角度去解决可配置的问题，但是这没能避免对硬件结构的重新设计，而且一旦硬件结构确定，它的性能也不能再改变了。

为了更便利的调整人脸检测性能，一些研究者用可编程逻辑硬件，例如 FPGA 和其他类似的可编程硬件平台去实现其可配置性。如文献[53]中，将人脸检测的关键部分采用可配置逻辑模块实现，这些模块的电路结构可以通过软件编程轻易改变。但是可编程逻辑模块，需要消耗大量的硬件资源，又往往不是最简电路，消耗的功耗也相对较多。文献[54]将其可配置性基于一个 DMV (Digital Machine Vision) 平台，也同样利用了该平台具有的软件可编程性。分析这类可配置的系统，其可配置性虽然实现方便，但是付出了巨大的资源成本代价。

分析这些前期研究，要实现人脸检测芯片的可配置性，难点主要在于以下几个方面：

- (1) 可配置往往要以牺牲芯片面积为代价，且有时候牺牲的面积还较大。
- (2) 移动设备的人脸检测需要随应用场合需要随时改变，一般 IP 的设计可配置性不能满足该要求。
- (3) 可配置性与高性能不能兼顾。

为解决上述问题，本文抓住影响检测性能的几个关键因素，采用寄存器存储这些变量，再由这几个变量控制整个电路工作于不同的模式。使用的寄存器及相关电路因为只涉及简单的判决，占用的芯片面积极少。而且由于寄存器中的数值可以通过用户赋值改变，这种可配置性是用户可调的，不需改动硬件电路，非常适合随着应用场合改变而快速变动。解决了便携移动设备多场合应用的需求。另外，此结构完全延用了前述高性能人脸检测 IP 的主体部分，兼顾了性能和可配置性。

5.2 多模式可调人脸检测分析

从前面对整个人脸检测过程的描述以及实现过程中可以看出，对于 AdaBoost 级联检测算法来说，有四个因素会影响包括检测率和检测速度、图像大小等等的检测性能。对它们的合理控制能使人脸检测系统呈现不一样的检测性能，以适应各个应用场合的需要。这四个因素如下：

(1) 采用的强分类器的数目

根据级联检测结构的特点，每级强分类器中包含的弱分类器的数目并不一样，越是后几级的强分类器，包含的弱分类器越多。这是为了不断提高人脸检测的精确度而设计的。但由于大部分的非人脸区域已经在前几级排除了，后几级排除的非人脸个数相对较少，却因为弱分类器数目的众多而占据较长的检测时间。

以 $320*240$ 图像为例，表 5.1 显示了对原图进行检测时，每个强分类器所包含的弱分类器数及其所要进行分类的原图中检测窗口数。每个强分类器中所含的弱分类器数在训练完成后就固定了，在整个检测过程中都不会改变。而子窗口数随着图像内容的不同会有所变动，但是总体变化趋势是相同的。

表 5.1 每级所对应的 Haar-like 分类器数与需检测的子窗口数对照表

Table 5.1 Haar-like classifiers and the sub-windows needed to be classify in every layer

层数	分类器数	子窗口数	层数	分类器数	子窗口数
0	1	31565	10	37	80
1	2	7967	11	53	50
2	3	4058	12	57	29
3	9	2451	13	81	24
4	14	1250	14	107	13
5	23	713	15	88	10
6	19	434	16	88	6
7	28	258	17	301	0
8	42	176	18	579	0
9	64	102	total	1596	

从表中我们可以看到，最后几级通常由几百个分类器组成，但是能排除或需排除的人脸个数却越来越少了。

对于需要高检测精确度的场合，保留所有的级联分类器是必要的，它能保证检测到的人脸的正确性。而对于有些比较关注目标的潜在位置而不在意错误接受率的场合（比如数码相机的拍摄），在整个检测中去除后面几级的强分类器可以减少我们对特征的读取时间和处理的时间，且对结果影响并不大，还是能符合应用场合的需求。事实上，当硬件资源有限的时候，这个方法经常被采用。文献[60]就是通过省略部分分类器达到快速、低功耗的目的。

(2) 输入图像的大小

对输入图像的大小进行可配置是非常必要的。由于不同应用场合中使用的传感器分辨率的不同，经常会得到不同大小的图像。如果为了统一图像大小而使用相同的传感器或特意将输入图像的分辨率改变，往往会引起检测精度或速度的下降。例如，如果我们将一个大分辨率的图像压缩成一个小分辨率的图像，那原先可用检测窗口检测到的比面积较小的人脸区域，可能因为压缩而变得小于检测窗口而被丢失。而如果采用小分辨率的传感器，将得到的小图像扩大，会凭空增加处理时的计算量。所以人脸检测系统对输入图像的大小有必要进行可配置化。

(3) 图像缩小比例

对图像缩小法执行的人脸检测，对图像的缩小要考虑两个因素：缩小比例和缩小次数。一般来说，图像缩小的比例设置为 1.25 时，可检测出图像中各种大小的图像。这在 Viola 和 Jones 提出基于 AdaBoost 人脸检测算法时就有所论述。

考虑到实际情况中，同一幅图像中出现的人脸大小的类型不可能差太多，我们可以适当加大缩小的比例以加速检测过程。虽然这可能引起某些情况下检测率的下降，但是相对于计算量的大幅度下降还是值得的。并且通过实验我们可以找到不同的应用场合中各自合适的缩小比例。

(4) 图像缩小执行次数

除了图像缩小比例，缩小执行次数是另外一个可以进行可配置的因素，它决定了可以被检测到的最大的人脸区域。理论上，为了保证任意大小的人脸均被检测到，如果采用的是图像缩小法，图像应该一直以一定的比例缩小，直到缩小后的图像小于或等于被检测的窗口。这样，即使像原始输入图像一般大的人脸也能毫无遗漏的被检测到。但以这种方法，处理分辨率较高的输入图像的计算量会非常大，而且在通常情况下，在很多应用场合中，人脸并不会占据整张输入图像。如果能按照应用环境，由用户自动调节图像缩小的次数，那计算量就会降低很多，而且不会引起检测精度的下降。因此对该因子进行调节，可控制检测精度、速度。

5.3 多模式可调人脸检测模块

既然人脸检测中的上述四个关键因素会影响整个系统的性能，我们可以通过对它们的控制达到人为可配置的目的。在这小节中主要介绍这种可调的人脸检测模块的设计，通过配置这些可调因子，使整个系统工作于不同的模式。

图 5.1 是整个多模式可调人脸检测系统的整体结构。“picW”，“picH”，“sl_step”，“sl_num”，“sg_num”正是决定系统可配置性的输入信号，它们在整个系统工作的初始状态时，保存于控制器的寄存器中以控制控制器发送不同的控制信号给各模块，使各模块工作在不同的方式下。控制的各模块包括分类器模块(Classifier buffer)、数据处理模块(Data process module)、图像缓冲器模块(Image buffer)、图像缩小模块(Image scaler)。各个模块的基本结构在第四章中均有详细介绍。这里只就增加的控制电路部分作一说明。

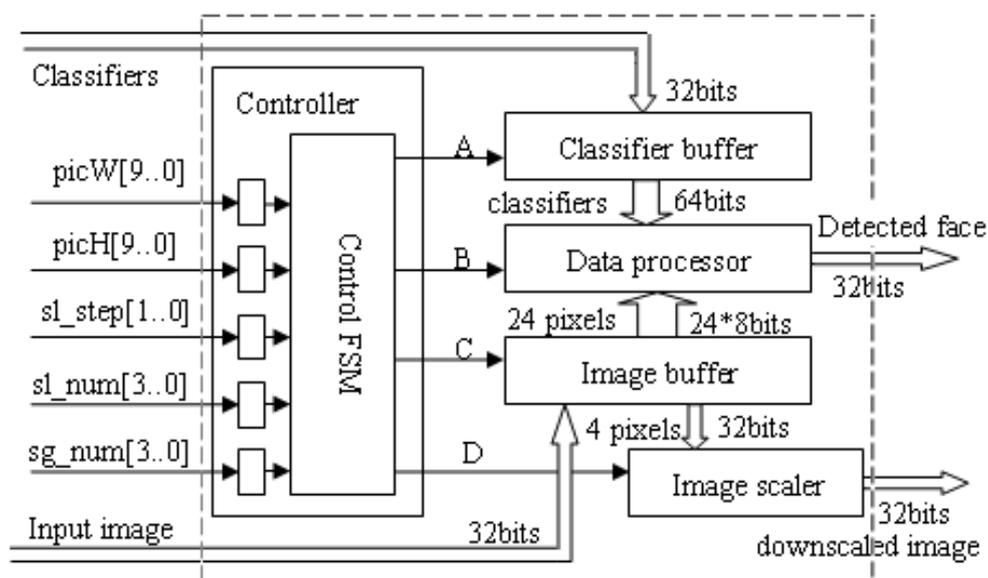


Figure 5.1 The overview of the reconfigurable face detection architecture

图 5.1 多模式可调人脸检测系统结构图

A, B, C, D 四个控制信号将控制对应模块执行在不同的方式下。如 A 信号控制分类器缓冲器读取分类器的个数，这决定于采用的强分类器个数，即“sg_num”决定。而 D 信号控制图像缩小模块执行缩小的比例和次数，这决定于“sl_num”，“sl_step”信号等等，控制信号具体由哪个或哪些输入信号决定的关系表见表 5.2。

表 5.2 控制信号与可调的输入信号之间的关系

Table 5.2 The relation between control signals and input adjustable signals

控制信号	决定控制信号的输入信号
A	sg_num
B	picW、picH、sg_num
C	picW、picH
D	sl_num、sl_step

信号“picW”和“picH”分别代表输入图像的宽度和高度，决定了输入图像的大小。考虑到分辨率为 1024*1024 的图像已经能满足很多场合所需，因此在设计中两者均被设置为 10 位，能最大处理宽度和高度均为 1024 像素的图像。“sl_step”信号决定了图像缩小的比例，共可设置 4 个值 1.25, 1.5, 1.75, 2。之所以设置这样的一个区间是因为缩小比例若大于 2，检测准确度下降太快，不适宜实际应用。而缩小比例如小于 1.25，则在计算量大幅增加的同时检测准确度与比例为 1.25 时几乎一样，所以没有意义。选择 1.25-2 之间的这四个比例，一方面可以在准确度较为稳定改变的同时，控制加速过程，另一方面由于四个比例的

小数部分都可以用有限的二进制表达，在定点化时比较方便。四种比例需用 2 位数值表示，所以缩小比例“sl_step”宽度为 2。“sl_num”可以配置图像缩小的执行次数。在设计中，它被赋予 3 位的宽度以代表 8 种可以设置的方式。当“sl_num”为“0”时，图像缩小将一直执行到图像大小小于检测窗口为止，在这种方式下，在原图中大于或等于检测窗口的人脸都可以被检测到。而当“sl_num”的值在 1-7 之间，图像缩小次数将分别被限定在 10-16 次。“sg_num”配置强分类器数。在设计中，强分类器最多可有 19 级，考虑到使用小于 12 级的分类器进行检测时错误接受率将非常大，不适合实际使用，实际设计时，“sg_num”的可调范围为 12-19 共 8 个级别，即“sg_num”也需要 3 位宽度来分别代表选用 12-19 级的强分类器。“sl_num”和“sg_num”，以及它们对应的缩小执行次数和强分类器的选择见表 5.3。

表 5.3 不同的 sl_num 和 sg_num 所代表的意义

Table 5.3 Represent of sl_num and sg_num in different values

sl_num	图像缩小执行次数	sg_num	使用的强分类器数
0	不限	0	12
1	≤10	1	13
2	≤11	2	14
3	≤12	3	15
4	≤13	4	16
5	≤14	5	17
6	≤15	6	18
7	≤16	7	19

综上所述，图像缩小比例有 4 种可调模式，图像缩小执行次数和强分类器数都可在 8 种情况下调整，因此系统的工作模式为 $4*8*8=256$ 种。也即本文中的人脸检测系统可以处理任意大小小于 $1024*1024$ 的图像，并且可以调整工作在 256 种不同的模式下，适应不同的应用场合，非常灵活有效。

当在初始状态，控制器内部寄存器存储五个可调的输入信号之后，电路的工作状态就根据这些信号进行相应的变化。控制器及其状态机如图 5.2 所示。

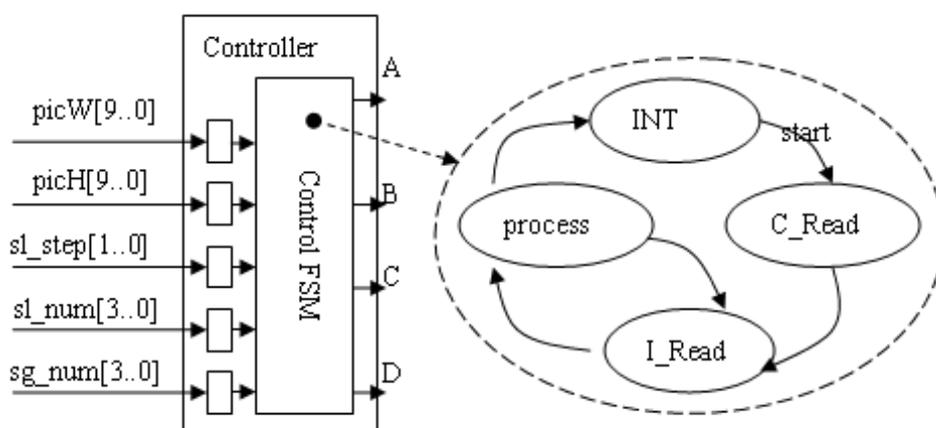


Figure 5.2 The controller and its state machine of multi-mode reconfigurable face detection system

图 5.2 多模式可调人脸检测系统的控制器及其状态机

一旦寄存器在初始状态(INT)更新完毕后, 系统就从初始状态进入了 C_Read 状态, 在这一状态中分类器模块(Classifier buffer)根据接收到的控制信号“A”, 从外界存储器读取一定个数的分类器信息。当需要的分类器信息读取完毕, 系统进入 I_Read 状态。在该状态中, 图像的数据将被陆续读取到内部的图像缓冲器(Image buffer)中来, 而图像的长宽等信息由 picW, picH 决定的“C”信号控制。根据第四章中介绍, 当第一个检测窗口的数据被图像缓冲器取得后, 系统将开始 process 状态, 在读取图像的同时进行人脸的检测, 以及图像的缩小操作。这时, 信号“B”, “C”, “D”同时控制三个模块协同工作。

5.4 可调模式相关实验与验证

5.4.1 可调性测试

为测试结构的可配置性, 做了如下实验以测试强分类器个数、图像缩小比例、缩小执行次数分别对检测准确度和速度的影响, 便于根据实际情况调配。这里采用的数据库来自于 CMU+MIT 人脸库, 包括 87 个不同大小的图片。

实验一: 强分类器个数的影响

本实验主要测试当采用的强分类器个数在 12-19 之间变化的时候, 检测准确度与速度的变化。这里假设采用第 1 个强分类器时标其级数计为 0, 那么强分类器数应在 11-18 个之间变动。这里, 检测准确度的变化使用第三章中所述的虚警率(FAR)和正确检出率(DR)两个指标来反映。得数据都是在缩小比例为 1.5, 缩小次数不限的前提下得到的。

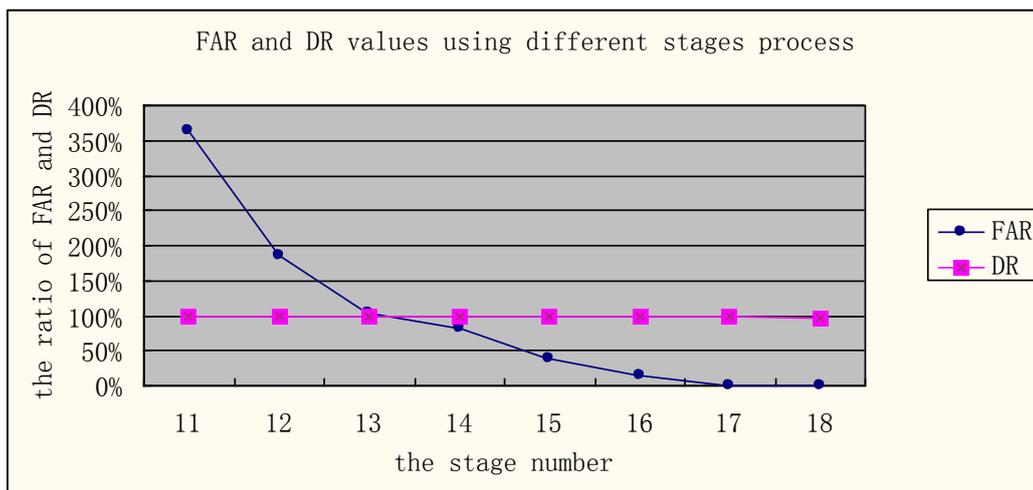


Figure 5.3 FAR and DR values using different stages process

图 5.3 使用不同个数的强分类器时 FAR 和 DR 的变化

从图中可以看到随着采用的强分类器个数的不同，虚警率和正确检出率各自具有不同的变化规律。正确检出率的变化非常小，而虚警率却下降地非常厉害。原因在于，本文所采取的人脸检测方法是一个不断排除背景，精确人脸区域的过程。在检测初期所有的区域都是潜在的人脸区域，而随着强分类器的增加，非人脸区域被逐步排除，在这个过程中，真正的人脸区域是始终包含在潜在人脸区域中的，因此人脸正确检出率并没有因强分类器个数的增加而变化。但这一过程中，被误认为是人脸的非人脸区域被逐步排除，所以虚警率大幅下降。但是，仔细观察，虚警率在不同个强分类器之间的下降幅度并不太一样。随着强分类器的增加，幅度逐步减小，表示其排除非人脸的个数在逐步减小。所以更可进一步证明，越是后面的强分类器，对整个检测的影响越小。

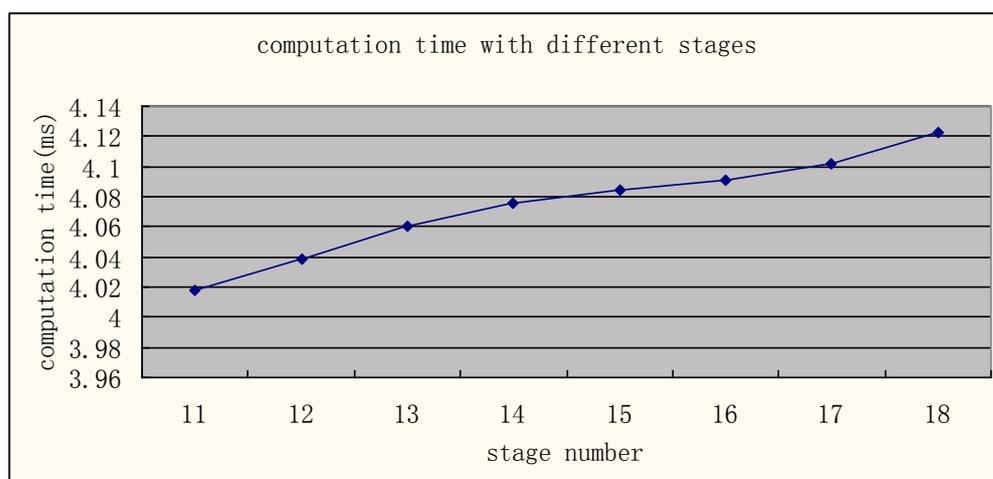


Figure 5.4 Computation time with different stages

图 5.4 使用不同个数的强分类器所需要的时间

图 5.4 显示了使用不同个数的强分类器时，检测时间的变化。在总时间增加的同时，检测时间的变化幅度是不一样的。在第 11-14 层时较大，15-16 层较小，17-18 层又恢复了比较大的幅度。根据图 5.3 和 5.4，我们可以得出结论：在速度要求比较高的场合，省略后几级强分类器的检测，不仅可以减少从外部存储器读取分类器的时间，而且可以减少分类时间。

实验二：图像缩小比例和缩小执行次数对检测性能的影响

表 5.4 中列举了在 4 种不同缩小比例因子下，CMU+MIT 数据库中人脸检测的平均准确度和执行时间。数据是在缩小次数不限的前提下得到的。

表 5.4 不同缩小比例因子下检测准确度和速度

Table 5.4 The accuracy and speed with different scale step

The step of scaling	Detection accuracy	Process time(ms)
1.25	95%	6.16
1.5	94%	4.12
1.75	90%	3.40
2	84%	3.15

如表 5.4 中所示，随着缩小比例的增大，检测准确度和时间都是下降的。要同时具有高准确度和高速度是矛盾的，两者之间存在着制约关系。但是在实际的某种应用场合中往往只对其中一种指标比较敏感，因此根据统计规律，选择合适的缩小比例，达到需要的检测准确度和处理速度是完全合适的。

另外，图像缩小的执行次数很大程度上是由实际情况中图片中的最大面积的人脸所确定的。所以我们要保证检测窗口最终能达到和最大面积的人脸区域一样大小。如果，在检测时能根据缩小后的图像的大小，停止图像缩小的操作，将在不增加任何计算量的情况下，达到与缩小次数不限的人脸检测同样的检测准确度。

实验 3：不同大小的人脸检测软硬件速度、准确度比较

在本实验中，将同样的人脸检测算法用工作在 600MHz 频率下的 DSP DM642 实现，测试了常见的 8 种视频规格下人脸检测的准确度与速度。检测结果见表 5.5。将该结果与硬件实现的结果相比较发现，两者的准确度基本一致，但采用本文所述结构执行的人脸检测速度几乎是 DSP 的 100 倍。

表 5.5 人脸检测对不同规格图片的软硬件检测比较

Table 5.5 The comparison of hardware and software implementation speeds in detecting different sized images

Image size	Software	Hardware	Image size	Software	Hardware
160*120	270ms	2.88 ms	640*480	2424ms	40 ms
176*144	372ms	4.12 ms	800*600	4852ms	62.5 ms
320*240	952ms	12.50 ms	1024*768	5205ms	83.3 ms
352*288	1045ms	13.33 ms	1024*1024	14184ms	125 ms

5.4.2 综合仿真结果

本文的设计使用了 Verilog 语言加以实现，仿真波形如图 5.5 所示。

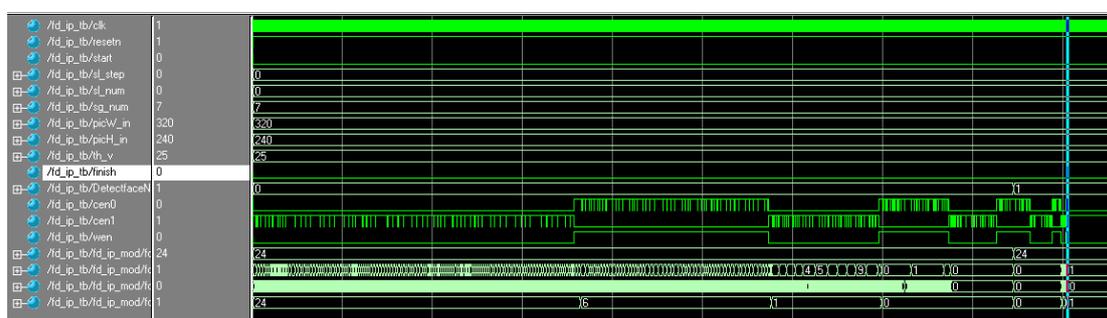


Figure 5.5 Simulation waveform of mode adjustable face detection chip

图 5.5 多模式可调人脸检测仿真波形

从图 5.5 中可以看到在设定的参数下，得到了人脸检测的定位结果。基于 SMIC 0.13um CMOS 标准工艺，本文还用 Synopsys 公司的 Design Compiler 进行了逻辑综合。综合结果表明，该模块占用了 5.3mm² 面积，可以在 1.2v 的电压下跑 100Mhz 的时钟频率，功率消耗为 1.8mw/fps。

5.5 本章小结

本章在指出人脸检测对多种模式和可配置工作的迫切需求后，对人脸检测的可配置可调因子进行了分析，发现有四个因素直接影响检测的各项性能指标，对它们的合理控制可以达到多模式可调的目的。本章也详细描述了多模式可调人脸检测模块的具体设计，并通过实验论证了在不同工作模式下检测模块所具有的性能，便于根据实际需求，对模块进行调整。

第 6 章 结论与展望

人脸检测最初只是作为自动人脸识别系统中的一个关键环节。随着科技的发展,人脸检测已作为一个独立的分支,在数码相机、人机交互、智能机器人等多个领域有了长足的发展,这也给高性能人脸检测芯片的设计带来了新的机遇和挑战。本论文主要研究具有高性能人脸检测芯片的低成本、低功耗设计,以及人脸检测芯片多模式可调可配的实现方案。在取得了一些研究成果的同时,也指出了人脸检测可进一步发展的方向。本论文的研究成果对其他类型的目标检测,如行人、路牌标志、车辆、船舶等的检测也具有直接的参考意义。

6.1 对本文工作的总结

为实现高准确度高速人脸检测,本文首先对目前最为高效的 AdaBoost 人脸检测算法进行了深入的分析,并根据算法特点进行了有效的算法改进。利用方差预处理手段大幅缩小检测区域,提高算法效率。采用图像缩小算法大大减小硬件实现时对外存数据的读取次数,为降低功耗、提高数据吞吐能力打好了基础。与此同时,对国际上公认的应用最为普遍的五个人脸检测数据库进行了介绍与总结,用这些数据库的测试结果证明了改进后人脸检测算法的有效性。

然后在进行了算法的定点化与芯片实现技术的一系列分析之后,第三章对高性能人脸检测芯片的设计进行了详细的描述。从存储器的访问次数和存储空间各方面对比分析了两大类六种人脸检测芯片的硬件架构,最终选择了存储部分原图像数据的帧存储器架构。接着对帧存储器的结构和工作方式进行了介绍,指出了采用的存储结构的五大优势,以及与存储结构相配合的检测扫描策略和图像数据读写策略。除存储方案的介绍外,第四章也对影响人脸检测速度与面积的关键部分,积分图的更新与存储,级联检测的结构,分类器,图像缩小模块等的设计进行了详细的描述,并仿真、综合、用 FPGA 验证了芯片的功能。

鉴于人脸检测应用的场合日益广泛,每个场合中对人脸检测又有不一样的性能要求,本文在第五章提出了多模式可调的人脸检测芯片设计。多模式可调比芯片的可配置设计更加灵活,它可以在不重新修改芯片设计的前提下,通过输入信息的改变,而使芯片具有不同的检测性能,这使得设计的人脸检测芯片可移动运用在多种系统中,节约了大量的开发、设计与资源成本。第五章对多模式可调设计的出发点以及设计的关键技术进行了详细介绍,并用相关实验给出了各种不同模式下芯片的性能,便于在各种不同应用场合中选择使用。

本论文的创新之处在于:

- (1) 提出了在不增加芯片复杂度的同时,可使计算量减少的方差预处理手段,结合级联检测算法由粗到精的特点,使检测速度获得了提高,芯片检测的功耗

也获得降低。

- (2) 提出了一个检测的高效存储方案，解决了人脸图片数据量大，需要多次访问外部存储器的缺点，有利于功耗的降低和检测速度的提高。
- (3) 提出了可单时钟更新的积分图快速更新法，为特征计算模块快速获取积分图数据提供了保障。
- (4) 采用独特的级联检测特征内四级流水线处理结构，加速了特征计算的速度。
- (5) 提出并实现了多模式可调人脸检测的概念，使得人脸检测在多种应用场合之间的切换更加高效灵活，节约了大量开发与资源成本。

6.2 对未来工作的展望

随着人脸检测应用的各领域越来越影响当今生活，人们对人脸检测的要求也必然会越来越高，主要体现在对准确度、速度的进一步要求上。以及如何能低成本低功耗的实现高质量的人脸检测。这可从以下几方面着手：

- (1) 通过对各类人脸检测算法的分析可以发现，统计学习方法在精确度上占有绝对的优势。只要保证足够多的训练模板进行学习，得到的分类器能满足大部分场合的人脸检测精确度要求。因此提高精确度的一个方法就是采集更加庞大的训练模板。但这里要指出一点，再多的模板也不能覆盖所有的情况，与其他手段相结合，弥补不足，是达到高精度人脸检测的另一个发展方向。
- (2) 人脸检测速度的提高还可继续从算法的简化变换上着手。这种简化要充分依赖人脸的特点。在本文对检测算法的介绍中可以看出人眼对人脸的位置其实并不十分敏感。由于现存的人脸检测算法的步进都是像素级的，检测的结果往往导致对同一人脸区域进行了多次的检测。而人脸区域属于计算集中地带，多次检测浪费的计算量非常可观。因此如使用基于前续的区域检测结果，自适应调整检测步进，对人脸和非人脸区域实行粗细不等的检测手段的方法，是实现高速人脸检测的一个方法。

另外从芯片的结构设计上，人脸检测也可进一步从检测路径的缩短着手，提高工作频率，加快检测速度。也可进一步采用多核等技术加以提高。

参考文献

- [] Hjelmas, E., Face detection: A survey, computer vision and image understanding. 2001: 236-274.
- [2]黎冰, 吴松与曾凡涛, 人脸识别在智能手机中的实现. 计算机工程, 2006. 32(7): 272-274.
- [3]蒋文荣, 基于手机的便携人脸远程比对系统. 江苏警官学院学报, 2008(5): 169-173.
- [4]卓力, 沈兰荪与张延华, 一种嵌入式头肩图像编码方法. 电子学报, 2003. 31(12): 1832-1834.
- [5]郝晓莉, 陈后金, 蔡伯根, 李杰.基于脸部检测和 FUZZY ART 的乘客检测算法.北京交通大学学报, 2007.31(5):19-22.
- [6]宋红与石峰, 基于人脸检测与跟踪的智能监控系统. 北京理工大学学报, 2004. 24(11): 966-970.
- [7]邢延超与强文萍, 基于人脸检测与跟踪的广告效果评估系统. 计算机应用, 2009(10): 2700-2702.
- [8]高永萍, 秦华标. 驾驶员疲劳检测系统.仪表技术与传感器, 2007,1: 60-62.
- [9]梁路宏, 艾海舟等,人脸检测研究综述.计算机学报, 2002. 25(5): 449-458.
- [0] Sung K, Poggio T. Example-based learning for view based human face detection. IEEE Trans Pattern Analysis and Machine Intelligence, 1998, 20(1): 39~51.
- [1] Yang MH, Roth D, Ahuja N. A SNoW-Based Face Detector. Advances in Neural Information Processing Systems 12, S.A. Solla, T. K. Leen, and K.-R. Muller, eds., pp. 855~861, MIT Press, 2000.
- [2] Schneiderman H. A statistical method for 3D object detection applied to faces and cars. In International Conference on Computer Vision, 2000
- [3] Jones MJ. Rehg JM. Statistical color models with application to skin detection. Technical report, Cambridge Res. Lab., Compaq Computer Corp., 1998.
- [4]Martinkauppi B. Face colour under varying illumination-analysis and applications. PhD thesis , University of Oulu, 2002.
- [5]Terrillon JC, Shirazi M N, Fukamachi H, and Akamatsu S. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. Proc. IEEE Int'l Conf. on Face and Gesture Recognition, pp.54-61, 2000.
- [6] Comaniciu D, Ramesh V, Meer P. Real-Time Tracking of Non-Rigid Objects using Mean Shift. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'00), Hilton Head Island, South Carolina, Vol. 2, 142~149, 2000.
- [7] Comaniciu D, Ramesh V. Robust Detection and Tracking of Human Faces with an Active Camera. IEEE Int. Workshop on Visual Surveillance, Dublin, Ireland, 11~18,

2000.

- [8] Hsu RL, Abdel-Mottaleb M, A K Jain. Face detection in color images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696~706, May 2002.
- [9] Sobottka K and Pitas I. Face Localization and Feature Extraction Based on Shape and Color Information. *Proc. IEEE Int'l Conf. Image Processing*, pp. 483~486, 1996.
- [20] Kim H, Kang W, Shin J, Park S. Face Detection Using Template Matching and Ellipse Fitting. *IEICE Trans. Inf. & Syst.*, Vol.E38-D, No.11, pp2008~2011, Nov 2000.
- [2] Wang J, Tan T. A new face detection method based on shape information. *PRL*, vol. 21, pp. 463~471, 00.
- [22] Govindaraju V, Srihari SN, Sher DB. A computational model for face location. In: *Proc IEEE Conference on Computer Vision*, Osaka, Japan, 1990. 718~721.
- [23] Miao J, Yin BC, Wang KQ et.al. A hierarchical multiscale and multiangle system for human face detection in a complex back-ground using gravity-center template. *Pattern Recognition*, 1999, 32(10): 1237~1248.
- [24] Schneiderman H, Kanade T. Probabilistic modeling of local appearance and spatial relationships for object recognition. In: *Proc IEEE Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California, 1998.45~51.
- [25] Viola P. Rapid object detection using a Boosted cascade of simple features. In: *Proc IEEE Conference on Computer Vision and Pattern Recognition*, pp:511~518, 2001.
- [26] Craw I, Ellis H, Lishman J. Automatic extraction of face features. *Pattern Recognition Letters*, 1987, 5(2):183~187.
- [27] Heisele B, Serre T, Prentice S, Poggio T. Hierarchical Classification and Feature Reduction for Fast Face Detection with Support Vector Machines. *Pattern Recognition*, Vol. 36, No. 9, 2007~2017, 2003.
- [28] Heisele B, Serre T, Mukherjee S and Poggio T. Feature Reduction and Hierarchy of Classifiers for Fast Object Detection in Video Images. *Proceedings of 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2001)*, Kauai, Hawaii, Vol. 2, 18~24, December 2001.
- [29] Garcia C, Delakis M. A Neural Architecture for Fast and Robust Face Detection. *Proc. of the IEEE-IAPR International Conference on Pattern Recognition (ICPR'2002)*, Volume 2, pages 40~43.
- [30] Boris E Shpungin, Javier R. Movellan. A Multi-Threaded Approach to Real Time Face Tracking. *UCSD MP Lab TR 2000*, July 2000
- [3] Liu C.J. A Bayesian Discriminating Features Method for Face Detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 725~740, 2003.
- [32] Rowley HA. Neural Network-Based Face Detection. PhD thesis, Carnegie

Mellon Univ., 1999.

[33] Rowley H, Baluja S, Kanade T. Rotation Invariant Neural Network-Based Face Detection. Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 38~44, 1998.

[34] Chen ZY, Qi FH. Cascading Neural Networks-based human face detection. Journal of Infrared Millim. Waves, Vol. 19, No.1 February, 2000 pp: 57~61.

[35] Juell P, Marsh R. A hierarchical neural network for human face detection. Pattern Recognition, 1996, 29(5) :781~787.

[36] Kouzani AZ, He F, Sammut K. Commonsense knowledge-based face detection. In: Proc Conference on Intelligent Engineering Systems, Budapest, Hungary, 1997. 215~220.

[37] Anifantis D, Dermatas E, Kokkinakis G. A neural network method for accurate face detection on arbitrary images. In: Proc Conference on Electronics, Circuits and Systems, Pafos, Cyprus, 1999, 1:109~112.

[38] Sanderson C, Paliwal KK. Fast Features for Face Authentication under Illumination Direction Changes. Pattern Recognition Letters 24 (14) 2003.

[39] Osuna E, Girosi F. Reducing the run-time complexity of Support Vector Machines. ICPR'98, Brisbane, Australia, 16~20 Aug., 1998.

[40] Murai K and Nakamura S. Real Time Face Detection for Multimodal Speech Recognition. Proc. of ICME2002(International Conference on Multimedia and Expo), Vol.2, pp.373~376, 2002.

[4] Feraud R, Olivier J Bernier, Viallet J, and Collobert M. A Fast and Accurate Face Detector Based on Neural Networks. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 23, No. 1, JANUARY 2001.

[42] Viola P, Jones M. Robust real time object detection Technical Report. CRL 2001/01, Compaq Cambridge Research Laboratory, February 2001.

[43] Lienhart R, Kuranov A, V Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. DAGM'03, 2003.

[44] Lienhart R, Liang L, and Kuranov A. A detector tree of boosted classifier for real time object detection and tracking. IEEE International Conference on Multimedia & Expo, 2003.

[45] Li SZ, Zhu L, Zhang ZQ, Zhang HJ. Learning to Detect Multi-View Faces in Real-Time. In Proceedings of the 2nd International Conference on Development and Learning. Washington DC. June, 2002.

[46] Li SZ, Zhu L, Zhang ZQ, Blake A, Zhang HJ, Shum H. Statistical Learning of Multi-View Face Detection. In Proceedings of the 7th European Conference on Computer Vision. Copenhagen, Denmark. May, 2002.

[47] Li SZ, Zou XL, Hu YX, Zhang ZQ, Yan SC, Peng XH, Huang L, Zhang HJ.

Real-Time Multi-View Face Detection, Tracking, Pose Estimation, Alignment, and Recognition. CVPR 2001 Demo Summary. Hawaii. December, 2001

[48] Liu C, Shum HY. Kullback-Leibler Boosting. Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03). 2003.

[49] Viola P. Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade NIPS 2001: 1311-1318.

[50] Zhang ZQ, Li M, Li SZ, Zhang HJ. Multi-view face detection with floatboost. WACV02(184-188). IEEE Top Reference. 0303 Bib Ref.

[51] Sahbi H. Coarse-to-Fine Support Vector Machines for Hierarchical Face Detection. PhD Thesis, University of Versailles, April 2003.

[52] Sahbi H, Geman D, Boujemma N. Face Detection Using Coarse-to-Fine Support Vector Classifiers In the IEEE, International Conference on Image Processing, ICIP 2002.

[53] Khattab, K., et al., Embedded System Study for Real Time Boosting Based Face Detection, 32nd Annual Conference on IEEE Industrial Electronics. 2006:3461-3465.

[54] Theocharides, T., N. Vijaykrishnan and M.J. Irwin, "A Parallel Architecture for Hardware Face Detection" . ISVLSI 2006, Karlsruhe, Germany, 452-453.

[55] Hung-Chih, L., M. Savvides and C. Tsuhan. Proposed FPGA Hardware Architecture for High Frame Rate Face Detection(>100 fps)Using Feature Cascade Classifiers. First IEEE International Conference on Biometrics: Theory, Applications, and Systems, 2007.

[56] Hiromoto, M., H. Sugano and R. Miyamoto, Partially Parallel Architecture for AdaBoost-Based Detection With Haar-like Features. IEEE Transactions on Circuits and Systems for Video Technology, 2009. 19(1): 41-52.

[57] Cho, J., B. Benson and R. Kastner, Hardware Acceleration of Multi-view Face Detection, SASP'09, 2009:66-69.

[58] Yuehua, S., Z. Feng and Z. Zhong. Hardware Implementation of ADABOOST ALGORITHM and Verification. 22nd International Conference on Advanced Information Networking and Applications, 2008. pp. 343-346

[59] 施跃华, 赵峰与张忠, AdaBoost 算法的人脸检测系统的 SoC 软硬件设计. 信息技术, 2008. 32(9): 151-153.

[60] Christos Kyrkou, and Thecharis Theocharides, "A flexible parallel hardware architecture for AdaBoost-based real-time object detection," IEEE Trans. on Very Large Scale Integration Systems, vol.19, no.6, pp. 1034-1047, Jun. 2011.

[6] Yu, W., B. Xiong and C. Chareonsak. FPGA implementation of AdaBoost algorithm for detection of face biometrics. IEEE International Workshop on Biomedical Circuits and Systems, 2004.

- [62] Ming, Y., et al. Face detection for automatic exposure control in handheld camera. IEEE International Conference on Computer Vision Systems, 2006.
- [63] Hori, Y., M. Kusaka and T. Kuroda. A 0.79mm² 29mW Real-Time Face Detection IP Core. 2006 Symposium on VLSI Circuits, 2006.
- [64] Hanai, Y., et al. A versatile recognition processor employing Haar-like feature and cascaded classifier. IEEE International on Solid-State Circuits Conference, vol.52, pp.148-149, Feb. 2009.
- [65] Wei-Su, W., C. Chih-Rung and C. Ching-Te. A 100MHz hardware-efficient boost cascaded face detection design. 16th IEEE International Conference on Image Processing, 2009.
- [66] Chen, C.R., W.S. Wong and C.T. Chiu, A 0.64 mm² Real-Time Cascade Face Detection Design Based on Reduced Two-Field Extraction. IEEE Transactions on Very Large Scale Integration Systems, 2010. 99: 1-12.
- [67]居然, 上海交通大学硕士学位论文, 2008
- [68]郑峰. 基于 AdaBoost 学习算法的人脸检测方法研究[硕士论文].上海: 上海交通大学, 2005.
- [69]Carnegie Mellon Univ., Pittsburgh, PA, "CMU face datasets," 2000.[Online]. Available: http://vasc.ri.cmu.edu/idb/html/face/frontal_images/index.html
- [70] R. Frischholz, "Bao face database at the face detection homepage," 2008. [Online]. Available: <http://www.facedetection.com/downloads/BaoDataBase.zip>
- [71] Robert Frischholz, "face detection homepage",2001.[Online]. Available: <https://www.bioid.com/download-center/software/bioid-face-database.html>
- [72]Deng Cai , "Four face databases in matlab format",1997 [Online]. Available: <http://www.zjucadcg.cn/dengcai/Data/FaceData.html>
- [73] Libor Spacek, "Description of the Collection of Facial Images" ,Updated 2008 [Online]. Available: <http://cswww.essex.ac.uk/mv/allfaces/>
- [74] Lu Peng, Chen Yi song, Chen Wen guang "Cascade Based Multi-feature Fusion MethodAlgorithm for Face detection". Computer Engineering. 2011.37(2):7-9.
- [75] Zhengming Li; Lijie Xue; Fei Tan. "Face Detection in Complex Background Based on Skin Color Features and Improved AdaBoost Algorithms". PIC 2010. PP:723 - 727
- [76] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," in Proc. IEEE Int. Conf. Image Process., Sep.2002, vol. 1, pp. I-900–903.
- [77] R. Gonzalez and R.Woods, Digital Image Processing, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2002.
- [78] Junguk Cho, Bridget Benson, Shahnam Marzaei, et al, Parallelized Architecture of multiple classifiers for face detection, IEEE International Conference on ASSAP, 2009,pp72-82

- [79] Ren C.Luo, Hsin-Hung Liu. Design and Implementation of Efficient Hardware Solution Based Sub-window Architecture of Haar Classifiers for Real-time Detection of Face Biometrics. IEEE International Conference on Mechatronics and Automation. 2010.1563-1568.
- [80] Junguk Cho, Bridget Benson, Shahnam Marzaei, et al, Parallelized Architecture of multiple classifiers for face detection, IEEE International Conference on ASSAP, 2009,pp72-82
- [81] 魏良, 苏光大, 邓亚峰, 基于 FPGA 的快速人脸检测, 电子技术应用, 2006 年第 11 期, 33-35.
- [82] B. Heiselet, T Serre, M Pontil, et al., Component-based Face Detection, Proc IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001, vol:1, 657-662.
- [83] T. Theocharides, G. Link, N. Vijaykrishnan, M. J. Irwin, and W.Wolf,“Embedded hardware face detection,” in Proc. IEEE Int. Conf. VLSIDes., 2004, pp. 133–138.
- [84] 谢长生,徐睿.FPGA 在 ASIC 设计流程中的应用, 微电子技术, VOL29, 第 6 期, 2001.12.
- [85] V. Kianzad etal., “An architectural level design methodology for embedded face detection”, International Conference on Hardware/Software Codesign and System Synthesis, Sep. 2005.
- [86] Mariatos, V., K.D. Adaos and G.P. Alexiou. “Design and implementation of a reconfigurable embedded real-time face detection system. in Rapid System Prototyping, 2007. RSP 2007. 18th IEEE/IFIP International Workshop on. 2007.

攻读博士学位期间发表的论文

编号	作者排序	论文名称	期刊会议	检索
1	—	A High Speed Reconfigurable Face Detection Architecture Based on AdaBoost Cascade Algorithm	IEICE TRANSACTIONS on Information and Systems	SCI
2	—	A high speed reconfigurable face detection architecture	IEEE 9th International Conference on ASIC (ASICON 2011)	EI
3	—	The Design and Realization of a Divider with Adjustable Precision in Fingerprint Enhancement	Asia-Pacific Signal and Information Processing Association Annual Summit and Conference 2011 (APSIPA ASC 2011)	EI
4	—	Fingerprint verification based on wavelet and edge detection	IEEE 9th International Conference on Electronic Measurement and Instruments (ICEMI 2009)	EI

致谢

首先，我要衷心感谢我的导师曾晓洋教授。在我攻读博士学位的四年时间里，曾晓洋教授对我的学习、工作、生活等各方面给予了悉心的指导和关怀，他渊博的知识、严谨的治学态度，深邃的洞察力和求实创新的精神给我留下了非常深刻的印象。能够在曾晓洋教授的指导下完成博士学位的工作，我感到非常的荣幸和自豪。

在复旦大学专用集成电路与系统国家重点实验室学习的四年时间里，我还有幸认识了一批杰出的老师，并能够在他(她)们的指导下学习，工作，使我获益匪浅。他(她)们是：韩军老师、虞志益副研究员、陈赟老师、陈旭老师、荆明娥老师、范益波老师、顾韵老师、张国权老师、……

在攻读博士学位期间，我还有幸认识了一批优秀的学长、学弟和学妹，并与他们一同学习、工作和科研。戴麟、邹尧、吴川、鲍丹、张章、陈淑玉、陈秀平、向波、黄伟、卢仕听、陈舟、阎伟伟、石泽文、黄贝、英彦、肖瑞瑾、王帅、李阳、……也许无法将名字一一列出，但我会永远记住这段一起度过的美好时光。

最后，要深深感谢我的家人，默默支持我的公公、在生活上对我无微不至的婆婆、无时无刻不在挂念关心我的爸爸妈妈、鼓励我帮助我的老公、还有陪伴我度过最后的博士生涯的豆豆宝贝，你们都给予了我莫大的动力和勇气，使我能够最终完成博士学业。谢谢你们！

谨以此文献给所有关心、帮助和支持过我的人！

论文独创性声明

本论文是我个人在导师指导下进行的研究工作及取得的研究成果。论文中除了特别加以标注和致谢的地方外,不包含其他人或其它机构已经发表或撰写过的研究成果。其他同志对本研究的启发和所做的贡献均已在论文中作了明确的声明并表示了谢意。

作者签名: 周薇娜 日期: 2012.6.3

论文使用授权声明

本人完全了解复旦大学有关保留、使用学位论文的规定,即:学校有权保留送交论文的复印件,允许论文被查阅和借阅;学校可以公布论文的全部或部分内容,可以采用影印、缩印或其它复制手段保存论文。保密的论文在解密后遵守此规定。

作者签名: 周薇娜 导师签名: 李 日期: _____