

西南交通大学

毕业设计

人脸识别算法分析比较及
人脸识别系统实现

年 级: 2011 级

学 号: 20112782

姓 名: 宋 超

专 业: 软件工程

指导老师: 王 献

二零一伍年六月

西南交通大学本科毕业设计

院 系 信息科学与技术 专 业 软件工程
年 级 2011 级 姓 名 宋 超

题 目 人脸识别算法分析比较及人脸识别系统实现

指导教师

评 语

指导教师 (签章)

评 阅 人

评 语

评 阅 人 (签章)

成 绩

答辩委员会主任 (签章)

年 月 日

毕业设计任务书

班 级 2011 级软件 4 班 学生姓名 宋 超 学 号 20112782

发题日期: 2015 年 1 月 1 日 完成日期: 2015 年 6 月 15 日

题 目 人脸识别算法分析比较及人脸识别系统实现

1、本论文的目的、意义

人脸识别是一种重要的身份鉴别技术, 具有广泛的应用前景。相比于同指纹和虹膜等生物特征识别技术, 人脸识别具有操作简易、隐蔽性较好、结果直观等优点。本设计的任务是对人脸识别的已有算法进行归纳总结; 在此基础上, 基于某种已有算法实现一人脸识别系统。具体任务如下:

- 1) 阅读相关的文献, 归纳总结已有的主要人脸识别算法;
- 2) 基于某种已有算法, 在 Windows、Linux、Android 或 iOS 平台编程实现一人脸识别系统, 包括:
 - 从选择的图片中检测出人脸;
 - 通过摄像头拍照检测人脸;
 - 单人图像间的人脸比对, 测量出人脸相似度。

2、学生应完成的任务

(1) 人脸识别系统;

(2) 毕业设计论文。

3、论文各部分内容及时间分配: (共 17 周)

第一部分 阅读文献, 归纳总结已有算法 (4 周)

第二部分 选定某种算法进行深入学习 (3 周)

第三部分 基于选定的算法, 实现人脸识别系统 (7 周)

第四部分 撰写论文 (2 周)

评阅及答辩 (1 周)

备 注 _____

指导教师: _____ 年 月 日

审 批 人: _____ 年 月 日

摘 要

随着模式识别的技术进步和人工智能领域技术的日益升温，占据极为重要位置的人脸识别课题正成为炙手可热的研究热点。利用准确而又稳定的人脸识别算法以构建自动人脸识别系统，通过计算机技术实现对面脸图像的自动识别在如今拥有广阔的应用领域与应用前景，诸如身份鉴定、身份确认、视频安全监视、人脸图像数据压缩等方面都有巨大的需求。

本文的目的在于探索人脸识别课题，重点有基于肤色检测的人脸识别算法的研究和人脸识别系统的实现两大部分，其中后者是对前者的应用与扩展。

对基于肤色检测的人脸识别算法进行的研究，内容包括介绍算法理论基础和结合已有方法和自己的理解完成各阶段的具体实现。详细解析了其颜色空间转换、肤色相似度矩阵计算、相似度矩阵邻域滤波、相似度灰度图二值化处理、基于边界检测的区域定位和基于直方图对比的脸部区域对比六个主要步骤。

本文的人脸识别系统的实现采用选定基于肤色检测的算法和 EmguCV 的基于 HaarCasCade 分类器的 Adaboost 方法进行开发，实现了四个核心的功能：基于肤色检测的人脸检测与定位、基于 Adaboost 算法的人脸检测与定位、两种算法的摄像头人脸检测与定位、基于巴氏距离直方图比对的人脸图像比对。最终识别系统的人脸检测定位、人脸比对的准确度都较好，取得了预期的效果。

关键词：人脸识别； 模式识别； 肤色检测； 人脸定位； Adaboost； 直方图比对

Abstract

With the advances of pattern recognition technology and the promotion of artificial intelligence technology, face recognition is becoming a hot topic of research focus, which occupies a very important position. Making use of accurate and stable face recognition algorithms to build face recognition systems so as to achieve the automatic recognition of face images with computer technology has a wide range of application and promising prospects, such as the identification of identity, identity confirmation, video security surveillance, face image data compression, and so on, which have huge demand on face recognition.

This thesis explores the subject of face recognition, with a focus on two major parts: the research of the skin detection based face recognition algorithm and development of a face recognition system, among which the latter serves as an application and expansion of the former. As for the research of the skin detection based algorithm, this thesis introduces the theoretical fundamentals of this algorithm and completes the specific realization of various stages with the help of existing methods and combined with the author's own comprehension. This thesis analyzes in detail the six main stages of this algorithm, i.e., the color space conversion, calculation of skin similarity matrix, similarity matrix neighborhood filtering, similarity gray image binarization processing, edge detection based on regional location, and face region histogram-based contrast.

Regarding the realization of the face recognition system, this thesis relies on the skin detection based recognition algorithm and the EmguCV's HaarCasCase classifier based Adaboost method to develop the system. In this process, this thesis designs four core functions, namely, face detection and location from images based on skin detection, face detection and location from images based on the Adaboost algorithm, face detection and location from a camera using two kinds of implemented algorithms, face image comparison based on Bhattacharyya distance histogram. Finally, the face detection and location as well as the face comparison of the recognition system have reasonable accuracy so that the expected performance has been achieved.

Keywords: face recognition; pattern recognition; skin detection; face location; Adaboost; histogram comparison

目 录

摘 要.....	III
ABSTRACT.....	IV
第 1 章 绪 论.....	1
1.1 课题背景和意义.....	1
1.2 国内外发展应用现状.....	2
1.3 本论文所做工作及思路.....	3
1.4 章节安排.....	3
第 2 章 图像处理基础与人脸识别算法归纳.....	4
2.1 图像文件结构.....	4
2.1.1 图像基本类型.....	4
2.1.2 文件结构.....	4
2.1.3 课题实现用图像载体类.....	5
2.2 图像识别步骤.....	6
2.2.1 图像预处理.....	6
2.2.2 图像分割.....	8
2.2.3 图像特征提取.....	8
2.2.4 图像分类.....	11
2.3 人脸识别算法归纳.....	11
第 3 章 基于肤色检测的人脸识别算法.....	14
3.1 RGB 到 YCbCr 颜色空间变换.....	14
3.2 肤色模型建立.....	15
3.3 肤色相似度矩阵计算.....	17
3.4 相似度矩阵的中值邻域滤波.....	18
3.5 二值化处理.....	20
3.6 人脸区域定位.....	22
3.7 人脸区域比对.....	25
第 4 章 人脸识别系统实现.....	27
4.1 系统实现方案.....	27
4.1.1 软件环境.....	28
4.1.2 硬件环境.....	28
4.2 系统的详细设计.....	29
4.2.1 系统功能设计.....	29

4.2.2 系统模块结构.....	34
4.3 系统各模块实现.....	35
4.3.1 系统界面模块.....	35
4.3.2 基于肤色检测的人脸识别模块.....	38
4.3.3 Adaboost 算法人脸检测模块.....	47
4.3.3 摄像头人脸检测模块.....	48
4.3.5 人脸比对模块.....	50
4.4 系统实现总结.....	52
结 论.....	53
致 谢.....	54
参考文献.....	55

第 1 章 绪 论

1.1 课题背景和意义

在计算机视觉的研究领域中，从图像中识别出感兴趣的目标并利用计算机进行解析是计算机视觉研究的一个需求日益增长的方向。而人脸识别作为能够确定人身份的便捷技术应运而生，它的应用范围随着技术适用性变广，安全性随着准确率提高，是如今备受关注的研究课题。

人脸识别课题的意义主要有两方面。一是推进对人类视觉系统本身的认识。随着应用生物学、心理学、数学和计算机科学等多学科应用于人脸识别研究^[1]，人们对人类的视觉认知方式和大脑记忆处理模型进行深入研究，从而有了更深入的认识。

二是将人脸识别应用于科技前线的人工智能领域。采用准确稳定的人脸识别算法，建立人脸自动识别系统，利用计算机技术实现对面脸图像或者视频的识别拥有着广阔的应用领域与应用前景^[1]，诸如以下不等方面都有巨大的需求：身份鉴定（一对多的搜索），根据已知面部数据搜索人脸数据库，确认相似度递减的身份列表或直接返回鉴定结果；身份确认（一对一的比对），将实时面纹数据与存储模型比对，确认身份；监视，应用面像捕捉，在监控范围中发现人脸，跟踪一个人和确定他的位置；人脸图像数据压缩，将面部数据压缩成小字节文件，以便智能卡、条形码或其他存储空间有限的设备中使用。

未来准确完整的人脸识别算法的设计将改变人们的生活和工作方式，应用到有准确识别需求的各行各业，重新定义“便捷与安全”，本课题研究只为探索学习这一愿景。

1.2 国内外发展应用现状

人脸识别研究早就始于 20 世纪 60 年代末, 90 年代成为科研热点, 近 20 年计算机人脸识别技术取得了很大的进步, 国内外论文数量大幅增长, 国际会议中关于人脸识别的专题也屡屡可见并引起了重视^[2]。人脸识别正成为炙手可热的技术热点和需求增长的研究领域, 吸引了众多的研究者。国内外高校和科研机构对人脸图像面部识别的研究都越加活跃。美国德国日本的很多大学都在进行这方面的研究, 并且能够得到大公司或军方的支持, 取得了许多实质性的技术突破。国内的院校(如南京理工、清华、浙大和电大等)和研究机构(中科院自动化研究所)也在从事自动识别技术的研究, 并在国际种常用方法的基础上做了发展性工作^[3]。

人脸识别系统的核心就是识别算法, 目前人脸识别算法的研究主要有两个方向: 其一是基于整体的研究方法, 考量人脸整体属性, 包括特征脸方法^[4]、神经网络^[5]、等密度分析匹配^[6]、隐马尔可夫模型^[7]、弹性图匹配^[8]和 SVD 分解^[9]。优点是保留了更多信息, 缺点是光照、视角和人脸尺寸对识别产生很大影响; 其二是基于特征分析的方法^[10], 将人脸基准点的相对比率和描述脸部特征参数构成识别特征向量。优点是同时保留人脸部件拓扑关系和部件本身信息, 设计识别算法为提取局部轮廓信息及灰度信息, 缺点在于其难点, 如何建立好的模型来表达。鉴于两者间长短处, 近年来出现将两者结合研究方法, 如 Kin Man Lam 的基于分析和整体的方法, Andreas Lanitis 提出的可变形模型对人脸解释和编码的方法^[10]。

现在人脸识别算法研究的难点主要包括: 人脸的相似性, 人类不同个体之间五官与面纹的结构和外形具有相似性, 尤其体现在有血缘关系的个体间; 易变性或者说干扰性, 人脸图像很不稳定, 年龄变化、表情、拍摄角度、光照和外物干扰等都会对算法产生偏差性或误导性的影响。如何克服这些难点达到识别高准确率正是人脸识别算法的研究重点, 决定着改进现有算法还是创造新算法两种方向。

1.3 本论文所做工作及思路

查阅图像模式识别的资料，归纳学习对数字图像的基础处理和模式识别，为识别方法研究和系统的实现作好铺垫；

查阅人脸识别领域的文献资料，对人脸识别算法进行归纳，总结现有的各类方法的实现算法及其优缺点，为以后对这些方法的深入学习作准备；

通过书籍学习图像模式识别和机器学习的知识，对基于肤色检测的人脸识别算法进行深入研究，分步骤进行剖析，研究算法各步骤的理论基础和实现方法；

选定基于肤色检测的算法和 EmguCV 框架的基于 HaarCasCade 分类器的 Adaboost 方法，在.NET 平台上 Blend +Visual Studio 开发环境下，采用 WPF 框架开发人脸识别系统，识别系统需要实现的四大核心功能目标：基于肤色检测的人脸检测与定位、基于 Adaboost 算法的人脸检测与定位、两种算法的摄像头人脸检测与定位、基于巴氏距离直方图比对的人脸图像比对。

1.4 章节安排

第一章，绪论：叙述人脸识别课题的背景与意义，课题的国内外发展应用状况以及论文的内容结构安排；

第二章，图像处理基础与人脸识别算法归纳：介绍计算机图像处理的理论基础，包含图像存储结构和图像模式识别的基本步骤；对现有的几大类人脸识别方法进行简要归纳，总结各类下的实现方法及其优缺点；

第三章，基于肤色检测的人脸识别算法研究：详细介绍基于肤色检测的人脸识别算法，分步骤对算法各步骤的理论基础和实现方法进行解析；

第四章，人脸识别系统的实现：详细介绍识别系统的功能和模块设计，以及各模块的具体实现。

第 2 章 图像处理基础与人脸识别算法归纳

2.1 图像文件结构

2.1.1 图像基本类型

- a) 二值图像。1 位像素值由黑“0”白“1”构成；
- b) 16 色图像。4 位像素值表示调色板中 16 种颜色索引值；
- c) 256 色图像。8 位像素值表示调色板中 256 种颜色索引值；
- d) 真彩色图像。像素值 3 字节组成，表示 R、G、B 三原色的亮度值，无调色板。

2.1.2 文件结构

一般图像文件结构：

- a) 文件头：图幅参数、类型参数、像素数据和调色板数据的位置参数、压缩类型参数等；
- b) 像素数据。以位图的形式存放；
- c) 调色板数据。

Windows 3.0 以后的 BMP 位图文件格式与显示设备无关，因此称为设备无关位图（Device-independent Bitmap,DIB）格式^[18]，本文算法以 DIB 图片为研究载体。

位图文件头(Bitmap-file Header)	
位图信息	位图信息头 (information Header)
	调色板数据 (Palette Data)
像素数据(Image Data)	

图 2-1 图像文件结构

2.1.3 课题实现用图像载体类

本论文的人脸识别应用系统实现部分因为在 .Net 平台上开发，所以图像像素的操作使用载体类 `Bitmap` 类和 `BitmapData` 类。本小节介绍这两个类的重要属性和方法。

算法处理用位图 `Bitmap` 类：

- 1) `GetPixel` 方法和 `SetPixel` 方法：指定位图中像素，获取或设置其颜色；
- 2) `PixelFormat` 属性：操作像素前需知道其格式，此属性即图像的像素格式，如 `Format8bppIndexed` 灰度图；
- 3) `Palette` 属性：获取和设置图像所使用的颜色调色板；
- 4) `Height Width` 属性：返回图像的高宽；
- 5) `LockBits` 方法和 `UnlockBits` 方法：锁定和解锁系统内存中的位图像素，在基于像素点的图像处理方法中使用 `LockBits` 和 `UnlockBits` 是一个很好的方式，这两种方法可以使我们指定像素的范围来控制位图的任意一部分^[1]，从而消除了通过循环对位图的像素逐个进行处理，每调用 `LockBits` 之后都应该调用一次 `UnlockBits`。

位图像素数据 `BitmapData` 类：

- 1) `Height、Width` 属性：被锁定位图的高宽；
- 2) `PixelFormat` 属性：数据的实际像素格式；
- 3) `Scan0` 属性：被锁定位图像素数据的数组的首字节地址，如果整个图像被锁定，则是图像的第一个字节地址；
- 4) `Stride` 属性：位图步幅，也就是位图的扫描宽度，一行的字节宽度，与图像宽度不同。

2.2 图像识别步骤

图像识别为模式识别的其中一类，如图 2-2 图像基本识别步骤所示的处理总流程在人脸识别中也有应用。

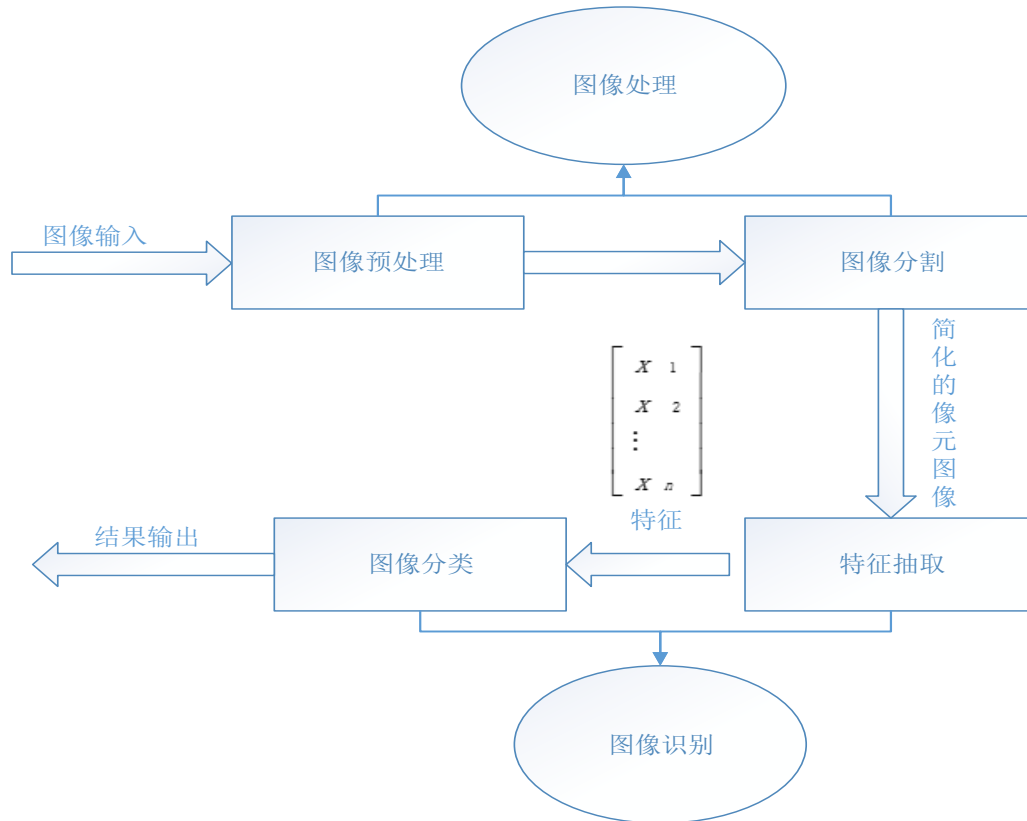


图 2-2 图像基本识别步骤

2.2.1 图像预处理

研究图像内容的识别，首先要结合知识和经验对待识别物体的特点概括，建立可行的操作流程方案，为之后的处理作铺垫；然后要对输入的图像进行预处理，去除各种格式不统一和像素信息对识别进行的影响。包括过滤干扰像素、降低图像噪声，当图像中的信息弱化或者较少时还需要对图像进行增强、几何调整、颜色校正等处理，以这些预处理操作满足进一步的分析需求。

此段介绍图像空域增强方法。空域指图像的空间信息，即反映出图像中目标物体的大小、形状、位置等可通过一定物理模式描述的特征。例如物体的边缘轮廓因灰度值变化波动较大而产生高频率特征，而物体的内部像素灰度值较为均一，呈现

低频率特征。因此可以通过增强高频和低频特征来对图像分别达到边缘锐化和内部平滑处理，达到图像降噪目的。空域增强应用模版卷积方法对每一像素的领域进行处理，一般分为非线性和线性两类^[18]，两个主要步骤如下：

- 1) 将设定的模版在图像中按一定次序遍历，计算时每个像素点与模版中心重合（边缘像素除外）；
- 2) 模版与像素值卷积方法计算，并赋值到像素。

图 2-3 为应用模版进行滤波的示意图。a 是图像的一部分包含 9 个像素， $P_i(i=1,2,\dots,9)$ 表示像素灰度值。b 是 3x3 参数模版， $T_i(i=1,2,\dots,9)$ 表示模版系数，模版大小一般取奇数。

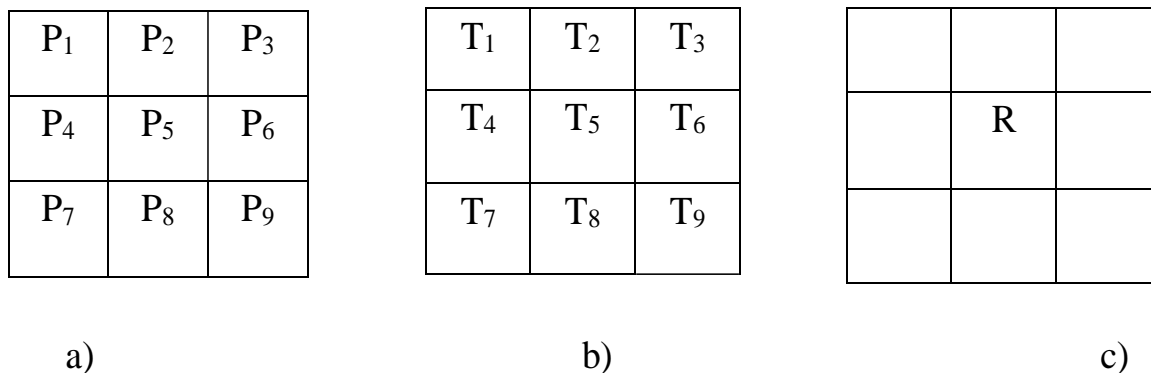


图 2- 3 空域增强滤波模版示意图

开始滤波时模版在图像中遍历，将 T_5 与 P_5 重合，进行公式 2-1 的卷积方法计算，并将结果赋予 P_5 像素对应的灰度值 R 。

$$R = \sum_{i=1}^9 T_i P_i = T_1 P_1 + T_2 P_2 + \dots + T_9 P_9 \quad (2-1)$$

对图像中的每个像素进行以上计算，最终得到经过滤波的增强图像的所有像素的灰度值^[12]。

在本论文采用的基于肤色模型检测的人脸识别算法中，图像的预处理操作包括：

- 1) 统一图像信息载体格式 DIB，利用输入图像初始化 Bitmap 实例，以进行后续操作；

- 2) 将图像像素由 RGB 颜色空间映射到 YCbCr 颜色空间;
- 3) 利用肤色特征模型对位图所有像素进行肤色相似度计算, 并进行归一化处理, 此步骤属于人脸的特征提取;
- 4) 对归一化后的相似度矩阵进行中值滤波, 目的是抑制图像噪声, 使图像平滑;
- 5) 将像素对应的相似度矩阵变换到 $[0, 255]$ 上, 以形成灰度图像像素矩阵;
- 6) 为了清晰的划分人脸和非人脸区域, 利用设定阈值对相似度灰度图进行二值化处理。

2.2.2 图像分割

从已经处理过的图像中找到需要识别的物体, 还需要对图像进行分割, 也就是定位和分离出不同的待识别物体, 这一过程中输入的是整幅图像, 输出的是元图像或者对待识别物体在图中的位置标识。

目前为止已经提出了很多基于不同模型按照不同特性的图像分割的方法, 由于需要识别和分割的物体特性不一致, 这些方法都有一定的优缺点和适用范围并且较难得到完美的分割结果。图像分割的方法大致可以分为直方图阈值分割法、基于边缘检测法和基于区域增长法^[6]。

直方图阈值一般先做出具有双峰状的图像灰度级直方图, 利用双高极点之间低极点灰度值作为分割图像的阈值, 最终得到二值化的分割结果; 基于边缘检测需要先检测出局部特性的不连续性再连成边界, 然后通过这些边界把图像分成不同区域; 区域增长法由新区域点出发检测像素连通区, 将所有像素分割成不同区域。

2.2.3 图像特征提取

特征提取是图像识别中至关重要的一步, 是图像分析算法的起点, 图像识别的物体在真实世界对应物理模式空间, 而图像模式识别即是从模式空间经过特征空间映射到类别空间。图像特征的提取需要专家系统总结出待识别物体的特性, 这往往需

要结合知识、经验和观测分析数据，对应图像中识别目标的特征，完成特征模型的建立，以供提取图像中包含重要识别信息的特征像素使用。提取的结果是将图像上的像素点分为不同的子集，这些子集往往属于孤立的像素点、连续的曲线或连续的区域^[13]。

以下对常用的图像特征进行简要介绍，有以下几类：

1) 颜色特征

基于像素点的全局特征，描述了图像中目标物体的表面性质。颜色特征对于图像区域的形状、方位等变化不敏感，所以颜色特征能够忽略这些变化的影响直接识别出物体，但是其缺陷是不能很好地捕捉到目标物体的局部特征。

介绍常用的提取和匹配方法：

- i. 颜色直方图：描述不同色彩在整幅图像中占比。典型的匹配方法有直方图相较法^[1]、距离法、中心距法、参考颜色表法、累加颜色直方图法；
- ii. 颜色集：近似直方图，将颜色空间量化成若干个柄。匹配时比较不同颜色集之间的距离和色彩空间关系；
- iii. 颜色矩：使用低阶矩表示图像中颜色分布；
- iv. 颜色聚合向量；
- v. 颜色相关图。

2) 纹理特征

同颜色特征是描述图像区域目标物体的表面性质的全局特征，但是它需要在包含多个像素点的区域中进行统计计算。具有旋转不变性和对噪声的抗干扰性。缺点是计算偏差受限于图像分辨率。

常用的提取和匹配方法：

- i. 统计方法：如灰度共生矩阵分析方法和图像自相关函数纹理提取方法；
- ii. 几何法：建立在基本纹理元素基础，利用复杂纹理的基元排列方式分析。分析算法有 Voronio 棋盘格特征法和结构法；
- iii. 模型法：以图像的构造模型为基础，如 Gibbs 随机模型法、马尔科夫模型法(Markov)；
- iv. 信号处理法：如小波变换、自回归纹理模型、Tamura 纹理特征法、灰度共生矩阵。

3) 形状特征

利用待识别目标的通用形状特征进行检测，优点是建立模型相对较为方便，缺点是当目标会发生形变时，方法往往得不到满意结果。

形状特征的表示方法一般分两大类基于轮廓特征表示的和基于区域形状特征表示的方法，常用提取和匹配方法，：

- i. 边界特征法：通过检测目标物体的边界特征描述来获取形状参数。如基于 Hough 变换的检测方法、边界方向直方图法^[14]；
- ii. 傅立叶形状描述符法：使用目标的边界傅立叶变换作为形状描述，利用其周期性和封闭性将二维问题降低为一维。
- iii. 几何参数法：如数学模型中有关形状的定量测度（如面积、周长、矩等）的形状参数法。
- iv. 形状不变矩法：利用目标所占区域的矩形作为描述参数
- v. 基于小波和相对矩的方法。

4) 空间关系特征

空间关系指图像中分割出的多个目标之间的相互空间位置或相对方向关系^[15]。通常分为相对和绝对空间位置信息两类。空间关系特征相对于其它特征优势在于可加强对图像内容的描述区分能力，但缺陷是图像或目标的旋转、反转、尺度等变化对它的影响交大，并且在实际应用中单独利用空间信息不能有效准确地表达场景信息，所以检索时，还需要其它特征来配合。

空间关系特征的提取通常方法有两种方类：一类是对图像进行自动分割，划分出图像中包含目标的对象或颜色区域，根据这些区域提取出图像特征，并建立索引；另一类方法则简易地将图像均匀划分为若干规则子块，然后对每个图像子块提取特征，并建立索引^[16]。

2.2.4 图像分类

将图像从特征空间映射到类别空间。利用人类认知系统的一些特定的知识和经验确定待识别目标的分类原则，即判别规则，选定适当的判别规则，用于从图像中提取的特征值上，使用模式识别方法进行分类，将特征空间的样本分成不同的类型，以便对图像的重要目标信息得到对应的理解和解释，即达到图像模式识别的最终目的。

模式识别的方法主要分为三大类：决策理论方法、结构方法和统计模式识别。其中统计模式识别的主要方法有：非线性映射法、特征分析法、判别函数法、近邻分类法、主因子分析法等。在统计模式识别中，贝叶斯决策规则从理论上解决了最优分类器的设计问题，但其实施却必须首先解决更困难的概率密度估计问题^[17]。BP神经网络直接从观测数据学习训练样本，简便并且效率高，因此获得了广泛的应用，但它是一种启发式技术，缺乏指定工程实践的坚实理论基础^[18]。统计推断理论研究所取得的突破性成果导致现代统计学习理论——VC 理论的建立，该理论不仅在数学基础上圆满地回答了人工神经网络中出现的理论问题，而且导出了 SVM（支持向量机）方法^[18]。

2.3 人脸识别算法归纳

本论文中的人脸识别指的是从图像或视频中识别出人脸，包含了人脸检测、人脸定位、人脸比对分析等问题。此处作为说明。以下对现有的识别方法进行简要归纳。

（一）基于知识的方法

此类方法重点在于将人脸识别问题转化为假设检验问题，利用人类的知识建立检测的规则。典型的是马赛克方法，它使用人脸区域的灰度分布作为规则，将图像从低分辨率至高分辨率进行筛选，以样本满足这些规则的程度作为检测的判据^[15]。

此方法的难度在于如何将人所具有的知识转化为检测用的规则，若规则过于严格，则由于样本和机器的限制会出现由于不能满足所有条件而识别不到人脸，若规则简单则会出现错误的识别结果。并且此法无法适应人脸姿态的变化。

(二) 基于特征不变性的方法

这类方法从生活中可以发现规律，人总是能够在普遍环境下发现不同姿态的人脸，大脑的处理必然是基于某些不变的特性。这类基于不变性特征的方法包括基于皮肤颜色的算法、基于脸部纹理特征的检测算法、基于五官几何分布的方法^[18]，并且可以有将多种特征结合的方法。这类方法的缺点在于这些不变性特征只是基于检测的目标——人不变，但检测的载体——图像会受照明、噪声和遮挡情况影响。

(三) 基于模版匹配的方法

这类方法利用人的脸部特征规律人工定义或者参数化地建立函数模型^[19]来描述标准的人脸模式，然后根据输入图像的人脸和定义的模版的相关性来进行检测。这类方法可分为预定义模版法和可变形模版法（弹性模板匹配法）两类。此类方法的优势在于函数化模型实现相对容易；缺点在于对形状变化、比例变化、姿态变化的自适应不是很理想，为对付这些缺点，最近提出了多分辨率模版、多比例模版、子模版等方法^[15]。

(四) 基于外观的方法

此类方法主要基于统计分析和机器学习理论，将人脸图像视为一个高维向量把问题转化为高维空间中分布信号的检测问题，通过样本图像的学习来获得或调整设定判别依据进行人脸的识别。此类方法主要包括以下几种：

1) 基于 K-L 变换的特征脸（Eigenfaces）方法：

利用特征脸空间的补空间进行人脸检测，检测的统计量为待检测区域到特征脸子空间的距离，距离越小，待检测区域越像人脸。人脸比对时使用特征脸主元子空间。

2) 子空间方法:

因描述性强、计算代价小而简便易行,但由于没有利用反例样本信息,因此对类似人脸的物体辨别能力不足。另外局部保持投影(Locality Preserving Projections, LPP)是近期兴起的一种子空间分析法^[20],解决了 PCA 等传统线性方法难以保持原始数据非线性流形,非线性方法难以获得新样本点低维投影的缺点。

3) 人工神经网络方法:

人脸检测可以看成是只有人脸样本和非人脸样本的两类模式识别问题,通过对它们的学习可以产生分类器。目前已经提出了多种神经网络模型用于人脸检测并取得了可观的成功。缺点在于通过经验来确定的神经网络中结点数、网络层选择及学习速度的调整等因素对结果有很大影响。

4) 主成份分析(PCA)方法:

PCA 模式识别领域一种重要的方法,已被广泛地应用于人脸识别算法中,这类方法在人脸识别系统的应用的难点是增量学习^[19]。

5) Adaboost 学习方法:

适应性的 Boosting 算法,基于级联分类模型的分器,该方法根据弱学习的结果反馈适应地调整假设的错误率,不需要预先知道假设的错误率下限,可以很好地利用反例样本,是目前的一个研究趋势。在本论文的实现系统上其中一个模块会利用 EmguCv 的基于 Haar 特征的 Adaboost 算法进行人脸识别。

第 3 章 基于肤色检测的人脸识别算法

本论文选用基于肤色检测的人脸识别算法进行人脸识别系统的研究与应用，最终结果是从图像或视频中识别出人脸。它的理解和实现相对容易，并且准确率也较理想。研究表明，不同种族、年龄、性别的人脸肤色尽管看上不一样，但是这种差异主要集中在亮度上^[18]，在去除亮度的色度空间中，不同的人脸肤色分布具有聚类性，并且对肤色表情和姿态的变化不敏感，算法稳定性较好。本节详细研究此种算法，识别方法的建立包括颜色空间变换、肤色模型建立、肤色相似度计算、中值滤波降噪、二值化处理和区域定位六个步骤。

3.1 RGB 到 YCbCr 颜色空间变换

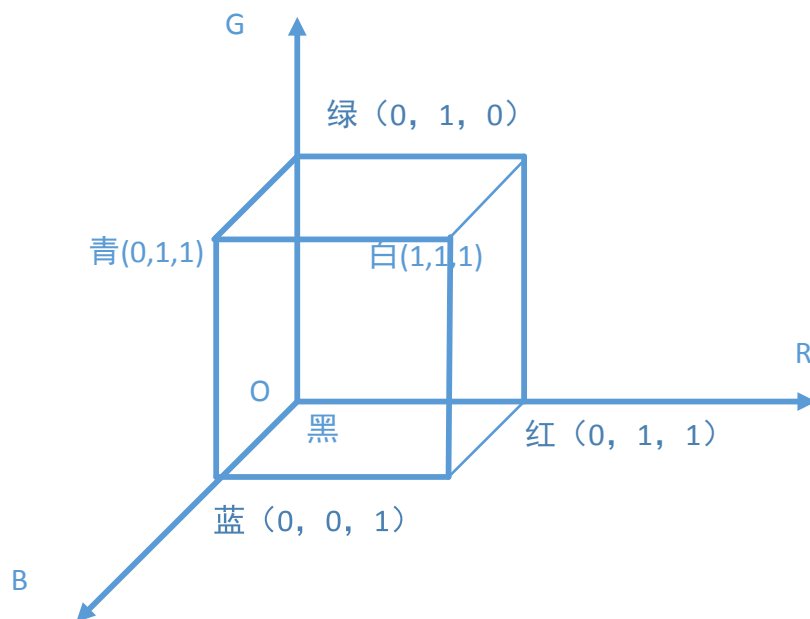


图 3- 1 RGB 颜色空间模型

如图 3- 1 RGB 颜色空间模型所示，为最典型且常用的颜色空间，契合人的视觉系统，在计算机系统的图像表示中 RGB 的数值为[0~255]，在 24 位二进制格式表示每个像素的 bitmap 中，颜色分量顺序为 B、G、R，每个分量占一个字节。大部分图像捕捉设备都采用这种模式，而 RGB 空间中色度信息和亮度信息是混合在一

起的，为了利用肤色在色度空间的据类型，需要把颜色表达中的色度信息和亮度信息分开，因此需要进行颜色空间的转换。

YCbCr 颜色空间是 YUY 颜色空间的缩放和便宜，Y 表示亮度，Cb、Cr 分别表示蓝色分量和红色分量，存储空间为 8 为无符号整型数据空间，YCbCr 进行了图像子采样，是视频图像和数字图像中常用的色彩空间。

人脸识别领域的实验结果表明，人类的肤色在 YCbCr 颜色空间相对比较集中，即肤色的聚类特性在此空间中比较好，有利于基于肤色的人脸识别处理，所以在进行图像人脸识别之前，利用公式 3-1 将图像从 RGB 颜色空间转换到 YCbCr 颜色空间。

$$\begin{aligned} Y &= 0.2990R + 0.5870G + 0.1140B \\ Cb &= 128 - 0.1687R - 0.3313G + 0.5000B \\ Cr &= 128 + 0.5000R - 0.4187G - 0.0813B \end{aligned} \quad (3-1)$$

式中，R、G、B 是 RGB 颜色空间中红、绿、蓝 3 中颜色分量的颜色值。

YCbCr 颜色空间中，Y 的范围为[16~235]，Cb 和 Cr 的范围为[16~240]，但由于 Y 和 Cb、Cr 可能会因视频处理和噪声偶然超出范围，由公式 3-1 可知此时 RGB 的值可能会偏移到[0~15]和[236~255]范围内，因此为使用方便，后期使用公式 3-2 进行空间映射。

$$\begin{aligned} Y &= 0.257R + 0.504G + 0.098B + 16 \\ Cb &= 128 - 0.148R - 0.291G + 0.439B \\ Cr &= 128 + 0.439R - 0.368G - 0.071B \end{aligned} \quad (3-2)$$

3.2 肤色模型建立

色度信息在众多彩色图像分割的应用中是相当有效的，同样非常适合于将皮肤区域从图像背景中分割而出。即使不同年龄、种族、性别的人的肤色的颜色相差交大，但在色度上往往是很相近的，并且聚集在一个较小区域。在去除亮度差异后，肤色具有很高的聚类性。

在 RGB 颜色空间中为去除亮度,人脸肤色需要进行归一化处理,通常使用 RGB 颜色分量和进行归一化,公式如 3-3 所示。

$$b = B/(R + G + B), g = G/(R + G + B), r = R/(R + G + B) \quad (3-3)$$

式中 b、g、r 分别表示归一化后分量。

但是这种归一化仅仅是去除了 R、G、B 的相对亮度成分 R+G+B,其分量本身还存在亮度信息并没有去除,所以这种方法用于肤色检测效果不好。

而在 YCbCr 颜色空间中,图像像素的亮度直接由 Y 分量表示。颜色空间归一化色度直方图后,实验结果表明,将肤色的范围定为 $110 \ll C_b \ll 125, 140 \ll C_r \ll 170$ 区间内的肤色提取效果较好,以下是肤色概率模型的表述。

$$\begin{aligned} m &= E(x) \\ x &= (C_b, C_r)^T \\ C &= E[(x-m)^2] \end{aligned} \quad (3-4)$$

式中 m 为 Cb、Cr 分量转置矩阵的均值, C 为协方差矩阵。

人脸的肤色符合高斯分布,满足二维高斯模型:

$$M = (m, C) \quad (3-5)$$

像素检测时使用肤色概率模型 3-6,通过该肤色模型检测任意一个像素是否为人脸皮肤的概率 $P(C_b, C_r)$ 。

$$P(C_b, C_r) = \exp[-0.5(x - m)^T C^{-1}(x - m)] \quad (3-6)$$

该肤色模型是之前的人脸识别领域的研究结果,根据大量样本在 YCbCr 颜色空间的色度分布的统计数据建立的,已经证明该概率模型能够较为理想地区分出人脸与非人脸区域,并且具有一般性。利用该模型进行相似度矩阵的计算,从而导出基于肤色相似度的灰度图。

3.3 肤色相似度矩阵计算

在得出相似度计算模型之后，扫描整幅图像的所有像素，利用已转化为 YCbCr 颜色空间像素的 Cb、Cr 分量计算其是肤色的概率，最终得到对应像素的二维肤色相似度矩阵。如图 3-2 所示，P 为图像的像素值，T 运算即为公式 3-6 相似度概率运算。

$$\begin{bmatrix} P_1 & P_2 & P_3 \\ P_4 & P_5 & P_6 \\ P_7 & P_8 & P_9 \\ \vdots & \vdots & \vdots \end{bmatrix} \rightarrow \begin{bmatrix} T(P_1) & T(P_2) & T(P_3) \\ T(P_4) & T(P_5) & T(P_6) \\ T(P_7) & T(P_8) & T(P_9) \\ \vdots & \vdots & \vdots \end{bmatrix}$$

图 3-2 相似度矩阵计算

对每一个像素点计算完毕之后，对矩阵进行归一化处理。首先统计出矩阵中最大值 T_{\max} ，对每个 T 用 T_{\max} 相除，则所有相似度归一到区间[0, 1]。T= P(Cb,Cr)所表示概率值的含义是该像素点与肤色的相似程度，归一化后当其值越接近 1 表示该像素点是皮肤的可能性越大，越不像肤色其值越接近 0。

为了显示像素肤色相似度计算的结果，再将其矩阵乘以 255，归一化到区间[0, 255]，如此即将彩色图像转换成相似度灰度图像。

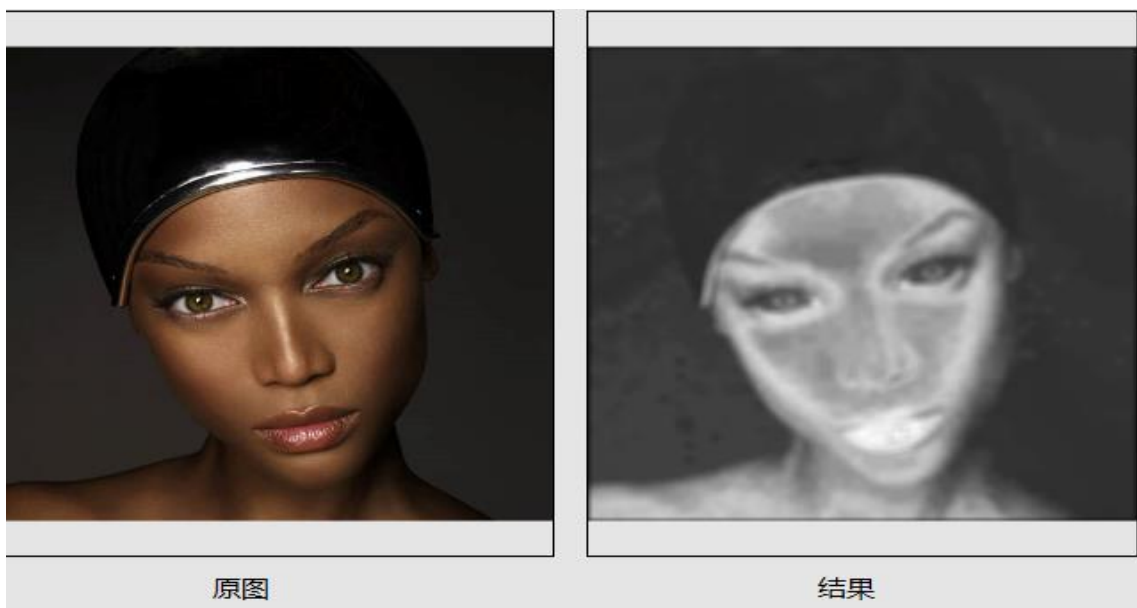


图 3-3 肤色相似度矩阵计算结果 1



图 3- 4 肤色相似度矩阵计算结果 2

如图 3- 3 肤色相似度矩阵计算结果 1 和图 3- 4 肤色相似度矩阵计算结果 2 所示，是本论文设计的软件调试时的截图，两份原图与肤色相似度矩阵计算结果的对比图，其中结果图已经经过中值滤波处理（详见节 3.4）。可以看到计算效果比较理想，越像肤色的区域其灰度值越接近 255，即直观印象显示为更接近白色，非肤色区域灰度值偏向 0，显示为更接近黑色。

3.4 相似度矩阵的中值邻域滤波

在第 2 章图像处理基础的图像预处理中，我已经介绍过图像的空域增强方法，在此基于肤色检测的识别算法中，为了后期的人脸区域定位处理，需要将肤色相似度矩阵计算得到的灰度图像进行空间域增强，抑制图像噪声，平滑化图像。此处采用中值滤波方法。

中值滤波最早由 J.W.Jukey 提出的非线性平滑处理方法，最初的应用是在时间序列的一维信号处理中，现如今常被应用到二维数字图像的处理中，主要作用是抑制图像噪声^[21]。中值滤波的实现方法是一种邻域运算，基本的思想是：在整幅图像

中滑动运算窗口，扫描到的当前像素与模版中心重合，每一步对设定规模的局部区域的像素灰度等级排序，取该区域中灰度的中值作为当前像素的灰度值。在实际操作时若取区域的灰度平均值作为当前像素平均值则为邻域平均法，接近于中值法。其本质是用与相邻像素接近的灰度值来取代跟邻域相差较大的灰度值从而达到平滑化处理。

$$M_4 = \frac{1}{4} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad M_8 = \frac{1}{8} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (3-7)$$

公式 3-7 所示为典型常用的四邻域和八邻域模版的平均算法，若是取中值则先排序，取出中值赋值给模版中心对应像素。若是降噪效果不佳，可能原因是图像像素规模较大，可以将模版调大，如 7x7，9x9 模版。计算方法如公式 3-8 所示平均法和公式 3-9 所示中值法。

$$G_{(x,y)} = \frac{1}{n} \sum_{(i,j) \in K} G_{(i,j)} \quad (3-8)$$

式中： $G_{(x,y)}$ 表示扫描到的像素灰度值，即模版中心

n 为模版大小

K 为模版对应的局部区域

$G_{(x,y)}$ 为像素灰度值

$$G_{(x,y)} = Mid\{G_{(i,j)} \mid (i,j) \in K\} \quad (3-9)$$

式中： Mid 表示取中位数操作

如图 3- 5 中值滤波前和图 3- 6 中值滤波后所示是本论文系统调试时的相似度矩阵灰度图滤波降噪前后的截图。起初采用 3x3 模版中值滤波，但是降噪效果不明显，后更改为 9x9 矩阵模版平均值法（依然成为中值滤波）。效果如图所展示，可以看到，滤波前图像噪声较为明显，滤波之后图像明显变得更为平滑，人脸边缘保存完整。



图 3- 5 中值滤波前



图 3- 6 中值滤波后

3.5 二值化处理

从以上操作的结果中可以看到，肤色相似度矩阵产生的灰度图并不能非常清晰地分辨人脸和非人脸区域，为此需要对相似度灰度图进行二值化处理^[18]。将人脸区域灰度值设置为 255，非人脸区域设为 0，如此产生黑白分明的二值图，区分检测操

作的时候将变得更加简便高效。

本文采用的二值化方法是设定一个全局的灰度值阈值 K ，扫描整幅图像像素灰度值，与 K 比对，若灰度值大于 K 则设置其为 255，即确定为人脸区域，显示为白色；若像素灰度值小于 K ，则设置为 0，确定为非人脸区域，显示为黑色。

$$G_{(x,y)} = \begin{cases} 0, & G_{(x,y)} < K \\ 255, & G_{(x,y)} \geq K \end{cases} \quad (3-10)$$

式中： $G_{(x,y)}$ 表示扫描到的图像像素点的灰度值

K 的选取能够决定二值化的结果准不准确，是否需要调整，本文中二值化处理设定阈值 K 为肤色相似度灰度图的灰度平均值。经测试验证，此值较为合理，符合预期。

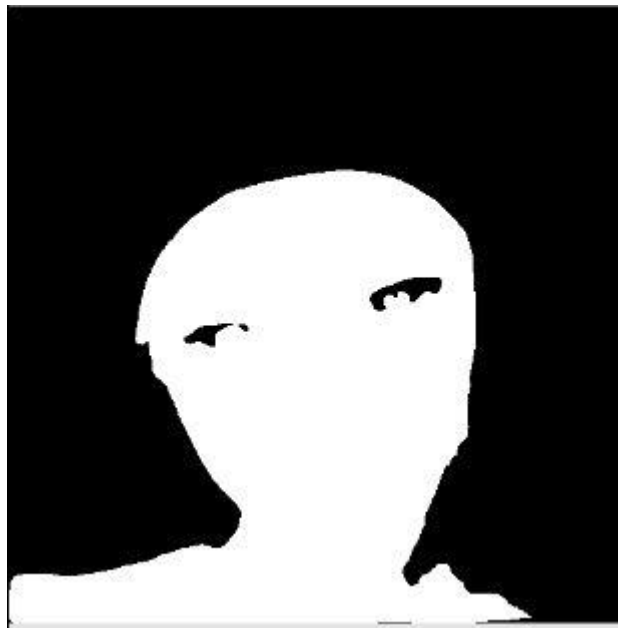


图 3- 7 二值化后图像

如图 3- 7 二值化后图像所示，与图图 3- 6 中值滤波后图像二值化的处理结果，效果较好，图像中肤色区域保留完整，人脸边缘平滑，图像几乎看不到明显噪点。为人脸区域定位做好铺垫，二值化的图像即为定位操作使用。

3.6 人脸区域定位

在二值化图像基础上对人脸进行定位，其方法主要有基于边缘检测和基于区域检测两大类，前者有诸如 Sobel 算子检测、Log 算子检测和 Canny 算子，后者有阈值法、区域生长与合并方法和一些基于统计学的算法^[22]。在本文中人脸识别考虑单人图像的检测，经过之前的处理之后干扰因素影响较小，因此二值化后图像人脸定位可以参照已有算法设计出自己的方法，本文称此方法为基于轮廓边界的脸部定位方法。

基于轮廓边界的脸部定位方法需要考虑诸多因素，包括与肤色相似的背景形成的高相似度区域、除脸部外其他如手臂颈部等裸露的皮肤，这些都会被二值化为白色区域，对脸部定位形成干扰。所以定位方法的规则一定要考虑到去除影响，避开非人脸区域，具有普遍性。

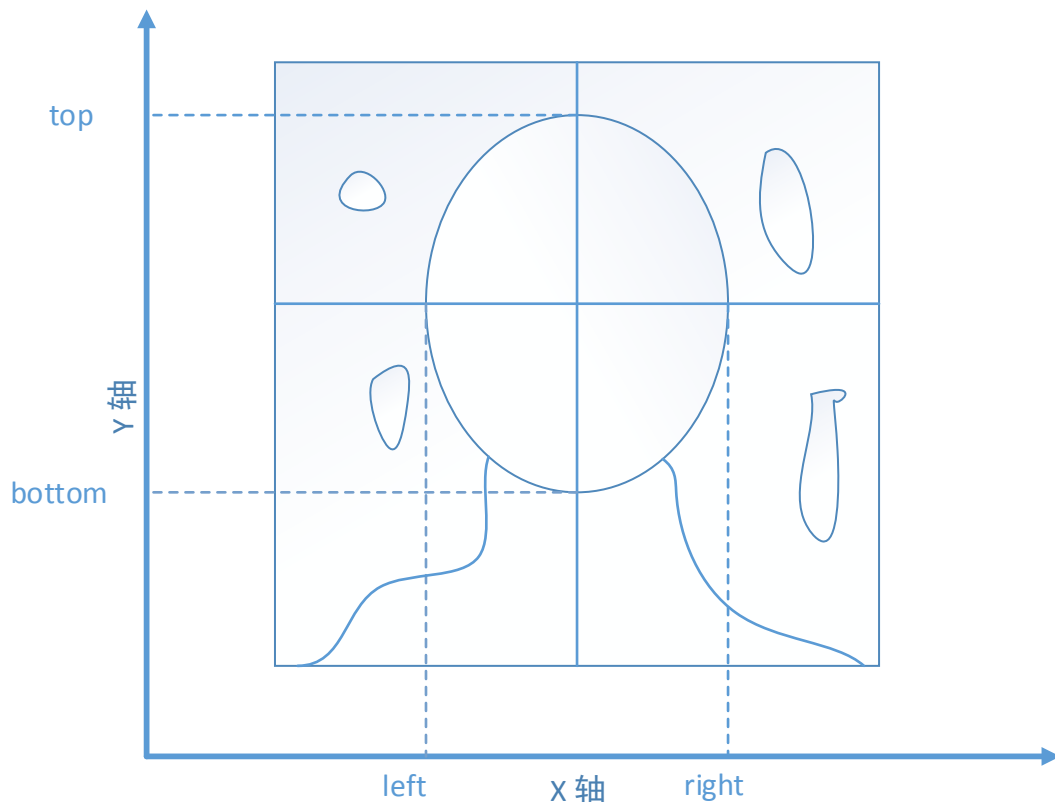


图 3-8 脸部定位示意图

如图 3-8 脸部定位示意图所示是脸部定位处理的示意图，具体步骤如以下：

- 1) 二值化后的黑白二维图像像素阵列以对应二维坐标系考量，在 Y 轴上对像素阵列灰度值为 255（即为白色，人脸区域）的像素数目进行投影，得到最大值所对应的 X 轴坐标，此坐标即为人脸区域高度对应的像素列，理论上额头的最高点所在列。再利用最大值归一化每一列人脸区域的像素值，相当于求取频度；
- 2) 利用最大频度列求人脸区域的左边界 left 和右边界 right，设定左右边界列归一化后频度阈值为 0.2。从最大频度列出发，向左和向右扫描，检测到左边频度小于 0.2 时停止，该列对应 X 坐标即为左边界 left；同理向右扫描，检测到右边频度小于 0.2 时停止，该列对应 X 坐标即为右边界 right；
- 3) 如同第一步，在 X 轴上对像素阵列灰度值为 255（即为白色，人脸区域）的像素数目进行投影，得到最大值并进行归一化处理。此最大频度行即对应人脸区域的宽度行；
- 4) 因为脸部直接连接颈部，或者其他皮肤区域露出，人脸区域的下边界难以确定，所以先求出区域上边界。从最大频度行出发，向上扫描，检测到频度小于 0.2 时停止，此行对应 Y 坐标即为上边界 top；
- 5) 求下边界 bottom，当人脸区域拟合为矩形时，其长度和宽度比值通常为 6:5，所以 bottom 会等于 $top - 1.2 * (right - left)$ ；
- 6) 得出用于人脸定位的矩形框位于位图中的位置与其长宽，在图像中绘制矩形框，起始坐标为(left,top)，宽度为 right-left，高度为 top-bottom。

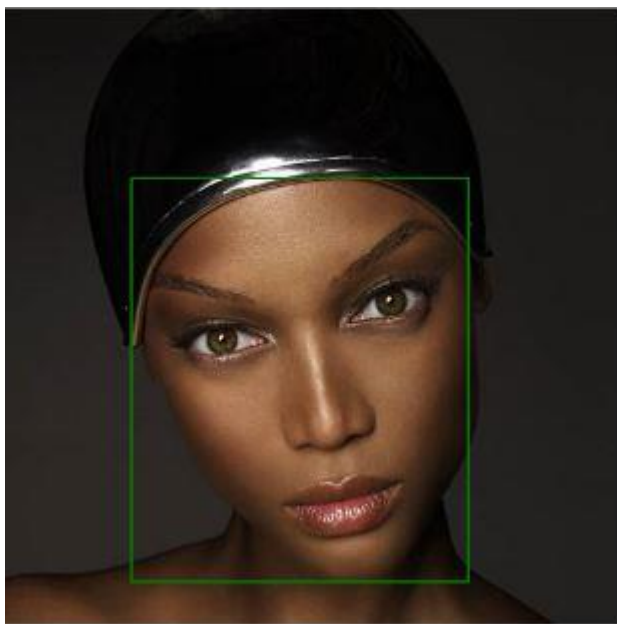


图 3- 9 人脸定位结果展示 1



图 3- 10 人脸定位结果展示 2

图 3- 9 人脸定位结果展示 1 和图 3- 10 人脸定位结果展示 2 是本论文实现系统调试时的两张截图，实现定位操作时用绿色的矩形框出识别出的人脸区域，可以看到，定位效果较为理想，基本较为准确地定位出了从图像中识别出的人脸区域。边界检测的频度阈值 0.2 较为合理。但是缺陷依旧明显，在图 3-10 中可以窥出端倪，下巴区域没有进入识别框，这是因为人脸区域拟合矩形的长宽比 6:5 设定并不具有

很好的普遍性。另外该定位方法受限于设定条件，即单人脸、干扰因素较小，若需要识别出一张图像的多张人脸，需要改变定位方法（修改为设定区域面积阈值的多个连通区域的检测定位方法），定位操作之前的步骤无需变化。

3.7 人脸区域比对

得到人脸区域的定位用拟合矩形框的坐标和长宽之后，即可将人脸区域从肤色相似度计算得到的灰度图中剪裁出，用作对两份单人图像的人脸对比原料。对比方法采用基于 Bhattacharyya 距离的单通道灰度直方图对比方法，结果是得出两份人脸区域灰度图的相似度。

图像的灰度直方图是数字图像处理中经典有用的工具，能够反映出图像中各灰度级别的分布情况。如图 3-11 所示是一幅图像的灰度直方图示意图，横坐标表示[0, 255]范围的灰度级，纵坐标代表灰度级对应的像素数目。

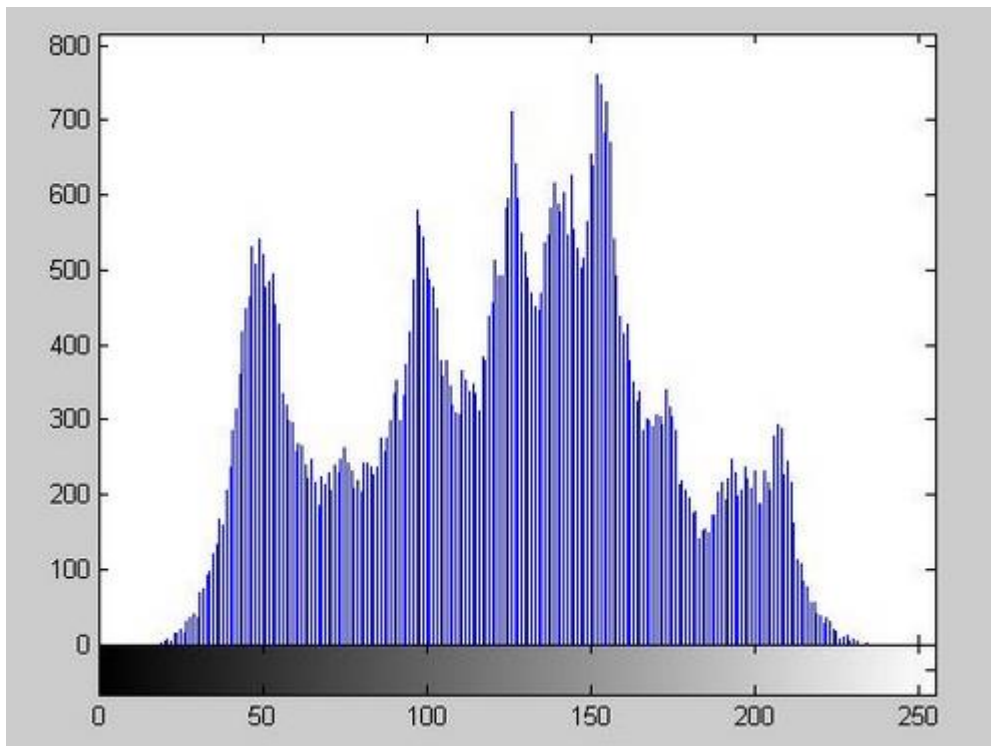


图 3- 11 图像灰度直方图示意图

Bhattacharyya 距离（巴氏距离）是统计学中测量两个离散数据或者连续概率分布的相似性，在直方图比对中的应用可适用到图像比对。设两份人脸区域灰度图的

直方图分别为 $H1$ 和 $H2$ ，衡量两份直方图相似度的对比标准则为 $d(H1, H2)$ ，那么 Bhattacharyya 距离的计算方式如公式 3-11 所示：

$$d(H1, H2) = \sqrt{1 - \frac{1}{\sqrt{H1 * H2 * N^2}} * \sum_M \sqrt{H1(M) * H2(M)}} \quad (3-11)$$

式中：N 代表直方图中 bin（图像灰度特征的统计量）的数目

M 为灰度级

所以本文中用到的人脸比对的方法实现步骤主要有三步：

- 1) 在之前的处理中获取人脸区域的定位用拟合矩形框的坐标和长宽，将人脸区域从肤色相似度计算得到的灰度图中剪裁出，得到基本只包含人脸的灰度图；
- 2) 从两份待比较的人脸区域灰度图中分别创建单通道灰度直方图；
- 3) 计算两灰度直方图的 Bhattacharyya 距离，得到两份人脸区域的相似度。

此对比方法优点在于高效率、易实行，而缺陷相当明显，它没有考虑人脸区域内各像素在区域内的位置信息，因此该人脸对比方法还有很大的改善空间，在本文中暂且使用该方法进行识别系统的实现。

第 4 章 人脸识别系统实现

本文论述人脸识别算法理解与人脸识别系统实现两大部分，在此之前我们完成了前半部分，介绍了图像处理基础和人脸识别算法简要归纳，并且选定基于肤色的人脸识别算法进行剖析。奠定理论基础之后进行第二部分，完整人脸识别系统的实现。

本论文实现的系统主要实现三大功能：静态图像人脸检测与定位、摄像头人脸检测与定位和人脸图像比对。

其中人脸检测与定位除了采用与第 3 章对应采用 C#编写实现的基于肤色检测的识别算法外，还利用了开源框架 EmguCV 的基于 HaaCascade 分类器的 Adaboost 算法进行实现，两个模块之间可以形成对比，加深对算法理解。

人脸图像比对模块采用基于巴氏距离（BHATTACHARYYA）的直方图比对方法，本系统并没有类似实现人脸读取身份信息这样的扩展功能，若要实现，只需要读取数据库图片与当前图像比对，按比对结果（即相似度）排序，选出相似度最大的人脸 ID 若超过设定阈值则进行相应的身份读取。实现起来较为容易，在本论文系统中并没有必要，因为我的重点是人脸识别算法探索。

4.1 系统实现方案

WPF(Windows Presentation Foundation)技术是.net 平台开发上浓墨重彩的一页，它是一种统一桌面与网络应用、统一模型和框架的应用，开发的应用具有优秀的人机交互和强大简练的系统框架，其最大的特色是清晰地将设计和功能两大部分分开，使它们可以分开进行和协调工作。WPF 应用程序使用标记和代码隐藏模型，使用并且扩展了许多以前的技术概念和类，其中包括 Window 窗体、ASP.NET、XML、图形设备绘制技术 GDI+和数据绑定技术等。它的强大与便捷深受开发人员喜爱。

基于 WPF 的开发优点与学习价值，本系统考虑采用 WPF 技术编写独立于.net 平台的应用程序。界面使用 XAML 语言设计（Extensible Application Markup

Language),识别算法和图像各阶段处理函数使用 C#语言编写,基本系统框架类似于三层架构:表现层(UI)、业务逻辑层(BLL: Business Logic Layer)、图像处理层(IHL: Image Handle Layer)。UI层为WPF桌面应用的界面设计,BLL层对应界面控件事件响应处理,IHL层对应图像处理的具体实现。

系统最终需要实现的四大功能:

- 1) 基于肤色检测的人脸检测与定位
- 2) 基于 Adaboost 算法的人脸检测与定位
- 3) 摄像头人脸检测与定位
- 4) 人脸图像比对

其中2和4的关键算法调用开源框架 EmguCV 的 API 实现,其他部分所有代码独立编写。

4.1.1 软件环境

开发环境所用操作系统为 64bits windows 8.1,开发环境为 Blend for Visual Stdio 2013 和 Visual Stdio2013。

开发的人脸识别系统是基于框架.net framework 4.5 的 WPF 应用程序,系统界面及控件消息响应处理事件在 Blend for Visual Stdio 2013 上搭建,结合利用可视化操作和 XAML 语言开发使工作的效率得以提高。各个模块的具体实现代码在 Visual Stdio2013 开发,各个模块之间的层次调用关系使用程序集 namespace 构建。

程序的调试也需要 Blend 和 VS 结合进行。系统设计阶段的架构图和流程图等使用 Visio 2013 编辑。

4.1.2 硬件环境

本系统采用的计算机CPU为 AMD Llano APU A6-3400M,4核,单核主频 1.4GHz,4G DDR3 内存条。系统中使用的视频采集设备为笔记本内置摄像头 Lenovo EasyCamera,摄像头采集速度为 30 帧/秒。

4.2 系统的详细设计

4.2.1 系统功能设计

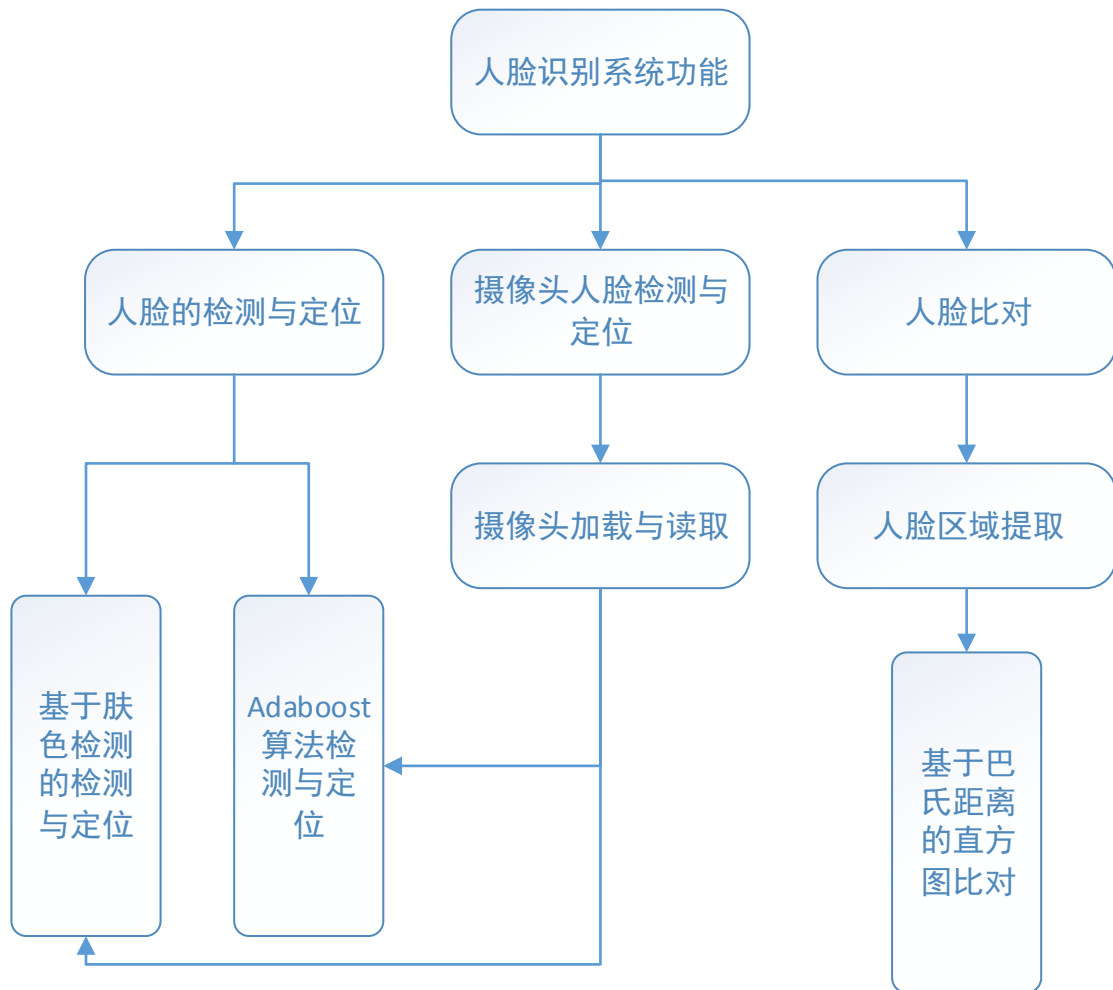


图 4-1 系统功能图

系统的功能总览如图 4-1 所示，主体功能包括两种方法的静态图像的人脸检测与识别、摄像头人脸识别和人脸比对。第一个功能为后面两个功能的基础，系统只有两类用户，即软件开发人员和需要进行人脸识别的普通用户，在软件功能面向用户上无区分。分别进行功能的详细描述如下：

1) 基于肤色检测的人脸检测与定位功能

功能编号：1

功能名：打开图片

简要叙述：用户从本地电脑中选择一张图像进行人脸检测与定位，系统读取文件显示图片，并建立 **Bitmap** 实例供后期处理。（此功能在其余几个大功能中也有应用）

输入数据流：**jpg**、**bmp**、**png**、**jpeg** 等格式图像文件。

输出数据流：**Bitmap** 实例。

处理逻辑：获取打开文件对话框中已经过滤的图像文件地址，初始化 **Bitmap** 实例。

用 户：所有。

功能编号：2

功能名：颜色空间变换

简要叙述：将位图从 **RGB** 颜色空间转换到 **YCbCr** 颜色空间。

输入数据流：位图 **RGB** 颜色分量。

输出数据流：**YCbCr** 颜色空间分量。

处理逻辑：将位图的像素数据转入 **Bytes** 数组处理，每三个字节分别代表位图像素的 **B**、**G**、**R** 分量数据，利用公式 3-2 计算 **YCbCr** 颜色空间分量。

用 户：所有。

功能编号：3

功能名：肤色相似度灰度图产生

简要叙述：利用肤色相似度模型从原图转变为肤色相似度灰度图

输入数据流：**YCbCr** 颜色空间分量。

输出数据流：肤色相似度灰度图。

处理逻辑：由 **YCbCr** 颜色空间分量利用肤色相似度模型计算对应像素的肤色相似概率，计算肤色相似度矩阵并归一化，由 **bytes** 数组得到灰度图。

用 户：所有。

功能编号：4

功能名：肤色相似灰度图的平滑处理

简要叙述：利用邻域滤波方法对灰度图进行消除噪声的平滑化处理，将结果图像在界面显示。

输入数据流：肤色相似度灰度图。

输出数据流：滤波后的肤色相似度灰度图。

处理逻辑：使用滤波模版对相似度灰度图的像素阵列进行滤波处理，计算方法详见 3.4 节。

功能编号：5

功能名：肤色相似灰度图二值化处理

简要叙述：以设定阈值二值化肤色相似灰度图，白色标记人脸区域，黑色标记非人脸区域，并将二值化的结果图在界面显示。

输入数据流：滤波后的肤色相似度灰度图。

输出数据流：二值化灰度图。

处理逻辑：扫描将滤波后的肤色相似度灰度矩阵，设定全图灰度平均值为区分人脸非人脸区域阈值，比较所有像素灰度值，人脸区标记灰度 255，非人脸区标记 0，输出结果灰度图。

用 户：所有。

功能编号：6

功能名：二值化图像的人脸区域定位

简要叙述：从二值化后的灰度图像中检测并定位出脸部区域，在界面上呈现结果。

输入数据流：二值化后的灰度图像。

输出数据流：待检测图像的人脸区域定位矩形。

处理逻辑：使用二值化的灰度图像，利用 3.6 节中介绍的方法获取脸部区域的拟合矩形的初始点坐标和长宽，在最初的位图中绘制出定位矩形框，再将结果显示到界面。

用 户：所有。

2) 基于 Adaboost 算法的人脸检测与定位功能

功能编号：7

功能名：Adaboost 算法的 Haar 分类器样本训练

简要叙述：在应用 EmguCV 的 Adaboost 算法人脸检测 API 之前，对其 Haar 分类器进行样本训练。

输入数据流：序列化的人脸与非人脸数据 XML 文件。

输出数据流：可用的检测人脸用的实例。

处理逻辑：利用序列化的人脸与非人脸数据 XML 文件的路径传递进分类器 HaarCascade 的构造函数，构造函数开始训练分类器，产生可用的检测人脸用的实例。

用 户：所有。

功能编号：8

功能名：Adaboost 算法检测与定位人脸

简要叙述：使用 HaarCascade 类的 Detect 方法检测与定位人脸，适用含多张人脸图像。这一功能主要为了解 Adaboost 识别算法的应用，并与基于肤色的算法对比。

输入数据流：待检测的位图。

输出数据流：算法检测出的人脸定位矩形框数组。

处理逻辑：使用 HaarCascade 类的 Detect 方法检测与定位人脸。

用 户：所有。

3) 摄像头人脸检测与定位功能

功能编号：9

功能名：加载摄像头

简要叙述：获取摄像头设备列表，用户选中一个设备后，开启该设备获取视频流。

输入数据流：加载摄像头的控件消息。

输出数据流：选定摄像头的视频流。

处理逻辑：检测摄像头设备，将它们罗列出供用户选择；用户选定摄像头后开启该设备，获取视频流的每一帧将其显示在软件界面上实现本系统的视频流。

用 户：所有。

功能编号：10

功能名：摄像头人脸检测与定位

简要叙述：由于实时的视频人脸识别算法运算需要时间而造成卡顿，改变策略，通过摄像头拍照并对照片进行人脸检测与定位。

输入数据流：按下快门时视频流的一帧图像。

输出数据流：检测并定位出人脸的照片。

处理逻辑：按下快门时刻选定视频流的那一帧图像，传入人脸识别算法中进行处理，导出结果图，识别算法可以在之前介绍的两种中二选一。

用 户：所有。

4) 人脸图像比对功能

功能编号：11

功能名：提取人脸区域

简要叙述：从两张待检测图像中提取出人脸区域。

输入数据流：两张单人图像。

输出数据流：人脸区域位图或者为空（从图像中未检测到人脸）。

处理逻辑：使用打开图片功能（功能编号为1）分别打开两张待检测人脸图像，调用算法获取人脸区域的定位，剪裁出该部分位图，若都存在人脸则可进行比对。

用 户：所有。

功能编号：12

功能名：直方图方法比对人脸

简要叙述：利用基于巴氏距离的直方图比对方法对提取出人脸区域的两张位图进行比较，返回相似度。

输入数据流：两份提取出人脸区域的位图。

输出数据流：两张人脸图像的相似度。

处理逻辑：从两份提取出人脸区域的位图中创建单通道灰度直方图，对两份灰度直方图进行基于 Bhattacharyya 距离的比对，输出比对结果。

用 户：所有。

4.2.2 系统模块结构

本文的人脸识别系统的功能主要按照识别算法的实际应用主线构造，而在确定出要实现的具体功能基础上进行模块结构的设计。系统总体模块结构如图 4-2 所示。

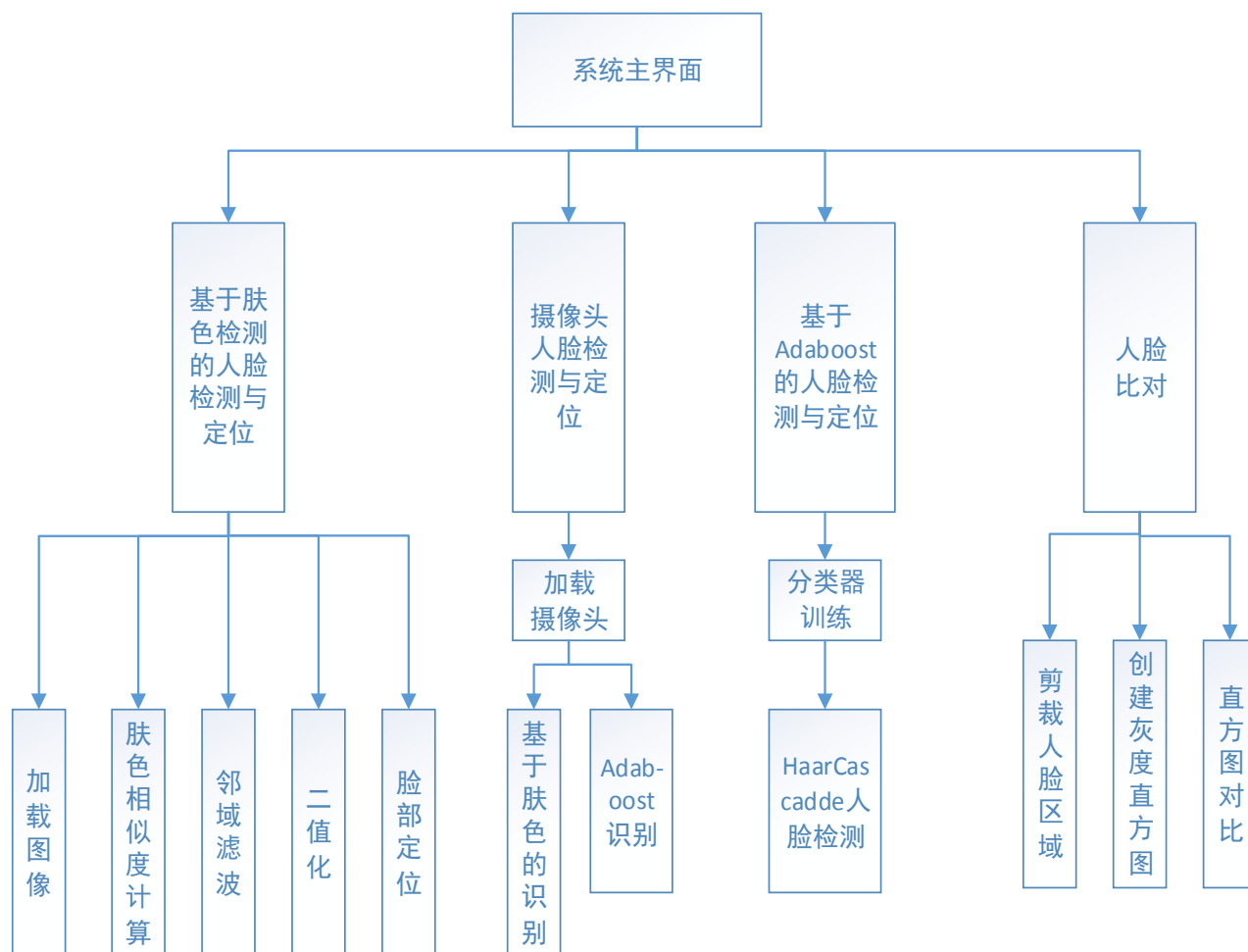


图 4-2 系统模块结构图

4.3 系统各模块实现

4.3.1 系统界面模块

系统的界面的开发使用 Blend for Visual Studio 的可视化设计工具，结合 XAML 语言编写进行。设计目标优先是功能清晰，再者界面简洁美观，具体各界面的设计如下几图展示。

- 1) 主界面。使用 TabControl 控件进行功能分区。在以下图中可以看到。
- 2) 基于肤色的人脸检测与定位界面。其中下方三个按钮代表基于肤色检测的识别算法的几个重要步骤呈现功能，它们具有操作的先后顺序，若没有按照顺序会弹出消息框 MessageBox。提示装载图像的控件为 Image 控件。如图 4-3 所示，该图为系统调试时截图，此时正处于肤色相似度灰度图计算阶段。

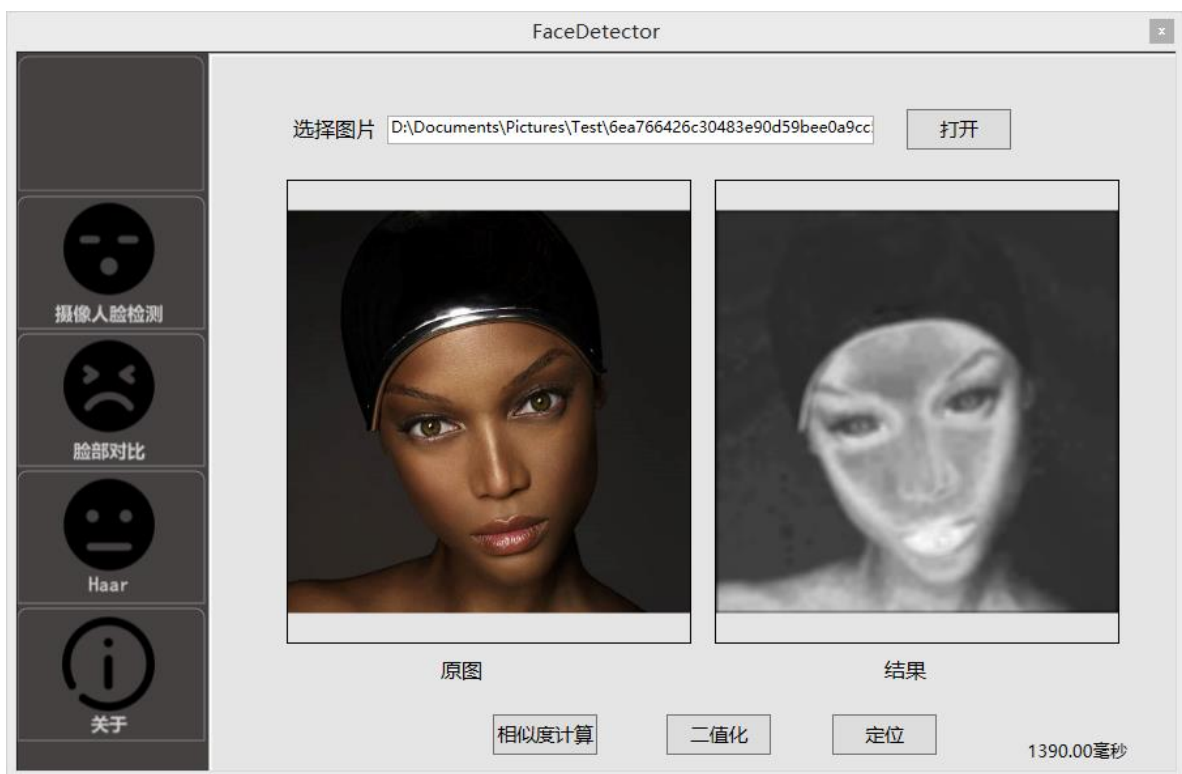


图 4-3 基于肤色的人脸检测与定位界面

- 3) 摄像头人脸检测界面。加载摄像头设备列表在 Combobox 中供用户选择，左侧 Image 控件播放视频流，右边 Image 显示检测结果，“快门”按钮代表获取一帧并进行基于肤色的检测，“Haar”按钮代表基于 HaarCasCade 分类器的 Adaboost 方法检测，“关闭”按钮即关闭摄像头。

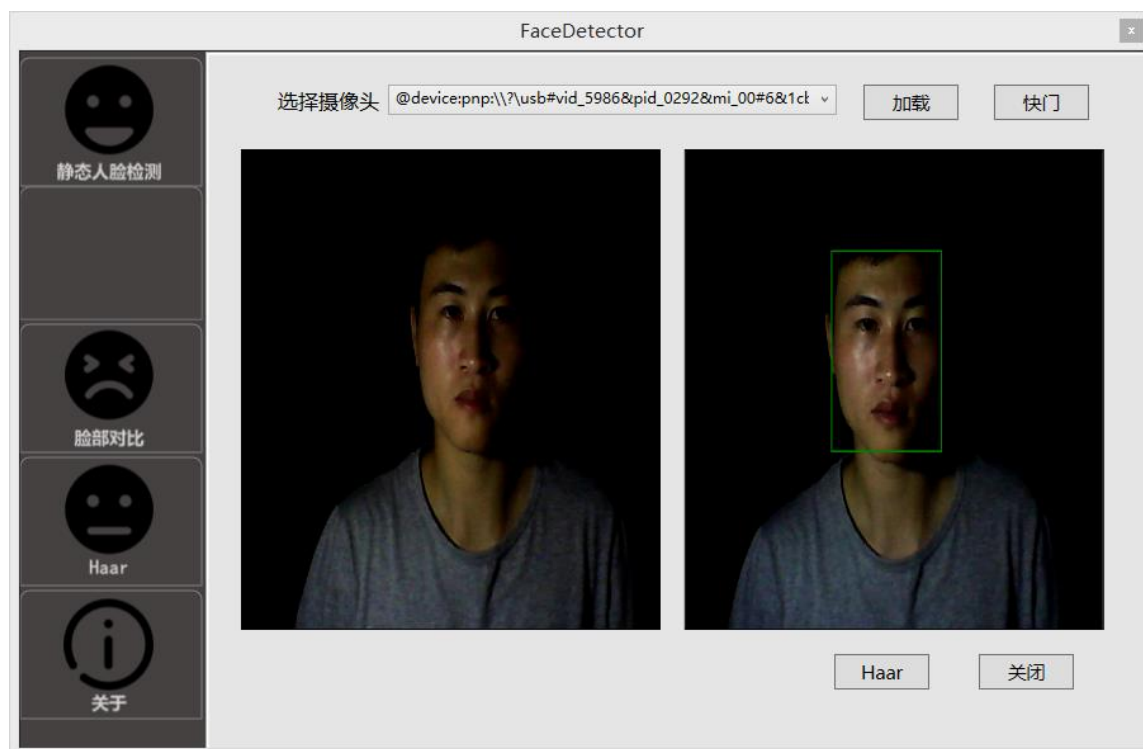


图 4- 4 摄像头人脸检测界面

- 4) Adaboost 算法人脸检测界面。用户选定图片后，直接对图片进行人脸检测与定位，显示在 Image 控件中；若没有检测到人脸，则弹出消息框提示“没有检测到人脸信息”。

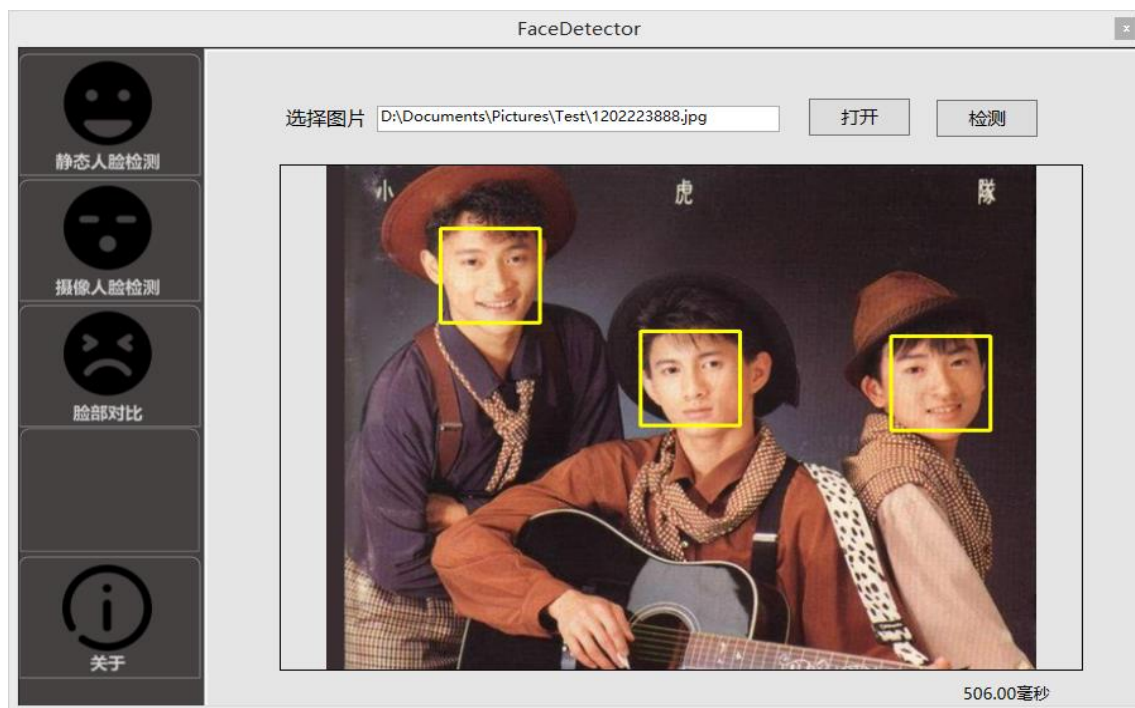


图 4- 5 Adaboost 算法人脸检测界面

- 5) 人脸比对界面。用户选定两张图像之后，点击“比对”按钮，如果在两张图像中都检测到人脸，则将比对结果显示在相似度 Label 中，否则弹出消息框提示哪张图像没有检测到人脸。

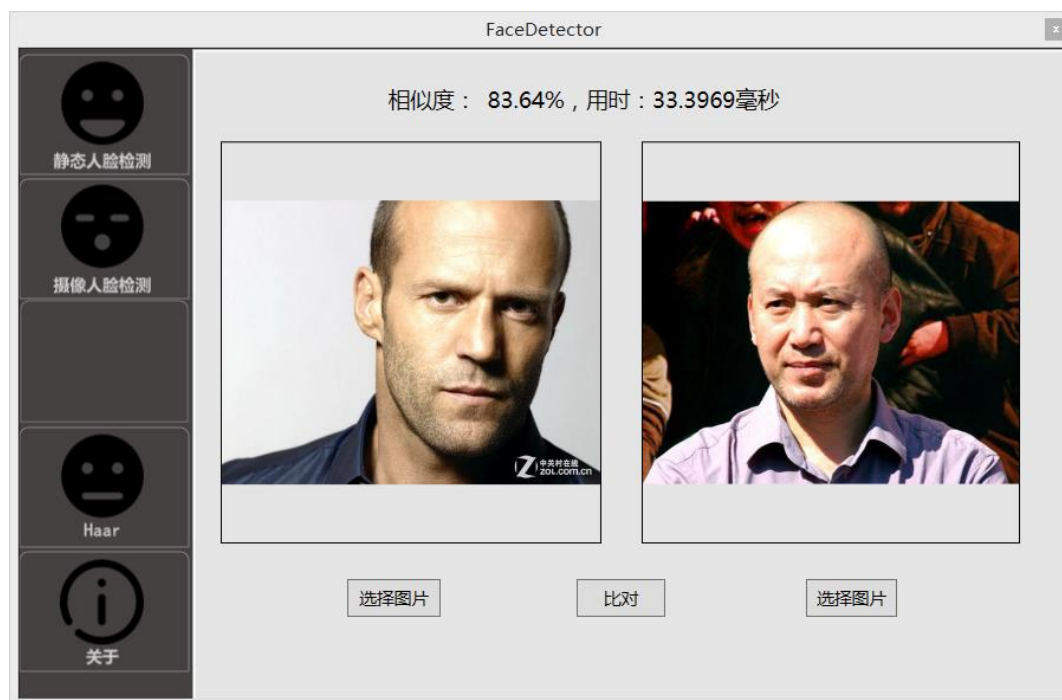


图 4- 6 人脸比对界面

6) 关于界面，仅仅是一个声明。



图 4- 7 软件声明界面

4. 3. 2 基于肤色检测的人脸识别模块

此模块作为第 3 章中基于肤色检测的人脸识别算法的实现，主要包含 RGB 到 YCbCr 颜色空间变换、肤色相似度矩阵计算、领域滤波、相似度灰度图像二值化处理、人脸区域定位五大步骤。在软件中可以实现展示肤色相似度计算结果灰度图、二值化后灰度图和最终的定位结果彩色图。图 4- 8 人脸肤色相似度算法流程展示的是前三步骤的处理顺序。

该模块的函数接口为 `public static Bitmap SkinSimDetect(string imgFile)`，重载接口为 `public static Bitmap SkinSimDetect(Bitmap srcBitmap)`。

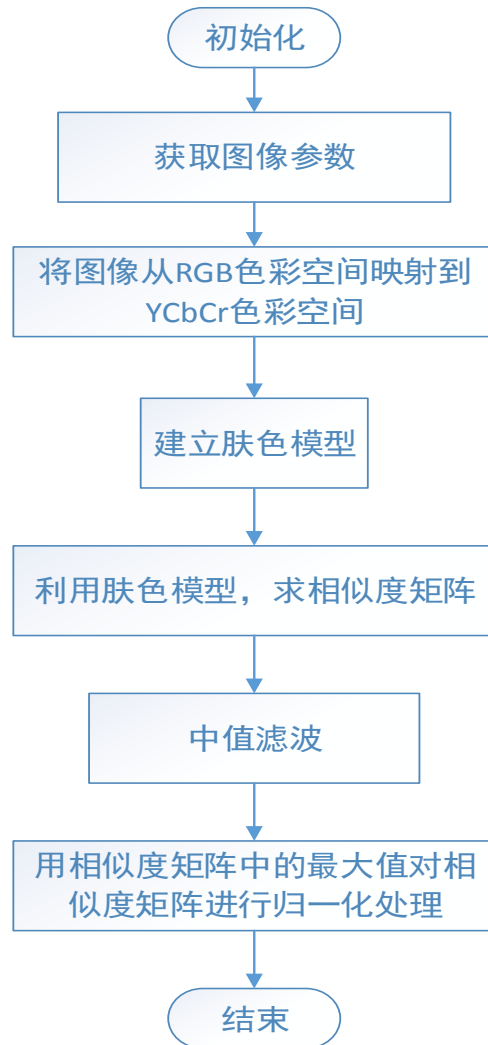


图 4- 8 人脸肤色相似度算法流程

该模块详细代码见本毕业设计源代码，该模块位于命名空间 `methodcore`。各个子模块介绍如以下，并且贴出核心实现代码

1) 位图从 RGB 到 YCbCr 颜色空间变换

因为人类的肤色相对比较集中在 YCbCr 颜色空间，也是像素的肤色相似度计算模型的基础，需要将位图从 RGB 颜色空间中转换过来。

利用待检测位图实例化 `BitmapData` 类，从 `BitmapData` 中将像素数据导入到 `Bytes` 数组，此时每个像素占 3 bytes，分别对应像素的 B、G、R 分量，利用此规律在对数组进行扫描时按公式 3-2 计算出 YCbCr 颜色空间的 Cb 和 Cr 分量。

以读写模式打开待检测位图的数据，存储到 24bitsRGB 空间形式 `BitmapData` 中

```
BitmapData src_BmData = sourceBitmap.LockBits(new Rectangle(0,0,wide,height)
,ImageLockMode.ReadWrite,PixelFormat.Format24bppRgb);
```

转入到 bytes 数组中

```
byte[] src_Values=new byte[src_BmData.Stride * height];
```

```
System.Runtime.InteropServices.Marshal.Copy(src_ptr,src_Values,0,
src_Values .len);
```

开始 Cb 和 Cr 分量的计算

```
for(int i=0;i<height;i++)
```

```
    for (int j = 0; j < wide; j++)
```

```
    {    int k=3*j;
```

```
        int C_b = (int)src_Values[i * sd + k];
```

```
        int C_g = (int)src_Values[i * sd + k+1]; ;
```

```
        int C_r = (int)src_Values[i * sd + k+2]; ;
```

```
        double Cb=(128-37.797*C_r/255-74.203*C_g/255+112*C_b/255);
```

```
        double Cr=(128+112*C_r/255-93.786*C_g/255-18.214*C_b/255);}
```

2) 肤色相似度矩阵计算

得到像素的 Cb 和 Cr 分量之后，利用肤色概率模型计算该像素的肤色相似度，

详见 3.2 节和 3.3 节。以下是该部分的实现代码：

定义相似度矩阵数组

```
double[,] pSimArray = new double[height,wide];
```

声明肤色高斯模型参数

```
double Cb_Factor = 117.4361;
```

```
double Cr_Factor = 156.5599;
```

```
double Cov00=160.1301;
```

```
double Cov01=12.1430;
```

```
double Cov10=12.1430;
```

```
double Cov11=299.4574;
```

计算肤色相似度矩阵

```
for(int i=0;i<height;i++)
    for (int j = 0; j < wide; j++)
    {
        int k=3*j;

        double temp=(Cb-Cb_Factor)*((Cb-Cb_Factor)*Cov11-
            (Cr-Cr_Factor)*Cov10)+
            (Cr-Cr_Factor)*(-(Cb-Cb_Factor)*Cov01+
            (Cr-Cr_Factor)*Cov00);

        temp=(-0.5*temp)/(Cov00*Cov11-Cov01*Cov10);

        SimArray[i, j] = System.Math.Exp(temp);}

```

3) 领域滤波

受诸多因素影响刚产生的相似度矩阵会有很多“噪声点”，领域滤波为了之后的二值化处理和人脸区域定位处理，需要将肤色相似度矩阵计算得到的灰度图像进行空间域增强，抑制图像噪声，平滑化图像并且保留物体边缘完整。

起初采用 3x3 模版中值滤波，但是降噪效果不明显，后更改为 9x9 矩阵模版平均值法，具体的代码实现如下：

邻域均值法,从相似度矩阵着手抑制噪声,n 为模版的规模

```
int n = 9;

int midn = (int)(n / 2);

double[,] temp=new double[height+2*midn,wide+2*midn];//存储模版计算用矩阵
初始化

for (int i = 0; i < height + 2*midn; i++)
    for (int j = 0; j < wide + 2*midn; j++)
        temp[i, j] = 0.0;

```

```

for (int i = 0; i < height; i++)
    for (int j = 0; j < wide; j++)
        temp[i + midn, j + midn] = pSimArray[i, j];

```

开始滤波

```

for (int i = 0; i < height; i++)
    for (int j = 0; j < wide; j++) {
        pSimArray[i, j] = 0.0;
        for (int r = 0; r < n; r++)
            for (int c = 0; c < n; c++)
                pSimArray[i, j] += temp[i + r, j + c];
        pSimArray[i, j] /= n * n;
    }

```

邻域滤波之后得到降噪的矩阵，利用相似度矩阵中的最大值对矩阵进行归一化处理，先归一化到[0, 1]区间，此处实现较为简单，代码不赘述。

为了将相似度计算的结果呈现出来，将矩阵变换到[0,255]区间上，将矩阵值赋值到 bytes 数组，将 bytes 数组赋值给新建的灰度图像类型 Bitmap 的 BitmapData。这样就产生了肤色相似度灰度图。

变换到[0, 255]，进行灰度图像显示

```

for (int i = 0; i < height; i++)
    for (int j = 0; j < wide; j++)
        resultValues[i * resultBmData.Stride + j] = (byte)((int)(pSimArray[i, j] * 255));

```

结果灰度图的定义

```

Bitmap resultBitmap = new Bitmap(wide, height, PixelFormat.Format8bppIndexed);
SetGrayscalePalette(resultBitmap); // 设定调色板类型为灰度图像

```

将数据传入 bitmapData

```

System.Runtime.InteropServices.Marshal.Copy(resultValues, 0, resultPtr,

```

```
result_bytes_len);
```

```
解锁, Unlock bitmap
```

```
resultBitmap.UnlockBits(resultBmData);
```

4) 二值化

肤色相似度矩阵产生的灰度图并不能非常清晰地分辨人脸和非人脸区域，为此需要对相似度灰度图进行二值化处理。将人脸区域灰度值设置为 255，非人脸区域设为 0，如此产生黑白分明的二值图，脸部定位操作的时候将变得更加简便高效。

此处设定对灰度图像二值化的阈值为整幅图像的灰度平均值，实现步骤如图 4-9 二值化处理流程所示。

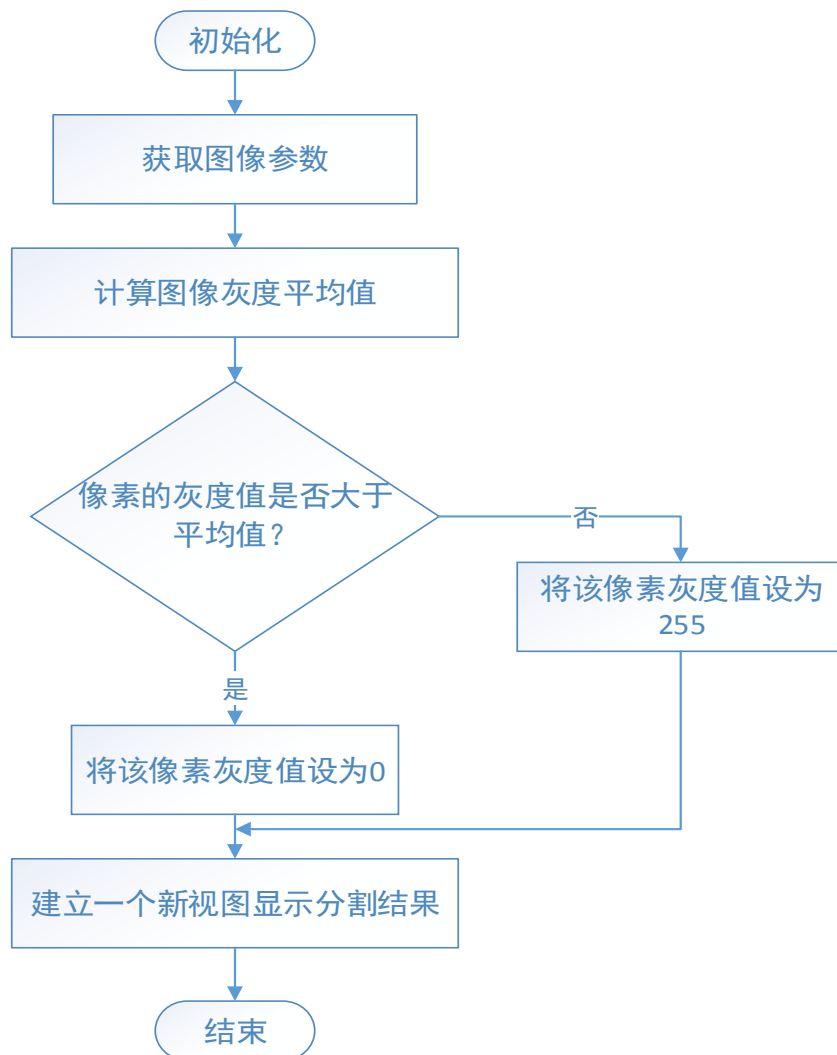


图 4-9 二值化处理流程

二值化处理接口具体实现如下

```
public static Bitmap ImageBinary(Bitmap bt){ int i, j;

int w = bt.Width;//位图宽

int h = bt.Height;//位图高

long pixel_scales = 0;//灰度总值

int threshold;//二值化阈值

获取像素数据

BitmapData btBmData = bt.LockBits(new Rectangle(0, 0, w, h),

ImageLockMode.ReadWrite, PixelFormat.Format8bppIndexed);

System.IntPtr ptr = btBmData.Scan0;

int ws = btBmData.Stride;

int bytes_len = ws * h;

将像素数据导入字节数组

byte[] btValues = new byte[bytes_len];

Marshal.Copy(ptr, btValues, 0, bytes_len);

for (i = 0; i < bytes_len; i++)

pixel_scales += btValues[i];

将灰度平均值设为二值化阈值

threshold = (int)pixel_scales / (w * h);

开始利用阈值二值化所有像素灰度值

for (i = 0; i < h; i++)

    for (j = 0; j < w; j++)

        { if (btValues[i*ws+j] < threshold)    btValues[i * ws + j] = 0;

          else    btValues[i * ws + j] = 255;}

    Marshal.Copy(btValues, 0, ptr, bytes_len);

    bt.UnlockBits(btBmData);return bt;}
```

5) 人脸区域定位

定位方法的详细步骤见 3.6 人脸区域定位介绍，编写实现它的接口为 `public static Bitmap faceLocate(Bitmap bt,string imgFile)`，重载接口为 `public static Bitmap faceLocate(Bitmap bt, Bitmap resultbt)`。以下为具体代码

定位矩形的上下左右边界

```
int left=0, right=0, top=0, bottom=0; int i, j;
```

计数用 X 轴和 Y 轴人脸区灰度值投影分量

```
double[] temph=new double[w];
```

```
double[] tempv=new double[h];
```

定位用临时变量

```
int max = 0;
```

```
int pos = -1;
```

投影到 Y 轴，找出人脸区域最大频度列

```
for (i = 0; i < w; i++) { count = 0;
```

```
    for (j = 0; j < h; j++)
```

```
        if (btValues[j * ws + i] == 255)    count++;
```

```
    temph[i] = count;}
```

```
for (i = 0; i < w; i++)
```

```
    if (temph[i] > max) {
```

```
        max = (int)(temph[i]+0.1);pos=i;}
```

利用最大值归一化

```
for (i = 0; i < w; i++)    temph[i] /= max;
```

找出左边界

```
for (i = pos; i >= 0;i--)
```

```
{ if (temph[i] < 0.2 || i == 0)
```

```
    { left = i; break;}}
```

找出右边界

```
for (i = pos; i <= w; i++)  
{ if (temph[i] < 0.2 || i == w-1) {right = i; break; } }
```

人脸区域投影到 X 轴，求取最大频度行位置

```
max = 0; pos = -1;  
for (i = 0; i < h; i++)  
{   count = 0;  
    for (j = 0; j < w; j++)  
        if (btValues[i * ws + j] == 255) count++;  
    tempv[i] = count;}  
for (i = 0; i < h; i++)  
    if (tempv[i] > max)  
        {   max = (int)(tempv[i] + 0.1);   pos = i;}
```

利用最大值归一化

```
for (i = 0; i < h; i++)  
    tempv[i] /= max;
```

找出上边界

```
for (i = pos; i >= 0; i--)  
{   if (tempv[i] < 0.2 || i == 0) {   top = i;   break;   } }
```

利用比例计算下边界（可省略）

```
bottom = (int)((right - left) * 1.2 + top);
```

计算定位矩形框的长宽

```
rWidth = right - left;   rHeight = bottom - top;
```

绘制定位用的脸部

```
Graphics g ;
```

```
Pen pe = new Pen(Color.Green, 2);
```



```

Bitmap resultbt = new Bitmap(imgFile);

g = Graphics.FromImage(resultbt);

g.DrawRectangle(pe,new Rectangle(left,top,rWidth,rHeight));

```

6) Bitmap 向 BitmapImage 的转化

由于 WPF 控件 Image 显示的数据格式为 BitmapImage 类,而图像处理算法使用 Bitmap 操作,因此 Bitmap 向 BitmapImage 的转化接口使用频率非常高。有多种方法可实现,代码不赘述。

4.3.3 Adaboost 算法人脸检测模块

1) HaarCasCade 分类器训练

为利用开源框架 EmguCV 的 API, 导入其三个命名空间如下:

```

using Emgu.CV;
using Emgu.Util;
using Emgu.CV.Structure;

```

位图转化为其 API 处理格式

```
Image<Bgr, byte> img = new Image<Bgr, byte>(bitmap);
```

利用官方样例提供整合数据序列文件 (haarcascade_frontalface_alt_tree.xml) 训练 HaarCasCade 分类器:

```
HaarCascade haar = new HaarCascade(haarXmlPath); haarXmlPath 为文件地址。
```

2) 利用 HaarCasCade 分类器人脸检测

```
if (haar == null || img == null) return null;
```

```
MCvAvgComp[] faces = haar.Detect(img.Convert<Gray, byte>(), 1.4, 1,
Emgu.CV.CvEnum.HAAR_DETECTION_TYPE.DO_CANNY_PRUNING, new
Size(20, 20));
```

为每一个被检测到的人脸绘制定位矩形框

```
if (faces.Length > 0)
{
    foreach (MCvAvgComp face in faces)
    {
        img.Draw(face.rect, new Bgr(Color.Yellow), 2);
    }
    return img.ToBitmap();
}
else {
    return null;
}
```

4.3.3 摄像头人脸检测模块

摄像头人脸检测模块负责视频流，并从用户拍照中检测和定位人脸，检测和定位的算法可以有两种，由用户选择“快门”或“Haar”按钮决定，分别对应调用基于肤色方法模块和基于 HaarCasCade 分类器方法模块。具体的流程如图 4-10 所示。

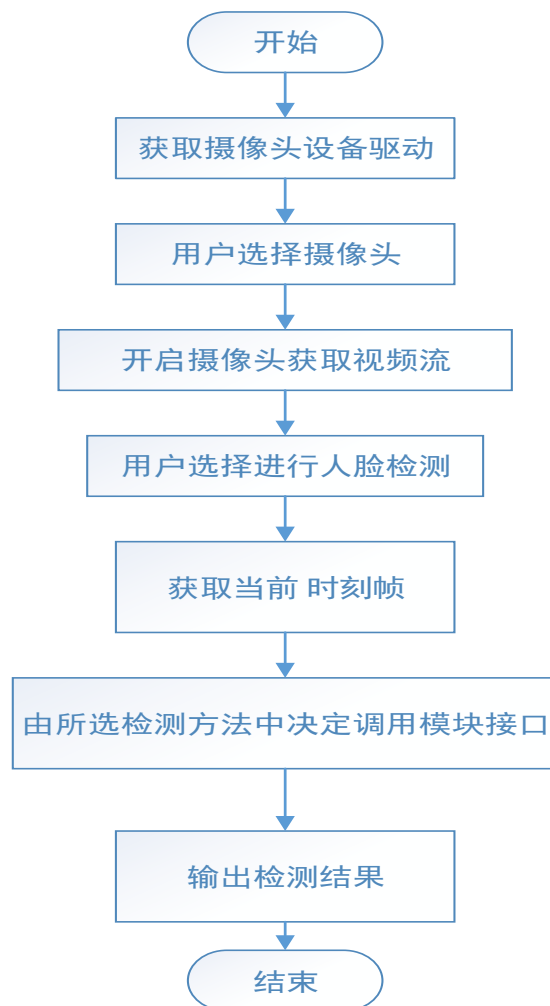


图 4-10 摄像头人脸检测模块流程图

1) 加载摄像头实现视频流

获取所有已插 USB 摄像头驱动信息,为了加载到选择列表 comboBox 供用户选择。方法为 AForge.Video.DirectShow 中开调用接口:

```
FilterInfoCollection filterInfo_collection =
new FilterInfoCollection(FilterCategory.VideoInputDevice);
循环将所有驱动加入到集合中:
camera_List.Add(_filterInfoCollection[i].MonikerString);
开启摄像头
if (comb_cameralist.SelectedItem != null)
{
    videoCaptureDevice=new
    VideoCaptureDevice(comb_cameralist.SelectedItem.ToString());
    videoCaptureDevice.NewFrame += HandNewFrame;}
if (_videoCaptureDevice != null)    videoCaptureDevice.Start();
```

获取视频流的帧显示到软件界面, 其中 HandNewFrame 的创建如下:

```
try{
    this.Dispatcher.Invoke(System.Windows.Threading.DispatcherPriority.Normal,
new Action(() =>
    {
        if (args != null)
            { //imgshoot 为获取到的帧, 用作人脸检测接口传参
            imgshoot = args.Frame.Clone() as Bitmap;
            img_m21.Source = BitmapToBitmapImage(imgshoot);} }));}
catch (Exception exception)
{
    System.Windows.MessageBox.Show(exception.Message);//throw;}
```

2) 调用人脸检测方法

可以选择调用前面介绍的基于肤色检测的人脸识别模块接口或者基于 EmguCV 的 Adaboost 算法人脸检测模块接口将 imgshoot 传入进行人脸检测。

4.3.5 人脸比对模块

该模块利用之前的检测模块处理接口，使用直方图比对的方法进行单人图像之间的人脸比对，处理流程如图 4-10 所示。

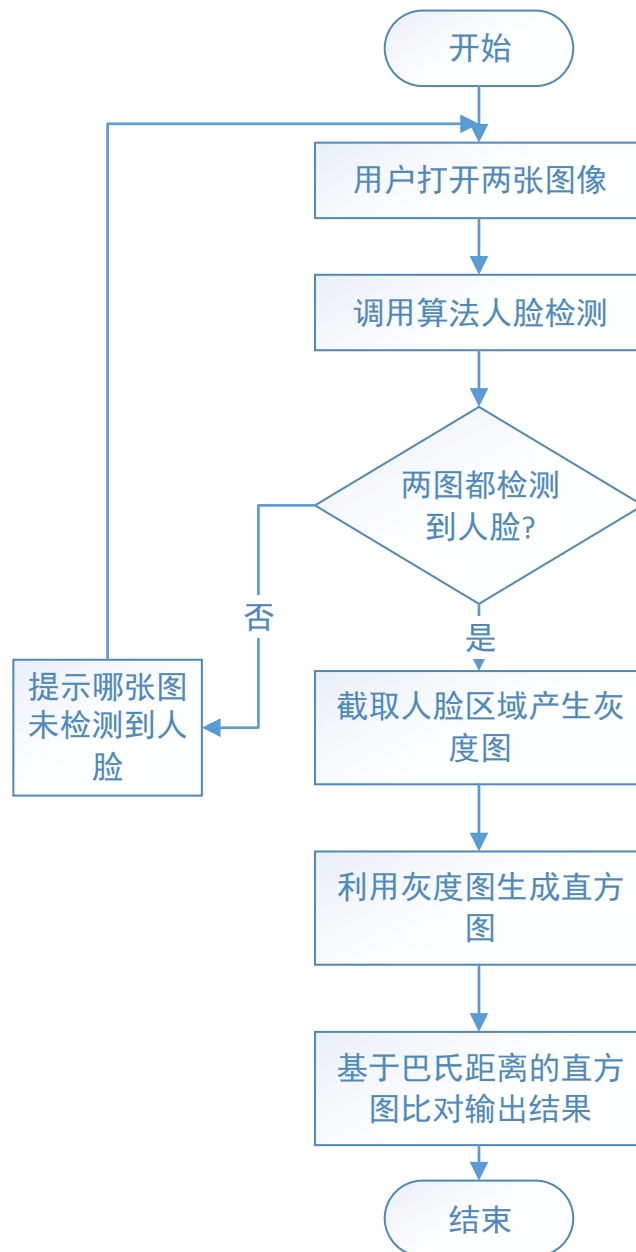


图 4-11 人脸比对模块流程图

1) 调用检测方法截取定位框中人脸区域位图

选择调用基于肤色检测的人脸识别模块接口或者基于 EmguCV 的 Adaboost 算

法人脸检测模块接口获取到人脸区域的位置和大小，截取出位图。设矩形框为 `rect`，则截取操作为 `Image.Copy(rect)` 即可返回需要位图。并且在创建直方图之前，这部分位图需要转换成灰度图像。

2) 创建灰度直方图

此部分直方图的实例化和数据导入都采用开源框架 `EmguCV` 的 API 完成，因为其使用方便快捷并且对直方图处理能力较强，基本能够满足需求。

创建一个空直方图

```
IntPtr Hist1 = CvInvoke.cvCreateHist(1, hist_size,
Emgu.CV.CvEnum.HIST_TYPE.CV_HIST_ARRAY, null, 1);
```

再将图像的灰度值数据导入到直方图中，导入方式采用 `IntPtr` 指向灰度图内存数据并将其传入即可。

如 `CvInvoke.cvCalcHist(inPtr1, Hist1, false, IntPtr.Zero);`

3) 直方图比对

当将两份可以检测到人脸的待比对图像截取出人脸区域灰度图并成功创建灰度直方图时，即可以利用 `Emgu.CV.CvInvoke.cvCompareHist` 接口进行直方图比对，从而得到相似度结果。

调用代码：

设置直方图对比方式为基于 `BHATTACHARYYA`(巴氏距离)计算

```
Emgu.CV.CvEnum.HISTOGRAM_COMP_METHOD histCompareMethod =
Emgu.CV.CvEnum.HISTOGRAM_COMP_METHOD.CV_COMP_BHATTACHAY
A;
```

标准化两个直方图

```
CvInvoke.cvNormalizeHist(HistImage, 1d);
```

调用比对

```
compareResult=Emgu.CV.CvInvoke.cvCompareHist(Hist1,
Hist2,histcompareMethod);// Hist1 和 Hist2 为两个生成的灰度直方图
```

此结果 `compareResult` 的值越小表示两直方图的相似程度越高，它范围在 $[0,1]$ 区间上的 `double` 类型数，因此将其使用 $(1-\text{compareResult}) \times 100$ 转换，结果是百分比表示的相似度再传值到软件界面上的结果 `Label` 上。

4.4 系统实现总结

本章详细介绍了本论文的人脸识别系统的设计与开发实现。用文字、流程图和框架图描述了功能和模块设计；在第 3 节模块实现部分，介绍了各子模块的实现步骤，并且展示了核心代码。系统的开发最终完成了四大功能：基于肤色检测的人脸检测与定位、基于 `Adaboost` 算法的人脸检测与定位、两种算法的摄像头人脸检测与定位、基于巴氏距离直方图比对的人脸图像比对，这四个功能的实现都比较符合预期的设计。

其中通过两种算法之间的比较，两种算法的检测和定位结果准确率较高，但各自有长短处。基于肤色的实现基本不受脸部方位、表情变化、光照强度的影响，能较好保持识别准确率，肤色相似度模型经得住考验，但是定位方法有待改进，目前只能定位单人图像；而基于 `EmguCV` 的 `Adaboost` 方法的实现能够识别多张人脸，且准确率很高，但是受光照、表情变化、脸部朝向、遮挡情况等因素的影响较大。另外人脸比对功能能够高效地得出结果，但是通过测试发现其比对结果值偏高，灰度直方图的方法还有待改进。

结 论

历时数个月的学习、设计和开发，人脸识别算法的研究与人脸识别系统的实现终于完成。归纳识别算法并选定基于肤色检测的人脸识别算法进行深入研究，识别系统则采用基于肤色和基于 Adaboost 两种算法实现，毕业设计包含的两大部分都满足了课题设立时的各方面要求。

本毕业设计的主要工作和成果在于：

对数字图像处理基础进行总结，为识别方法研究和系统的实现作好铺垫；查阅文献资料，对人脸识别算法进行归纳，简略地总结了各类方法的实现算法及其优缺点，方便未来对这些方法的深入学习；通过学习图像模式识别和机器学习的知识，对基于肤色检测的人脸识别算法进行深入研究，分步骤进行剖析，囊括了算法各步骤的理论基础和实现方法；在对人脸识别算法研究学习之后，选定基于肤色检测的算法和 EmguCV 的基于 HaarCasCade 分类器的 Adaboost 方法，进行在.NET 平台上人脸识别系统的开发，最终识别系统实现了四大核心功能，包括基于肤色检测的人脸检测与定位、基于 Adaboost 算法的人脸检测与定位、两种算法的摄像头人脸检测与定位、基于巴氏距离直方图比对的人脸图像比对，这些功能都有比较符合期望值的测试结果。

本毕业设计还需要改善的地方：

基于肤色检测的人脸识别算法中，邻域滤波的降噪模版有待改进，使降低噪点与保证物体边缘完整、图像相对清晰三者间的平衡能够达到更好的状态；人脸定位部分，本文的方法效率虽较高但是适用普遍性并不高，可以将本文的方法提高为基于 Hough 变换的连通区域检测，则其定位效果更加准确与稳定，并且可以定位多张人脸；人脸比对部分，本文采用的灰度直方图比对方法可以将灰度级在人脸区域图像矩阵排列的位置信息进入比对方法，这样其比对准确率将有较大提高。

致 谢

课题从选题到规划再至设计，直到最终完结，经历了很长一段时间。首先，我要特别感谢我的毕业设计导师王献。王老师是一个非常负责任的导师，每周都从忙碌的工作中抽出时间，对我们本科生的毕业设计在立题、规划、学习情况、各模块设计和实现的各个阶段进行指导与解疑，从来都是耐心和蔼，尽心尽力。王老师认真负责的工作态度、实事求是的风格、渊博的知识使每个同学都受益匪浅，在毕业设计的这个学期从始至终都给予了我很大的帮助。在此对老师表示衷心地感谢！

然后在这里要对大学四年里每一位教导过我课程的师长表示感谢，学院老师们严谨的治学态度、勤劳务实的品质是我们打下专业的基石，也是我们生活和学习的精神榜样。交大正是有这些朴实而伟大的老师们才显得不凡，在未来的路上走得更宽更远。感谢老师们，感谢交大。

最后在这里要感谢的是我的大学室友们和同学们，感谢你们的陪伴和学习生活上的所有帮助。最后一个毕业设计的学期也一如既往，大家互相勉励，在专业上做到力所能及的交流与讨论；在寝室生活上友好互助，营造了非常好的和谐氛围，提高了学习工作效率。平时同学们更是相互督促按计划完成每个阶段的学习任务和课题设计，这样使得我不浪费时间，抓住机会努力学习，顺利地进行课题知识的学习、识别系统的实现和论文的撰写。

参考文献

- [1] 史承东. 人脸图像信息处理与识别技术[M]. 北京: 电子工业出版社, 2010.10.
- [2] Qiuyu Zhang, Yanfeng Jin, Zhanting Yuan, Jiawen Hu, Lei Sun, Wenjing Li.
An Improved Adaptively Weighted Sub-pattern PCA Approach for Face Recognition[R]. 6th Distributed Campus Management System Based on SOA 2007.
- [3] Yong Xu,Jing. Local Correlation Classification And Its Application To Face Recognition Across Illumination[R].第五届机器学习与控制论坛. 2006.
- [4] 杨淑莹, 图像模式识别:VC++技术实现[M]. 北京: 清华大学出版社; 北京交通大学出版社. 2005.7.
- [5] 赵立强, 张晓华, 高振波, 张洪亮. 基于 BP 神经网络的主分量分析人脸识别算法[J]. 计算机工程与应用 2007 第 36 期. P44-56.
- [6] 孙继文. 人脸识别算法的研究[D]. 沈阳工业大学硕士学位论文. 2014.
- [7] 苏波. 隐 Markov 模型和神经网络人脸识别算法研究[D]. 电子科技大学硕士学位论文. 2007.
- [8] 孙大瑞, 吴乐南. 基于特征的弹性图匹配人脸识别算法[J]. 应用科学学报. 2002 第 4 期.
- [9] Drew Conway , John Myles White 著, 陈开江, 刘逸哲 ,孟晓楠 译, 机器学习实用案例解析[M]. 北京: 机械工业出版社, 2013.
- [10] 王志良, 孟秀艳, 人脸工程学[M]. 北京: 机械工业出版社. 2008.6.
- [11] 史海成,王春艳,张媛媛. 浅谈模式识别[J], 今日科苑, 2007 年 22 期 P132-139.
- [12] 欧凡, 刘冲, 欧宗瑛. Region Pair Grey Difference Classifier for Face Detection [J], Transactions of Tianjin University 2010 第 2 期 P118-122.

- [13] 丁嵘, 苏光大, 林行刚. 使用关键点信息改进弹性匹配人脸识别算法[J]. 电子学报. 2002(09) P14-20.
- [14] 朱明早, 罗大庸. 模块 2DPCA 的缺陷与改进[J]. 中国图象图形学报. 2009 第 1 期 P33-36.
- [15] 宋加涛. 基于二值边缘图像的眼睛定位和人脸识别[D]. 浙江大学博士学位论文, 2004 .
- [16] 龙伶敏. 基于 Adaboost 的人脸检测方法 & 眼睛定位算法研究[D]. 电子科技大学硕士学位论文, 2008.
- [17] Peter Harrington. Machine Learning In Action[M]. 李锐, 李鹏, 曲亚东, 王斌译. 北京: 人民邮电出版社. 2013.6.
- [18] 冯伟兴, 梁洪, 王臣业. Visual C++数字图象模式识别典型案例详解[M]. 北京: 机械工业出版社, 2012.6 .
- [19] LIN Lan, ZHAO Ge, TANG Yan-Dong, TIAN Jian-Dong, HE Si-Yuan, Illumination Compensation for Face Recognition Using Only One Image[J]. 自动化学报 2013 第 12 期 P2090-2099.
- [20] LU, Guan-ming, ZUO, Jia-kuo. Orthogonal isometric projection for face recognition [J]. 中国邮电高校学报(英文版) 2011 第 1 期 P23-31.
- [21] 徐涛, 冯志全, 吴鹏. 与颜色空间选择无关的肤色检测方法[J]. 计算机辅助设计与图形学学报. 2013 第 1 期 P49-54.
- [22] 崔更申, 杨飞, 黄廷辉. 基于相似度的光照鲁棒性肤色检测[J]. 计算机工程与设计. 2014 第 12 期 P4259-4263.