

中文摘要

摘要：人脸识别技术是当前模式识别和人工智能领域一个重要的研究课题，它的研究跨越了图像处理，模式识别，计算机视觉和人工神经网络等等学科。随着计算机技术的快速发展与商业和军事等应用领域的需求，关于人脸识别技术的研究将会不断的深入和完善，在未来将会应用到日常生产生活的各个方面。

人脸特征提取是人脸识别中复杂且极具挑战性的环节，对人脸的识别率起着至关重要的作用。传统的人脸特征提取方法由于受到光照，姿态等复杂环境的干扰，识别率会产生一定的影响。在最近的研究发展中，提出了许多具有较强鲁棒性的人脸特征提取方法，但大部分都是从矩阵算法角度去改善特征提取的效果来提高识别率。本文研究的重点是利用多目视觉在感知阶段最大程度地获取人脸的信息，并通过构建一个 Agent 框架对多目视觉进行调度和控制来实现提取出比较好的人脸特征。本文做的工作主要有：

- (1) 对传统的人脸特征提取方法进行深入地探讨，主要研究了基于 ASM 的特征提取方法，并对其经典算法进行实现，通过实验总结这种算法的优点和弊端。
- (2) 提出了一种利用多目视觉解决因为角度和姿态等原因引起的人脸信息匮乏的方法，利用多个摄像机从不同的角度去感知人脸的环境，极大丰富了所得到的人脸信息，为下一步人脸特征提取奠定基础。
- (3) 提出了一种基于 Agent 的多目视觉人脸特征提取框架。利用 Agent 的协作性和智能性对每个摄像头进行代理，每个 Agent 通过感知环境得到要代理的人脸姿态的参数，并对环境进行反应，通过一定的规则计算出人脸姿态参数，而多个 Agent 之间通过竞争机制去成为最优的人脸姿态目标，由最优的人脸姿态图像得到最终的人脸特征，从而满足整个 Agent 系统对人脸特征较好提取的要求。
- (4) 用多目视觉系统采集多组人脸图片，利用 ORL 人脸库进行识别率对比试验表明本文方法能较好地每组图片中提取出较好人脸姿态的图片，并从中提取人脸特征，解决了由于角度问题带来的误识别问题，由于 Agent 平台的并行性，在处理时间上也有很大的优势。

关键词：人脸特征提取；Agent 理论；ASM 算法；多目视觉；人脸姿态参数；Agent 交互机制；并行处理

分类号：TP391.4

ABSTRACT

ABSTRACT: Face recognition technology has always been an important research topic in the field of current pattern recognition and artificial intelligence, its research across the image processing, pattern recognition, computer vision and artificial neural network and other subjects. With the rapid development of computer technology, the increasing needs of commercial and military applications, the research of the face recognition will be more and more depth, it's more likely to be applied to all aspects of daily production and life.

Face feature extraction is a complex and challenging issue in face recognition, it has important affect on recognition rate. Because traditional methods of face feature extraction usually affected by light, gaze of the face and other complex environment, the recognition rate may not accurate. In recent research, many robust face feature extraction methods have been proposed, while most of them are improved from matrix algorithm aspect to improve the recognition rate. The research purpose of the paper is to solve the perception problem by multi camera system, and extract face feature with a system of multi camera managed by an Agent framework. The work of the paper includes:

Firstly, we have a deep study of the traditional face feature extraction method, especially the ASM method, and realize the classical ASM algorithm, through the experience of two face feature extraction, we summary the virtues and defects of this algorithm.

Secondly, propose a method of multi camera system which can solve the problem of lack of face information caused by angles and gaze. We use several cameras to perceive human face environment from different angles, which can enrich the face information, and lay a foundation of face feature extraction.

Thirdly, a novel approach was proposed to extract face feature base on Agents with multi-camera system. According to the ability of cooperation and intelligence of the Agent, we can give each camera an Agent, each Agent tries to get the pose parameters of face feature from the environment, also the Agent will react to environment, which will compute the facial pose parameters. Those Agents also compete with each other, in order to become the best facial object. In this way the face feature can be better extracted with the Agent system.

Finally, We collect a couple of face pictures from the multi camera system, and have a face recognition test with the ORL face database which recognition rate experiment shows that the method can better extract the face feature. From the experience we also solve the error identification problem caused by face angles, Due to the parallelism of the Agent platform,

the processing time also have a big advantage.

KEYWORDS: Face feature extraction; Agent theory; ASM algorithm; Multi camera system; Face pose parameters; Agent interaction mechanism; Parallel processing

CLASSNO: TP391.4

致谢

这篇论文是在我的导师阮秋琦教授的认真指导下完成的，阮秋琦教授对科研工作的严谨态度和在学习方法上的耐心引导在学术上给了我很大的影响。开始进入实验室的时候对图像方面知识都不太懂，阮教授不仅在科研环境给我创造了条件，更在学习上对我有很大的帮助，在暑期期间，阮教授还给我们科研实践的机会，参与了实验室的科研项目，在学习生活上都给了我很大的帮助，在这里对阮秋琦老师这两年来教诲和关心表示衷心的感谢。

在实验室工作的这两年，实验室的师兄师姐们也给了我很大的帮助，他们热情的关怀和耐心地指导在心灵和学习上都给了我很大的动力。在这个课题的研究以及这篇论文的撰写中，阮成雄、李小利等学长对我实验设计和研究方向的把握给了很大的帮助，在这里也向他们表示衷心的感谢。

最后我要感谢我的家人，是你们一直以来对我的鼓励支持和关心才让我在学业上能走到今天，是你们对我的教诲让我树立了自己的目标，感谢你们。

序

随着计算机网络的飞速发展和信息化进程的日益加快，公共安全和信息安全越来越显示出其前所未有的重要性。人脸识别技术是当前模式识别和人工智能领域一个重要的研究课题，它的研究跨越了图像处理，模式识别，计算机视觉和人工神经网络等等学科。在安全应用上它广泛应用于国家安全、公安系统和城市的公共安全等各个方面。虽然经过了许多年的发展，但是由于人脸是一个非刚性物体，并容易受到光照、自身姿态和表情等因素的影响，因此要实现一个高识别率、高鲁棒性的全自动的人脸识别系统仍然是一个极具挑战性的课题。

特征提取是人脸识别研究中的最基本问题之一，是人脸识别中复杂且极具挑战性的环节，对人脸的识别率起着至关重要的作用。在这一领域的研究中，提出了许多经典的算法，如线性鉴别分析方法，基于子空间的主分量分析方法，独立成分分析方法以及在基于核技巧的非线性特征提取方法等。对于特征提取问题的深入研究在很大程度上促进了人脸识别领域的发展。

现有的很多人脸特征提取算法都具有较强鲁棒性，但是他们大部分都是从数学方法上改善了特征提取的质量来提高识别率。本文研究的重点是利用多目视觉在感知阶段最大程度地获取人脸的信息，并通过构建一个 Agent 框架对多目视觉系统进行调度和控制。在各个 Agent 的交互和相互合作中通过一系列的行为属性提取出比较好的人脸特征。

本课题受到国家自然科学基金项目 (NO. 60973060) 的资助。

1 绪论

1.1 课题来源和研究意义

本课题是国家自然科学基金项目《基于多目视觉的形体语言感知与识别研究》中的一个重要研究内容。着重研究在智能代理（Agent）的基础上实现多目视觉的协同工作，建立基于智能代理的多目感知框架，在理论和技术上解决形体语言感知中的遮挡和方位带来的误识别问题。

人脸识别技术^[1]是当前模式识别和人工智能领域一个重要的研究课题，它的研究跨越了图像处理，模式识别，计算机视觉，人工神经网络和生物医学等等学科。随着计算机技术的快速发展与商业和军事等应用领域的需求，人脸识别技术研究得到了愈来愈广泛的关注和参与。人脸识别虽然有许多其它识别方法无法比拟的优点，但是直到现在还没有一种很完美的人脸识别系统，人脸识别的准确率目前为止还很难达到我们的要求，对许多场合的应用还不能满足^[2]。虽然对于人类而言可以很容易的从复杂的环境中识别出人脸，但是对于计算机识别系统来说，这是一个巨大的挑战。归根结底是由下面几个因素造成的。

1. 人脸结构的相似性，不同人脸之间的区别并不是很大，所有的人脸都是由几个相同的部分组成，外形结构相似。这是计算机能够自动识别人脸的主要原因，但是这种相似性对于区分不同的个体来实现人脸识别是不利的。

2. 人脸具有极强的不稳定性，首先人脸具有丰富的表情，不同表情下采集的人脸图像会有变化，其次观察人脸的角度不同，得到的人脸的觉图像差距也会很大。另外，光照强度的影响，角度的变化、遮挡等多方面因素都会人脸识别产生影响。

3. 人脸识别属于一个大类别的识别问题，全球人口数量是非常庞大的，而且还在不断的增长中，因此识别的类别数很多，因此识别的难度也会相应的增大。

人脸识别的研究具有很高的理论和应用价值，但是同时又是一项非常具有挑战性的工作，在大类别，复杂的背景，有局部遮挡，人脸角度变化等复杂条件下的人脸识别的正确率，可靠性和稳定性方面还需要很大的改进和深入的研究。人脸识别的一般过程由三部分组成，分别为人脸检测^[3]，特征提取，分类识别，如图 1-1 所示。人脸特征提取是指抽取表示可分离性而言最有效的特征，即从人脸中提取出用于区分不同个体的判别特征。提取要求对于同样的个体具有稳定性，对于不同的个体具有差异性。特征提取是人脸识别中最为重要的一个部分，它的效率直接决定最终的识别性能。到目前为止，人们已经给出了许多种特征提取方法，如主分量分析，线分析以及非线性投影方法等，这

些方法都在人脸识别中被广泛使用并取得好的特征提取效果。特征提取是模式识别中的一个关键问题，在模式识别等领域有着广泛的应用。本文主要针对人脸特征提取进行研究，以此来解决因为角度等问题对人脸识别带来的问题。

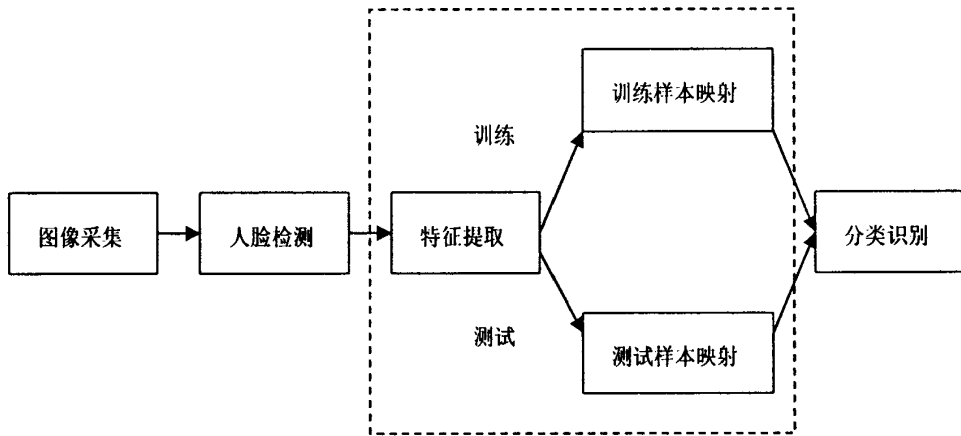


图 1-1 人脸识别流程图

Fig. 1-1 The flow of the face recognition

1.2 研究现状

到目前为止，国内外各个研究机构和人员对人脸识别的研究正在如火如荼地展开，美国等西方国家成立专门从事人脸识别研究的小组，军方，大的国际公司和高校对人脸识别技术的研究使得近些年来人脸识别技术取得了重大的发展。国外比较有名的有：MIT, CMU, UCLA, University of Manchester, UMD 等。国内的中国科学院计算的所，自动化所，清华大学，南京理工大学，北京交通大学等单位都在人脸识别研究方面取得了不少的成果。人脸特征提取在人脸识别中起着不可或缺的作用，特征提取又称人脸描述，是在基于人脸检测定位的基础上进行的人脸各特征提取的过程，也是对面脸进行特征建模的过程。特征提取是人脸识别中最为重要的一个部分，它的效率直接决定最终的识别性能。

在过去的几十年内，研究者们提出了很多用于人脸识别的方法，大致可以分为两类：一类是基于统计学习的特征提取方法，一类是基于知识的特征提取方法^[4]。其中通过基于统计学习提取的特征称为代数特征，基于知识的特征提取方法提取的特征为几何特征。基于代数特征提取的方法又可以分为两类，分别是线性和非线性特征提取方法，线性方法有主成分分析(PCA)，独立成分分析(ICA)，线性鉴别分析(LDA)等，非线性方法有核方法，流形学习方法，张量方法等。

1. 基于代数方法

主成分分析(PCA)的基本思想是利用一组包含了较多信息量的特征去尽可能精确地

表示原始样本的分布情况。Fukunaga 和 Young 等人对 PCA 方法做了深入的研究^[5], 讨论了 PCA 作为线性特征提取方法的稳定性。Kirby 等人^[6]讨论了利用 PCA 进行人脸图像的最优表示问题, 然后 Turk 和 Pentland^[7]讨论了这种表示的物理意义。PCA 有两个方面的优势, 首先它从 2 阶上消除了模式和样本间的相关性, 第二它实现了对原始样本的维数的降低。PCA 只是在最小平方误差意义下给出的模式样本的最优表示, 但是不能从高阶上消除特征之间的相关性。独立成分分析(ICA)^[8]方法是根据最小化相互信息准则或者最大化非高斯线性准则提出的从线性混合的信号中恢复出独立信号源的算法, Bartlett 等人^[9]采用 infoMax 算法来实施 ICA, 提出了利用独立分量分析进行图像表示的两种架构, 这两种架构被众多的研究人员应用于人脸识别问题。线性鉴别分析(LDA)^[10]的基本思想是基于一个与分类相关的特征提取准则, 即 Fisher 准则^[11], 目的是为了选定一组最优的鉴别矢量, 并且使得原始数据在该鉴别矢量上的投影达到最佳的分类效果。近来 Zhuang 等人^[12]提出了逆 Fisher 鉴别分析, 并且成功地应用于人脸识别的应用。逆 Fisher 鉴别分析有以下优势: 1. 原始数据总体散布的矩阵的小的特征值对应的特征向量参与了选择。2. 考虑了类内散布矩阵的零空间信息。

基于核方法^[13]特征提取的主要思想是通过适当的非线性映射将非线性可分的原始样本变换到某一个线性可分的高维的特征空间, 核技术最早是由 V.Vapnik^[14]提出并应用于向量机中。Mika 等人^[15-16]利用核方法将 Fisher 线性鉴别分析进一步推广到了非线性情况, 提出了一种核 Fisher 鉴别方法, 这种方法解决了两类模式的分类问题。Yang^[17]等人提出了完备的核鉴别分析应用框架。徐勇等人^[18]从所有训练样本中选取少量显著的样本, 使得核投影方法的特征提取效率有了很大的提高。与基于核技术的非线性特征提取方法不同, 流形学习^[19]的目的是期望找出产生数据集的内在规律性, 即从观察数据中找出本质, 在高维的输入空间中找出其内嵌的低维流形, 在低维空间中恢复数据集内的集合特征。Seung^[20]提出了感知是以流形方式存在的, 并且通过实验证明了人脑中的确存在稳定的流形。当前比较流行的流形学习方法有 Isomap^[21], LLE^[22], Laplacian Eigemap 等。这些学习方法仅定义在样本上, 但是不知道如何去评价样本, 所以无法应用于模式识别领域。近来, X.He 等人^[23-24]提出了一种局部保持性的投影方法(Locality Preserving Projection, LPP)并成功地将其应用于人脸识别。与 Laplacian Eigemap 方法不同, LPP 是一个线性投影算法, 但是它在某些方面保留了样本点在观察空间内的局部的领域信息。基于 LPP 方法又相继提出了 MFA, LDE 等相关的学习方法。二维主成分分析, 二维鉴别分析和二维 LPP 等方法近些年来提出的针对图像模式的特征提取方法, 该类方法直接在图像矩阵上计算散布矩阵, 矩阵的维数等于图像的行数或者列数, 使得计算起来很方便。Zheng 等人^[25]在分析了基于向量和基于矩阵的线性鉴别方法, 指出根据偏置估计观念, 2DLDA 可能会好于 1DLDA 方法。

2. 基于几何方法

我们的人脸由几个特征器官构成,比如眉毛,眼睛,鼻子,嘴巴等,并且他们在人脸上的位置相对来说是比较确定的,但是同时又不都是完全一样,因此我们每一个人的脸才会不一样。基于几何的方法^[26]要求我们由人脸的特征器官点建立一个相似的并且可以用参数去改变它的模型,在检测的时候,可以规定一条准则来判定这个模型和实际人脸的相似度,只有当满足这个判定条件的时候才会认为这个模型和检测的人脸匹配。虽然几何方法相对会简单一些,但也有它的弊端,它对环境的要求会比较高,比较复杂的环境下这种方法的效果不是很好。Snakes 方法^[27-28]同样使用一个准则去判定是否和实际的检测人脸相似,它使用一条闭合的曲线去检测,这条曲线不断去逼近人脸的形状,并且在逼近的过程中由这个准则来监督,当满足准则条件的时候,表示曲线的形状已经和人脸特征相似。BLEDSOE^[29]利用特征器官之间的关系,比如他们之间的距离等建立了一个人脸识别系统。文献^[30]使用了一个 35 维的人脸矢量,这个矢量可以用积分投影法得到,矢量包含眼睛和眉毛之间的距离,眉毛的弯曲程度,厚度,嘴巴的弯曲程度和厚度和鼻子到嘴巴和眼睛的距离,下巴的形状等等。Cootes 等人^[31]提出了 ASM(Active Shape Model)的方法,ASM 的思想是首先对比较有特点的一类目标进行统计,然后进行建立一个能反应目标的形状在二维空间上的变化规律的模型和一个能反应目标的特征点的灰度分布规律的灰度模型。由这两个模型相互呼应和相互间的调整来进行特征点定位。Yuille 等人^[32]提出了参数化的可变形模板法,这种方法对于人脸特定位有很好的效果。对于人脸中眼睛的检测,该方法根据眼睛形状的先验知识,定义 11 个参数的模型,通过搜索能量参数极小化来确定参数。Hallinant^[33]提出了一种基于统计学检测正面姿态下人的眼睛的算法。该算法采用了一个模板,它拥有不同亮度的两块区域,一块是虹膜区域,另一块是眼睛的白色区域,算法根据分布的不同来构造模型。Pentland 等^[34]扩展了其“特征脸”的概念,提出了“特征脸”、“特征鼻”等算法来检测人脸特征。Reisfeld 等^[35]提出了一种采用一般对称性算子寻找眼睛和嘴巴的算法。该算法首先利用先验知识假设人的整个脸部相对于鼻子的垂线是对称的,并且相应的五官也是这样,在这个前提下,对称算子可以去定位对称性最高的点,并根据这个条件去检测眼睛和嘴巴。

1.3 本文主要研究内容

本文主要讨论了人脸识别中特征提取的方法,首先我们对传统的人脸特征提取方法进行深入地探讨,我们主要研究了基于 ASM 的特征提取方法,并进行算法的实现,然后总结其优点和弊端,利用多目视觉在感知阶段最大程度地获取人脸的信息,根据一定的规则计算估计出每个人脸的姿态,并通过 Agent 框架对多目视觉进行调度和控制以此来实现提取出比较好的人脸特征。这就是我们提出的一种基于 Agent 的多目视觉人脸特征提取方法。

传统的单目和双目视觉感知在计算机视觉中已有多年的研究，主要集中在中心投影摄像机模型指导下的感知研究，在该模型下解决形体语言感知的问题尚有诸多难以解决的限制。因此我们准备从多摄像机系统解决感知问题，从感知阶段最大限度的解决由于遮挡、姿态等复杂环境带来的难点，进而提高感知的精确性，为后续的分析打下良好的基础。在多目视觉的基础上，感知的能力有了很大的提高，如何从更多的信息量中筛选出对我们有用的信息，对信息进行决策，从而得到比较好的，有利于识别的人脸特征，以此满足我们对人脸特征提取方法的优化。本文采用了 Agent 方法，在智能代理（Agent）的基础上实现多目视觉的协同工作，建立基于智能代理的多目感知框架，在理论和技术上解决形体语言感知中的遮挡和方位带来的误识别问题。智能代理研究方案如图 1-2 所示。

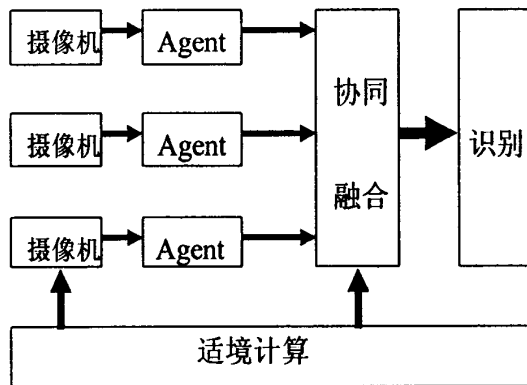


图 1-2 基于智能代理多目视觉协同工作原理框图

Fig. 1-2 The framework of multi camera system based on Agent

本文主要研究了在基于 Agent 的框架中如何协调多目视觉的协同工作，建立一种可行的 Agent 合作通信机制，以及 Agent 之间的交互规则。结合 ASM 人脸特征提取和人脸姿态估计等算法，使得多目视觉在经过 Agent 系统后，提取的人脸特征的准确性得到很大的改善，并通过识别率体现出来，同时 Agent 系统的并行特性使得处理的速度得到好的提高。

1.4 论文的章节安排

本文一共分为 5 章，各章的主要内容如下：

第一章绪论。本章主要介绍了人脸特征提取的研究背景、本文研究的主要内容和意义。详细介绍了在人脸特征提取领域各种算法的发展。

第二章 ASM 特征提取方法。本章主要介绍了基于 ASM 的特征提取方法，对 ASM

理论进行了详细的描述，并对经典的 ASM 算法进行实现，通过实验分析总结其优点和弊端。

第三章 Agent 系统理论。本章主要介绍了 Agent 体系的基本理论和多 Agent 体系的结构。对 Agent 的特性和体系结构进行了总结，并分析了不同体系结构的优缺点。然后分析了在多 Agent 系统中的几个核心的问题，如 Agent 之间的通信语言的制定，Agent 之间传递消息的机制和多个 Agent 之间如何协调合作来达到共同的系统目标。

第四章基于 Agent 多目视觉人脸特征提取系统。本章主要介绍了如何实现基于 Agent 的多目视觉人脸特征提取系统。首先详细介绍了开发 Agent 的平台 JADE。然后介绍了整个系统的大致流程图，各个不同功能的 Agent 如何构造，他们之间如何交互，交互的规则等关键问题。本章还介绍了 Agent 系统下的关键代码并展示了系统的实验界面。

第五章实验结果与分析。本章主要用多目视觉采集了多组图片进行实验。利用 ORL 人脸库进行识别率计算，通过和单目视觉的识别率进行对比得到结论，在准确率和速度上进行分析。

第六章总结与展望。针对本文的研究内容进行总结，并且针对以后的研究方向和内容予以展望。

2 ASM 特征提取

2.1 引言

人脸面部的特征点，比如眉毛，鼻子和嘴巴的定位是人脸识别，模式识别中的一个重要研究课题，是后续的姿态估计，人脸识别，人脸跟踪和三维重建等课题研究的基础。近年来对人脸特征点提取的研究正在如火如荼地展开。人脸特征提取是指利用计算机技术在人脸中自动定位出人的眼睛，嘴巴，鼻子，耳朵等精确的位置并且能描述出其形状的特征，以此来确定输入的人脸模式。在目前的研究中，研究人员重点的研究方向是对特征点定位的准确性和定位的鲁棒性的提高。目前的人脸特征定位技术还很薄弱，在复杂的环境下，例如人脸表情的变化，周围环境的影响，人脸姿态的变化等都会使人脸特征定位的很困难。

经典的主动形状模型^[36] (Active Shape Model, ASM)最早是由 Cootes 和 Taylor 提出的一种基于统计学的并且灰度和形状可分离的模型。简单来说，经典的 ASM 模型的特点主要表现在它对一维轮廓上每一个特征标记点运用了马哈拉诺比斯距离，并且提出了一个线性点分布模型。在经典 ASM 模型方法中，训练决定了轮廓特征点的分布模型。ASM 最初主要用于对医学图像的结构定位。由于 ASM 的一般性，它可以为各种不同的模式建立形状模型。

ASM(Active Shape Model)方法是通过训练集中所要进行描述的各种目标形状进行统计，以此来建立一个能够反映这个目标的形状变化情况和灰度分布情况变化规律的模型。在进行目标搜索的时候，首先用训练中的局部灰度模型进行搜索，然后利用形状模型对之前灰度模型搜索到的区域进行相似性比对，利用一定的规则对比对的结果进行判定，当其满足判定的条件时，即这时是理想的匹配，当不满足时，表明匹配不理想，此时对不合理的形状进行调整，在调整的时候注意保证形状在统计意义上的合理性，通过多次的迭代循环，可以得到理想的匹配。利用 ASM 我们可以得到目标图像的轮廓，即使在目标图像由于各种原因会有一定变化的时候，只要保证所遇到的变化的目标模式包含于我们的训练库中，此时 ASM 也可以保证能对目标进行准确地定位。

ASM 的优点在于它能够对我们所输入的各种各样的模式进行模式的建立，这是由于我们所需要的关于模式的知识都可以在训练的过程中得到。在训练的过程中，我们需要对模式进行手工的标定，这虽然会使我们的工作变得复杂一些，但是这种方法能够对各种不同的模式来进行建模，尤其是脸部的眼睛，眉毛，鼻子等不是很规则的形状。和变形模板相比，主动统计对象模型的优势在于主动统计对象模型中的所有参数均是从训

练的过程中得到的,是统计参数,各个参数之间的相互关系十分明确,而变形模板中的参数都是人为自己设计的,并且主动统计对象模型的匹配也是通过训练学习得到的。主动统计对象模型的方法属于基于统计学方法,并且得到了广泛的应用。

在对主动统计对象模型的研究中,研究人员主要致力于对主动统计对象模型方法两个方面的改进,即对主动统计对象模型方法中的特征的扩展,还有就是对收敛效率的提高。T.F.Cootes^[37]等在 ASM 的基础上进行了改进和扩展提出了一种 AAM(主动外观模型)来改进 ASM 在定位方面的准确性。AAM 在继承了 ASM 特性的同时还增加了一种灰度纹理模型,这两种模型相结合组成了一个组合的模型。在进行特征点定位的时候,我们通过改变这个组合的参数就可以来获得形状和纹理模型,图像的纹理可以从形状模型提取,最优匹配的条件可以设定为模型纹理和图像纹理之间差值的最小值。由于在 AAM 中加入了图像的纹理信息,因此 AAM 的效果会比 ASM 好,但是同时由于更容易受到光照的影响,速度上会有劣势。在提高 ASM 方法的精确性和速度方面,T.F.Cootes 等^[38]提出了 MRAS (Multi-Resolution Active Shape Models) 的方法。MRASM 方法首先会在比较粗糙的图像中寻找目标,并且在寻找的过程中会有很大的跨度,即对图像先有一个大致的扫描,在寻找的过程中逐步地减小搜寻的范围,使得越到后来,搜索的图像范围愈发清晰,因此定位的准确性也会提高,在多次迭代后会逐渐将制定的模型定位到图像中的目标位置。这种方法的另一个好处就是即使开始的形状和要搜寻的目标形状相距比较远,由于在搜寻过程中是跨度式搜寻,因此即使初始形状位置距离目标较远,MRASM 也会很快速地定位。

GHamarnah 等^[39]将 Snake 模型加入到 ASM 方法中,由于 snake 模型能产生光滑、不间断边界的优点,在分割人的左心室图像的时候,利用 ASM 的能产生于和训练集相似的目标形状的优点,使得分割的效果得到很大改善,Snake 模型对约束的光滑性使得这种方法的准确性相对传统 ASM 方法而言得到了很大的提高,但是在处理速度上会有一些的损失。

Zuo^[40]等用 Haar 特征去检测是否满足纹理的相似,用梯度变换初始化模型,这两种方法的结合提高了在搜索的过程中匹配的精度。T.A.Faruque 等利用主动统计对象模型方法检测嘴唇的轮廓,并且将彩色信息加入进去,在嘴唇轮廓描述方法它使用了五条曲线,这使得参数的维数额达到了降低,匹配的效果得到了改善。将彩色信息加入 ASM,并用五条曲线代表嘴唇的轮廓,大大降低参数的个数,获得了更好的匹配效果。Yan^[41]等提出了 TC-ASM(Texture-constrained Active Shape Model),在这种方法中,由于使用了全局的纹理信息,因此在对形状模型进行约束的时候,不会使得模型产生不合理的形变。Tu^[42]等使用 Fisher 特征代替 PCA 分析得到的统计特征,这种替换使得当样本数量很小的时候,分类器也会有比较良好的性能,并且在搜索的准确性和速度方面也有很好的性能。Jiao^[43]和 Zhan^[44]等在纹理建模时分别使用 Gabor 小波特征和 Haar 特征,由于小波

变换包含的丰富的纹理信息，因此建模的效果得到很大的改善。Zhao 等在进行目标搜索的过程中引入了局部纹理信息的加权值，在此引导下，这种方法对边缘形状的定位比传统的 ASM 方法要好。

2.2 建立 ASM 模型

为了定位到一个我们感兴趣的结构，我们首先必须为这个结构建立一个模型。为了建立一个统计模型的外观，我们需要一系列典型实例的注解图像。我们必须首先决定一系列合适的标记点，这些标记点描述了我们所要搜寻的目标图像的形状并且这些标记的特征点在每个训练的图像中都能找到。

2.2.1 特征点标记

对于特征点标记来说比较好的选择就是指向目标边界清晰的角落，比如在边界连接处的 T 形交叉点和很容易找到的生物特征点。然而，这些比较好的特征点远远达不到我们需要的数量来达到我们对目标形状的一个大致的描述。因此，我们增加了一些在我们定义的比较好的特征点之间沿着边缘等间距排列的点(如图 2-1 所示)。

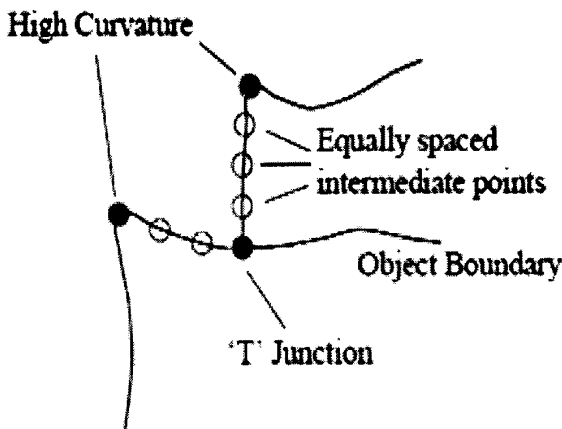


图 2-1 定义的比较好的特征点

Fig.2-1 The definition of Good landmarks

为了描绘出目标的轮廓，我们必须还要记录那些标记点如何在图像中形成边缘的连接定义。这就允许我们通过一个给定的点来决定边界的方向。假设某条曲线上的标记点标志为 $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ 。对于一个二维图像来说，我们可以将这 n 个标记点表示为 (x_i, y_i) ，举一个简单的例子，对 n 个标记点需要 $2n$ 个元素的向量 X ，其中

$$X = (x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n)^T \quad \dots (2-1)$$

如果我们有 s 个训练样本，我们用类似上面的向量 X_j 来生成样本 s 。在我们对这些

向量进行统计分析之前, 保证所要描述的模型在同一个坐标框架内是十分重要的。目标的形状一般相对于它的位置, 方向和大小来说是独立的。就好比一个正方形, 无论将它翻转, 改变大小还是移动它, 它依旧是一个正方形。

Dryden 等介绍了如何将一系列训练的样本形状在同一坐标框架内排列, 将每一个样本形状翻转, 平移和改变大小使得样本形状距离和均值距离之间差的平方的和 ($D = \sum |x_i - \bar{x}|^2$) 最小。

2.2.2 形状模型

假设我们现在有在同一坐标框架内排列的 s 个由点 x_i 组成的集合, 这些向量在他们所在的二维空间中形成某个分布。如果我们可以对这种分布进行建模, 我们就可以生成新的类似于原始训练样本的例子, 并且我们可以来检查新的形状是否是合理的例子。

为了使问题简单化, 我们首先需要将二维数据的维数降低到某个我们比较容易处理的层次, 一种比较有效的方法就是在数据分析中引入主成分分析方法(PCA)。数据在 $2n$ 维空间中形成云状点, 尽管需要在一个 $(2n-4)$ 维的复制空间中排列这些点所在的位置。PCA 计算这些云状点的主轴心, 并且允许我们使用一个少于 $2n$ 个参数的模型来近似任何一个原始的点。PCA 算法如下:

1. 计算训练集中的平均形状和协方差:

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N X_i \quad \dots (2-2)$$

$$S = \frac{1}{N-1} \sum_{i=1}^N (X_i - \bar{X})(X_i - \bar{X})^T \quad \dots (2-3)$$

其中, N 表示训练集中图像的个数, X_i 是第 i 个训练图像的形状向量。

2. 计算特征值和特征向量

计算得到 S 的特征值和特征向量, 然后将特征向量按照对应的特征值的大小进行排列: $\phi_1, \phi_2, \dots, \phi_n$, 他们的特征值为: $\lambda_1, \lambda_2, \dots, \lambda_n$, 其中 $\lambda_i \geq \lambda_{i+1}$, 我们只要前 t 个的特征值, 保证 $\sum_{i=1}^t \lambda_i / \sum_{j=1}^n \lambda_j \geq 98\%$ 。

根据 PCA 的相关算法, 可以计算得到一个矩阵 $\Phi = \{\phi_1, \phi_2, \dots, \phi_t\}$ 。因为较大的特征向量 ϕ_i 已经体现了一些重要的形状, 因此和它相对应的前面的 t 个特征向量完全可以表示大多数形状向量 X 。

如果我们在数据中引入了 PCA, 那么我们可以对任何一个样本集 X 进行近似,

$$X \approx \bar{X} + Pb \quad \dots (2-4)$$

其中, $P = (P1 | P2 | \dots | Pt)$, 它包含了协方差矩阵的 t 个特征向量, b 是一个 t 维的

矢量，定义为：

$$b = P^T (X - \bar{X}) \quad \dots (2-5)$$

矢量 b 定义了一个可变形模型中的一些参数，通过改变 b 中的元素，通过公式(2-4)我们就可以来改变形状， b 中第 i 个变化的参数 b_i 在训练集中由 λ_i 表示。我们规定生成的模型如果和变化的参数 b_i 相差在 $\pm 3\sqrt{\lambda_i}$ 以内，则就可以认为它和训练样本集中的例子相近似。

如果是某个模型的变化正好相对于第 i 个参数 b_i ，那么我们就称它为第 i 个模型。特征向量 P 定义了一个翻转的坐标框架，其中排列了原始的样本形状矢量，矢量 b 定义了框架内的点。

图 2-2 是从 300 幅标记的人脸训练得到的形状例子。每一个图像都由 133 个点标记。形状模型有 36 个参数，可以解释训练集中标记位置 98% 的变化。图 2-3 显示了在和均值 ± 3 的标准偏差内分别改变前三个形状参数的效果，其余的参数都设置为 0。这些模式解释了由于三维姿态的改变引起的全局变化，比较不显著的模式引起的变化更小，更加局部。获得的这些模式在一定程度上和我们要建立的参数模型是类似的，比如摇头，点头和表情的改变。然而，由于他们是从训练集的统计的数据中直接得到的，因此他们不能用一种明显的方式将形状的变化分离出来。

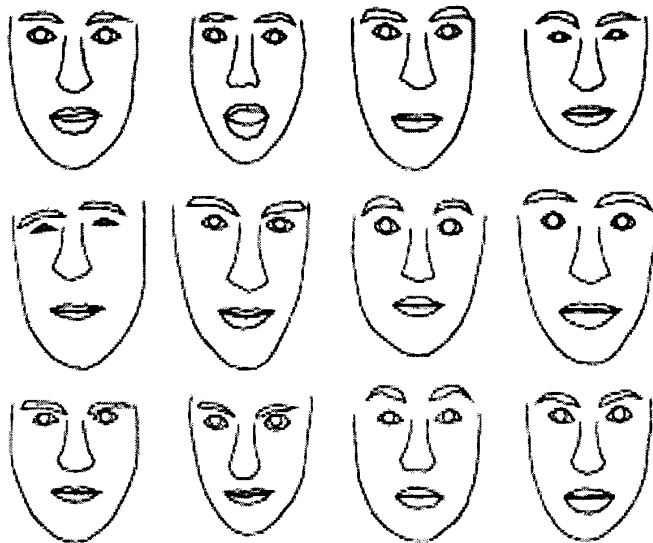


图 2-2 人脸训练集得到的形状例子

Fig.2-2 A sample of face training set

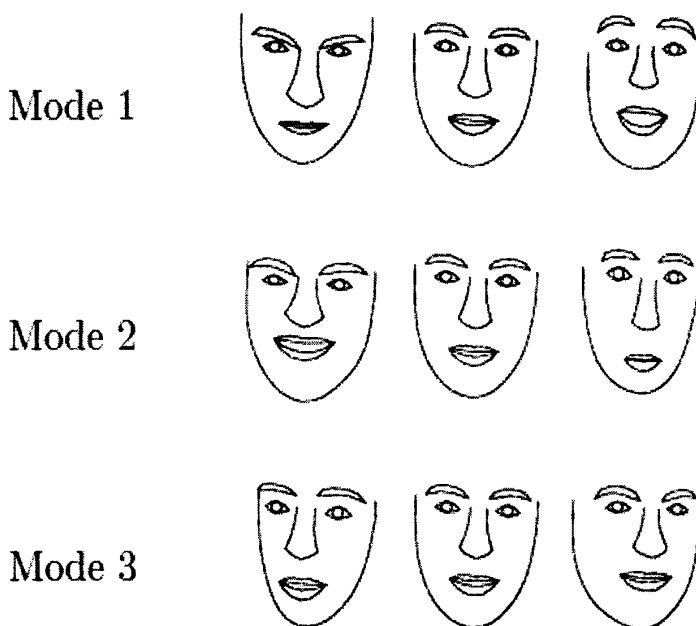


图 2-3 改变前三个形状模型参数后的效果图

Fig.2-3 The picture of changing the first three model parameters

某一个特定值的形状向量 b , 与其对应的点在翻转的空间中由 P 表示, 由此可以对应一个形状模型。我们可以用变换公式将这个模型从模型坐标框架转换到图像坐标框架。并且这种转换是一种典型的欧几里得转换, 转换后将在图像中定义模型的位置 (X_i, Y_i) , 方向 θ 和大小 s 。在图像中的模型的点的位置 X 由式(2-6)给出:

$$X = T_{X_i, Y_i, s, \theta}(\bar{X} + Pb) \quad \dots (2-6)$$

其中, 函数 $T_{X_i, Y_i, s, \theta}$ 表示了旋转了 θ 角度, 改变了大小 s 和偏移了 (X_i, Y_i) , 举个简单的例子, 我们将它应用到一个点 (x, y) 上,

$$T_{X_i, Y_i, s, \theta} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} X_i \\ Y_i \end{pmatrix} + \begin{pmatrix} s \cos \theta & -s \sin \theta \\ s \sin \theta & s \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad \dots (2-7)$$

假设我们希望找到一个最好的姿势(位置, 大小, 方向), 和形状参数, 能够将一个模型实例 X 匹配到一个新的图像点集 Y 中。当相关的模型和图像点集的平方的和最小的时候满足条件, 即使得式(2-8)最小。

$$|Y - T_{X_i, Y_i, s, \theta}(\bar{X} + Pb)|^2 \quad \dots (2-8)$$

下面介绍一种可以达到上述目标的简单迭代方法:

1. 初始化形状参数 b , 都设为 0。
2. 使用式 $X = \bar{X} + Pb$ 生成模型的点的位置。

3. 找到那些可以将模型点 X 最好地排列到现在找到的点 Y 中的姿势参数。
4. 使用 T 的反变换(式(2-9))将 Y 映射到模型坐标框架中。

$$y = T_{X_i, Y_i, s, \theta}^{-1}(Y) \quad \dots (2-9)$$

5. 将 y 映射到 \bar{X} 的切线空间:

$$y' = y / (y \cdot \bar{X}) \quad \dots (2-10)$$

6. 将模型参数更新到和 y' 想匹配。

$$b = P^T (y' - \bar{X}) \quad \dots (2-11)$$

7. 如果不收敛, 返回步骤 2。

2.3 ASM 搜索

2.3.1 算法描述

给定一个开始比较粗糙的逼近, 一个模型的实例就可以适应调整到一幅图像中, 通过给模型选定一系列形状参数, 可以在目标集中的坐标框架中定义目标的形状。我们可以在图像框架中通过定义位置, 方向和大小来建立一个模型实例 X 。比如现在有一张没有做过标记的人脸图片, ASM 将自动利用训练都得到的模型去匹配它。在利用 ASM 搜索和匹配人脸的时候, 我们需要先确定人脸的大概的位置, 我们将这个位置称为初始位置, 并且对这个位置的定位的精确性会对后面的工作有着很大的影响, 当定位不准确时, 即初始位置不恰当, 往往会找不到人脸, 后续的工作也就无法进行。因此在开始搜索之前, 我们最终可能会搜索不到人脸, 也就无法完成匹配, 所以在开始搜索之前我们有必要对人脸进行手工标定。

ASM 搜索算法:

- 1 在图像中搜寻每个点 X_i 的周围区域, 找到和 X_i 最相匹配的点 X_i' 。
- 2 更新参数 (X_i, Y_i, s, θ, b) 来最好地适应新找到的点。
- 3 对参数 b 进行条件约束来使得形状的合理性 $(|b_i| < 3\sqrt{\lambda_i})$ 。
- 4 重复以上步骤直到收敛。

在实践中, 我们通过模型点沿着轮廓观察模型的边界(图 2-4), 如果我们希望模型的边界和某一个边缘相关, 我们可以简单地将最强的边缘(包括方向, 如果已知的话)设定为轮廓的边界。新的位置也就是模型点的新的建议的为位置。

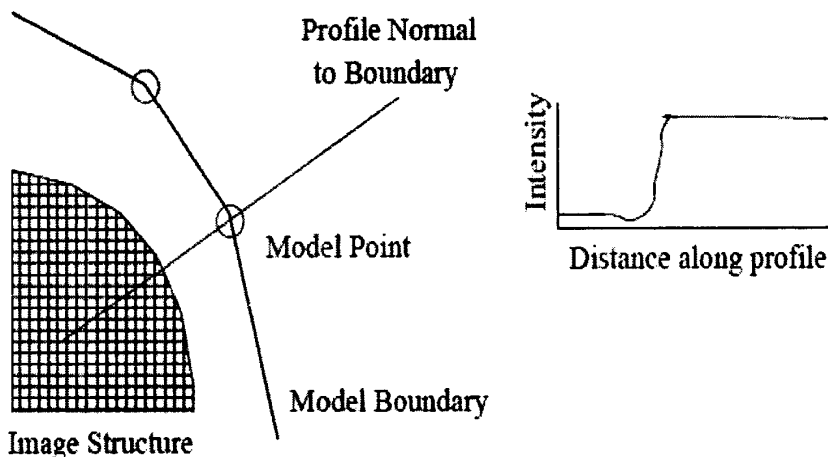


图 2-4 每个模型样本点沿着轮廓到边界

Fig.2-4 At each model point sample along a profile normal to the boundary

然而，模型点并不是每次都落在最强的边缘上，它也可能会落在比较弱的边缘或者其他的图像结构上。Cootes 提出了一种的方案，主要方法是在样本训练的学习中找到每个点的灰度分布模型，在建立形状模型的同时也建立局部灰度模型。局部灰度模型的建立在手工标定之后，它是指在某个点周围的灰度分布情况。

2.3.2 局部灰度模型

在进行标记之后，我们将标记的点用线连接起来，这样就可以得到一个轮廓。沿着每一个特征点的外轮廓在该点的法线方向进行灰度信息统计就可以得到局部灰度建模。以此来找出相应的特征点在这个区域的灰度分布情况。

要得到灰度模型我们首先要在每一幅训练的样本中以标记点为中心选取一定的像素(n_p)，沿着中心点的法线方向，使用标记点的灰度值形成一个向量 h_{ki} ：

$$h_{ki} = \left(h_{ki0}, h_{ki1}, \dots, h_{ki(n_p-1)} \right) \quad \dots (2-12)$$

其中，k 表示第 k 幅训练图像(k=0, 1, ..., N-1)，i 是指所进行标记图像中的第 i 个特征点(i=0, 1, ...N-1)。其次，我们需要计算灰度在法线方向的倒数值 dh_{ki} ，并求得其归一化的值 g_{ki} (如式(2-13)h 和(2-14)所示)。

$$dh_{ki} = \left(h_{ki1} - h_{ki0}, \dots, h_{ki(n_p-1)} - h_{ki(n_p-2)} \right) \quad \dots (2-13)$$

$$g_{ki} = \frac{dh_{ki}}{\sum_{q=0}^{n_p-2} |h_{ki(q+1)} - h_{k iq}|} \quad \dots (2-14)$$

我们计算训练集中的 N 幅图像的平均值(2-15),并求其协方差矩阵(2-16),这样我们就能为每一个点建立一个局部灰度模型(如图 2-5 所示)。并且这一标准化灰度导数向量在整体上满足高斯分布。

$$\bar{g}_i = \frac{1}{N} \sum_{k=1}^N g_{ki} \quad \dots (2-15)$$

$$S_i = \frac{1}{N} \sum_{k=0}^{N-1} (g_{ki} - \bar{g}_i)(g_{ki} - \bar{g}_i)^T \quad \dots (2-16)$$

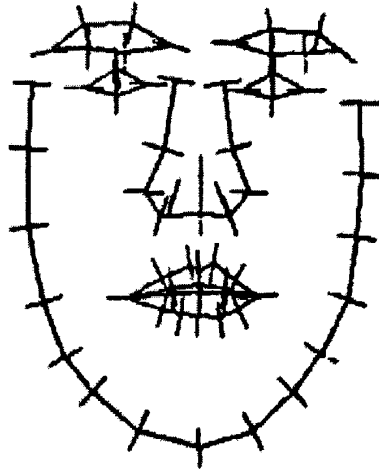


图 2-5 各标记点法向轮廓灰度模型

Fig.2-5 The outline gray model of the normal all label points

2.3.3 最佳模型匹配

我们对图像初始位置的估计一般和被建模的图像的形状有一些差别,因此这就需要我们对估计的初始位置进行调整来找到模型特征点在图像中的精确的位置。在调整的时候,我们可以利用特征点周围的局部灰度信息,沿着特征点法线的方向,分别向着特征点的两侧方向进行搜索,两个方向搜索长度为 $L(L > n_p)$ 的像素点,即一共有对其进行变换处理。搜索的时候,将其对应的平均模型沿着长为 L 的像素点进行比较,一共有 $(L - n_p + 1)$ 种可能。在图像中进行搜索的时候,当出现一个新的点灰度向量 g_s ,我们可以用马氏距离函数来描述这个点的匹配度,马氏距离的表达式为式(2-15)。 $f(g_s)$ 越小的点表示这个点成为最匹配的新位置的可能性就越大。搜寻的流程图如图 2-6 所示。

$$f(g_s) = (g_s - \bar{g}) S^{-1} (g_s - \bar{g})^T \quad \dots (2-15)$$

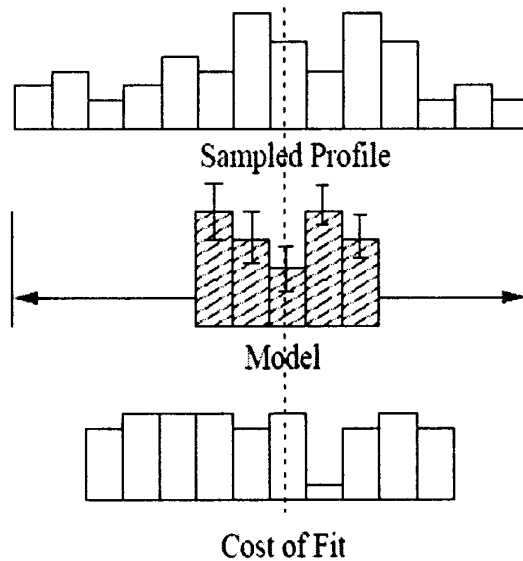


图 2-6 沿样本轮廓搜索局部纹理模型的最佳匹配

Fig.2-6 Search along sampled profile to find best fit of grey-level model

这个搜索的流程对每一个模型像素点都要进行，给出每个像素点的可能的信的位置。当所有的可能像素点都完成了一次新位置的搜索后，就可以得到一个新的模型 Y ，对于新的模型，还需要利用之前建立的形状模型去计算一个与新模型匹配度最好的模型，这里我们可以利用 2.2.2 中的迭代算法。

2.3.4 多分辨率框架

同一幅图像可以用不同的分辨率去表示。我们可以通过降低图像的分辨率来降低阵列的大小，但是这样一来会损失掉一些图像的信息。一般来讲，我们可以在比较低的分辨率下就对图像的特性进行计算，然后再在高的分辨率下对感兴趣的区域进行更加向下的计算。图像的层级表示得到了很广泛的应用，比较典型的有金字塔、四叉树方法等等。在进行特征点匹配搜索时，由于初始定位和图像自身的影响，特征点在候选的最佳匹配中会出现一定程度的误差，如果出现的误差比较大就会影响到搜索的结果。基于这方面的考虑，我们希望能将搜索的范围扩大，这样就可以增大需要的理想匹配点在搜索的范围内。但是这个扩大的范围得有个界限，如果范围太大，则会将那些与理想匹配点模型相似的点包含进来，这样会降低搜索的准确性。Cootes 等人在这些考虑的因素下提出一

个多分辨率搜索框架。

图像金字塔是一种简单并且有效的使用多分辨率来解释图像的方法,对一幅图像而言,将图像集合按照分辨率逐渐降低并且以金字塔形状排列就形成了图像的金字塔。在金字塔的底部是感兴趣的图像的高分辨率表示,底部是低分辨率的近似,金字塔越往上,图像的尺寸和分辨率就会越小,图像的金字塔模型如图 2-7 所示。

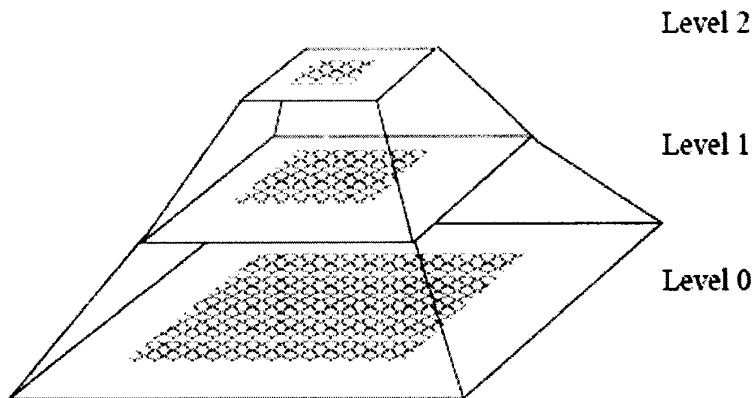


图 2-7 高斯分布的图像金字塔模型

Fig.2-7 A gaussian image pyramid is formed by repeated smoothing and sub-sampling

ASM 在搜索的时候采用多分辨率框架,它可以提高搜索的速度,减少因为错误的图像结构带来的影响,提高了算法的鲁棒性。对每个训练或测试图像,首先需要构造一个高斯金字塔模型。基本层的图像(level 0)即是原始图像,接下来的一层(level 1)通过对原始图像进行高斯平滑滤波(滤波的方法有邻域平均可生成平均值金字塔)。每一维所得到的图像的像素的个数都是原来的一半,只有同样的才做进一步的滤波和降采样得到后面的每一层。在高斯金字塔的每层进行模型训练时,在每一个边界点沿轮廓的法线方向构造局部灰度模型。每一层的轮廓模型一般都会采用相同个数的像素点。L 层上像素的大小是对应基本层(level 0)像素大小的 2^L 倍,因此在相对高层次的模型上,由于每个点调整的距离比较大,模型可以迅速地定位到目标物体周围,越高层的模型描述了图像越大的部分,如图 2-8 所示。

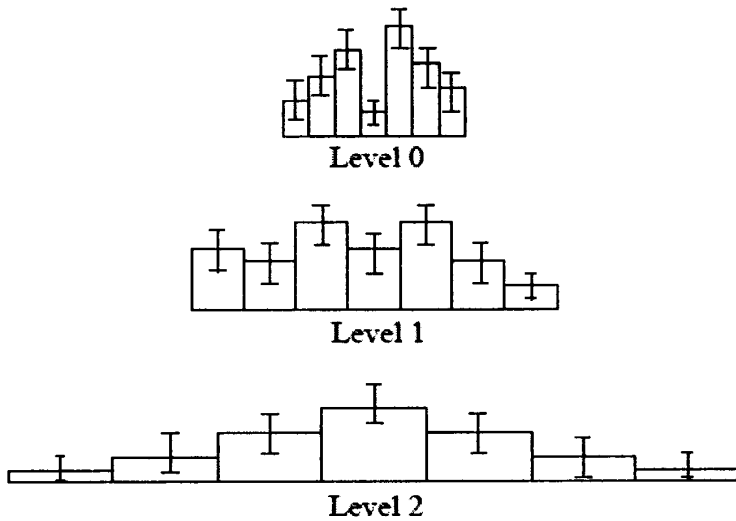


图 2-8 轮廓中的边界点在不同层次的金字塔模型的体现

Fig.2-8 Statistical models of grey-level profiles represent the same number of pixels at each level

在金字塔模型中，是按照自顶向下的过程进行搜索的。当搜索到某一次层时，我们必须定义某个规则，即搜索什么时候终止并转移到下一层搜索。比如可以设定当总数点的 95% 达到了匹配位置(在轮廓法线中心长度一半以内)时，此时停止在本层的搜索并转下一层。

以同样方法对每一层进行搜索直到收敛。当在最后一层 (level 0) 得到收敛时，整个搜索停止。整个多分辨率 ASM 搜索算法如下。

多分辨率 ASM 搜索算法：

1 设定层数 L 的值， $L = L_{MAX}$ 。

2 当 $L \geq 0$ 时

(a) 首先计算金字塔第 L 层图像形状子空间模型点的位置，并且将平均形状作为初始形状 $S_i = 0$ ；

(b) 对 S_i 上每个关键特征点的两侧轮廓法线上进行搜索，搜索像素点个数为 n_s 个，作为候选点，其中具有最小马氏距离的那个将作为特征点的新位置。

(c) 对模型参数进行更新，将轮廓定位到一个新的位置，形成新的模型。

(d) 当超过 95% 的点达到匹配位置((在轮廓法线中心长度一半以内)时，或者已经在这一层上达到了最大的迭代次数，继续，否则返回 2(a)。

(e) 当 $L > 0$ 时， $L \leftarrow L - 1$ 。

2.4 实验结果及分析

图 2-9 和图 2-10 是我们利用传统主动形状模型算法对系统采集的两幅幅目标图像面部的关键特征点进行定位的实验结果。ASM 在样本训练集中由于得到的平均形状模型较好涵盖了多种人脸几何外形, 因此定位的精确性比较高, 如图 2-9 所示, 脸部外轮廓的各个特征点的定位都比较精确。但是同时又由于它过分依赖平均形状模型的初始化位置, 如果仅仅只有整体形状模型的约束, 将会导致部分特征点, 比如眼角和嘴角的定位结果偏离在算法的迭代过程中精确定位的位置, 算法的精确性将会受到影响。由于光照对灰度值影响比较大, 特征点周围的纹理模型是由灰度梯度信息建立的, 因此也会受到影响。如果目标搜索图像的光照环境和训练样本集中的图像的光照条件有很大的差距, 那么我们就不能使用原来的纹理模型去指导特征点匹配。当人脸图像相对于正面人脸偏转角度比较大时会丢失部分特征点信息, 也会导致无法匹配, 算法也因此失败。如图 2-10 是我们采用传统 ASM 算法对有角度偏转的目标图像进行关键特征点定位时, 一些关键的特征点(比如眉毛, 眼睛和其他脸部外轮廓的特征点等)的定位出现了偏差。

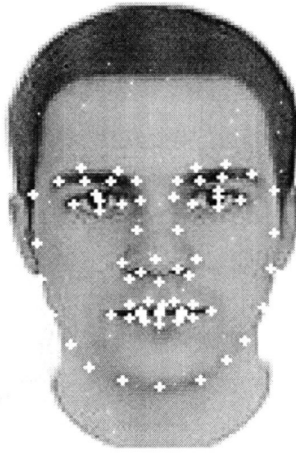


图 2-9 ASM 人脸特征点定位

Fig.2-9 Face feature location by ASM



图 2-10 ASM 有角度偏转人脸特征点定位

Fig.2-10 Face which has an angle feature location by ASM

2.5 本章小结

本章着重介绍了 ASM（主动形状模型）算法，首先介绍了特征点标记，形状模型等基础理论，接着详细描述了主动形状模型的搜索算法，并介绍了整个搜索的过程（计算关键特征点的新位置、计算形状参数以及对形状参数的更新）和 ASM 多分辨率框架。主动形状模型是一种基于目标先验知识的特征提取方法，它采用了自顶向下的搜索和定位方法。通过对一系列目标的训练样本集标定，就可以使用那些标记点向量去描述物体的形状。PCA 方法可以对样本进行统计分析，并建立一个关于目标和形状模式的初始模型，然后根据这个模型在目标图像中利用迭代算法进行搜索，在搜索的过程中更新模型的形状参数，以此来使得选取的实例形状的标记点逐渐拟合到目标的实际边缘上面，这样就最终就可以准确地目标图像中定位到目标形状。

我们对系统采集的图片进行了实验，并对实验结果进行了分析，总结了影响 ASM 算法精确定位到特征点的相关因素，以此来引入从感知阶段解决因为角度，光照等问题带来的特征提取难点的方法。

3 Agent 系统理论

3.1 引言

人工智能的发展使得我们在先验知识的基础上能通过有机推理使应用程序具有灵活反应的特点,这其中最典型的的就是专家系统。随着计算机网络技术的发展,应用程序之间的交互正在从单一的集成式向着分布式转化。中间件技术的发展使得应用程序之间可以以共同的方式来获得服务,这样一来就使得分布式下的软集成成为可能。在计算机领域,我们可以将 Agent 看成是一种个人软件助理,它是一种在分布系统或者相互协作的系统中能持续自主地发挥其自身特性的计算实体,并且是一种具有智能的实体。它具有一定知识、思维和能力,拥有特定的目标,并且和环境能够进行相互的感知和反应,和其它 Agent 能进行合作。Agent 的基本思想是使软件实体模仿人类社会的行为和组织形式,协作关系,进化机制以及认知和解决问题的方式。

我们可以从人工智能和分布式计算技术两个角度来分别研究 Agent, 他们具有各自的重点,对于人工智能来说,主要研究多 Agent 系统(Multi-Agent System, MAS),而分布式方向则研究了分布式问题求解(Distributed Problem Solving, DPS)。随着网络技术包括分布式计算的发展,有关 Agent 的研究正在日益受到人们的重视,它已经成为人工智能领域的一个研究热点。

3.2 Agent 特性和结构

3.2.1 Agent 特性

Agent 的概念由 Minsky^[45]在其 1986 年出版的《思维的社会》一书中提出。他认为社会中的某些个体经过协商之后可求得问题的解,这些个体就是 Agent。他还认为 Agent 应具有社会交互性和智能性。在分布式系统中,我们一般把能够自主发挥其自身行为并且能够和其它的个体进行交互的独立个体叫做 Agent, 它一般有下面的特性。

1. 首先 Agent 有自己明确的目标,能够感知环境,并且是一个具有行为的独立个体。Agent 对自己的行为有规划能力,能够自身控制,并且不受外界影响,行为是自发和主动的,并且 Agent 会对它做出计划。

2. Agent 具有交互能力,任何一个 Agent 都是处于一个大的社会环境中,这个社会环境由多个 agent 组成,agent 之间通过相互通信和交互来完成目标,在交互的时候遵循

一些约定的准则和规范。

3. agent 能够对环境的感知做出反应，并且激发自身的某个行为，在行为的执行和调整中不断适应环境。

4. agent 可以在环境变化的时候做出反应，当我们需要的时候也可以对 agent 进行一些特定的行为的定义来使它可以主动地改变环境。

一般来说，我们可以这样定义 Agent：Agent 是一种现实系统的抽象，他为了在其环境中满足自身设计的某些目标，会采取一定的自治行为^[46]；Agent 能够感知所在环境的情况(软件 Agent 由软件传感器感知)，并且可以对环境进行反应和适应环境，并且有能够影响环境的行为。在 Agent 系统中，各个 Agent 之间可以通过一系列规则进行通信，协商和合作。

3.2.2 Agent 体系结构

Agent 的体系结构是描述 Agent 的组成和每个部分的联系关系的模型，它对 agent 的逻辑，行为，目标和社会的角度进行了详细得描述。目前,比较普遍的 Agent 的体系结构大致有下面三类。

1. 审慎式体系结构(Deliberative Architecture)

Agent 可以通过模板的消息和和符号对外界意图进行推理^[47],它的认知部分由两个部分组成，规划器和世界模型。审慎式体系结构模型如图 3-1 所示。

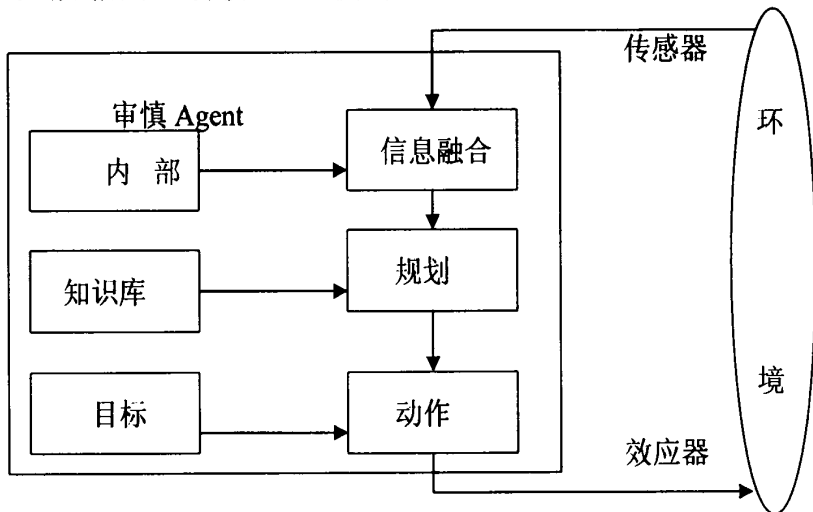


图 3-1 审慎式结构

Fig.3-1 A structure of prudent type

2. 反应式体系结构(Reactive Architecture)

该结构的设计源自如下假设：Agent 行为的复杂性可以是 Agent 运作环境复杂性的

反应，而不是 Agent 复杂内部设计的反应^[48]。Agent 通过规划来建立自己的模型，通过感知外界的环境出发某个行为，对外界进行反应。反应式体系结构模型如图 3-2 所示。

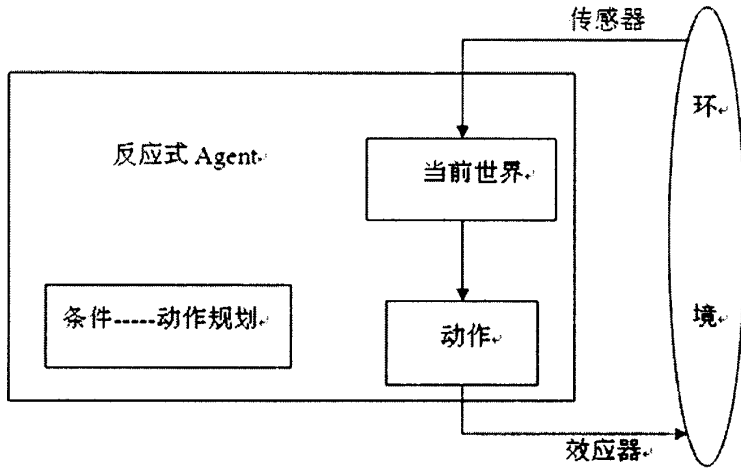


图 3-2 反应式结构

Fig.3-2 A structure of reaction type

3. 混合式体系结构(Hybrid Architecture)

这种体系结构包含了审慎式和反应式体系结构的特点，是这两个系统的结合。一般是审慎式体系结构建立在反应式体系结构的基础上。这种结构的研究现在在人工智能的领域比较活跃,它的相关性的成果会对分布式系统中 Agent 的应用功能增强产生很大的影响^[47]。混合式体系结构不但能够对长期的目标进行规划，还能够对实时的变换进行反应。混合式体系结构模型如图 3-3 所示。

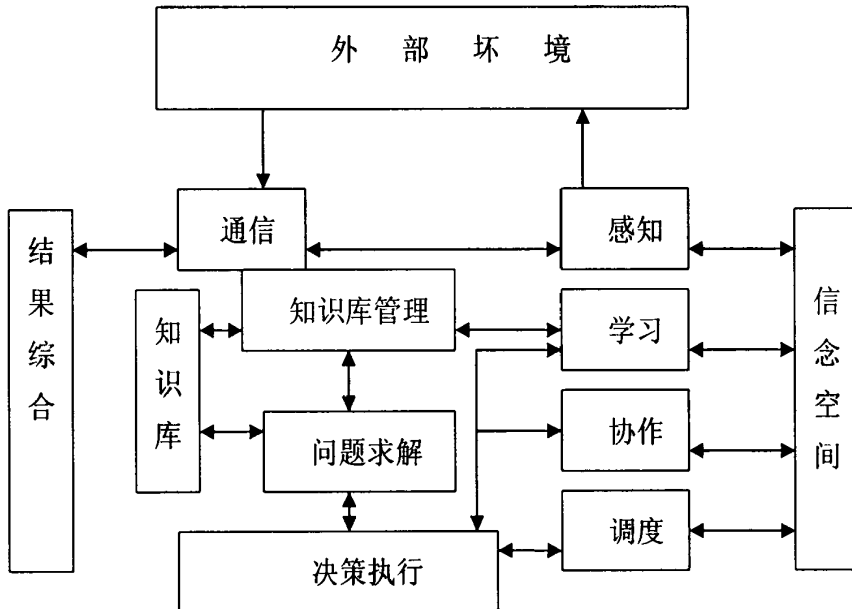


图 3-3 混合式结构

Fig.3-3 A structure of hybric type

3.3 多 Agent 系统

3.3.1 概述

随着计算机技术的发展,我们开发的系统也正在变得复杂,系统的子系统在空间上也朝着分散,可变性、和智能性的方向发展。对现在比较复杂的系统,我们使用单个的 Agent 去描述已经明显达不到系统的要求,因此必须使用多 Agent 来描述、抽象这样的系统。我们通常将多个 Agent 形成的系统叫做多 Agent 系统(Multi Agent System,简称 MAS)。MAS 吸取了分布式人工智能理论的特性和优点,主要采用分而治之的理论,提供了一种解决复杂问题的方法,解决了单个 Agent 解决的规模系统庞大,结构复杂的弊端^[49]。MAS 方法的最鲜明的特点是其系统内 Agent 之间的动态交互。

3.3.2 MAS 体系结构

MAS 体系结构一般指的是各个 Agent 的交互关系,以及 Agent 之间通过怎样的组织和交互方式完成系统特定的任务。这种交互方式定义 Agent 之间的关系来实现。完美的系统结构能使得系统的实时性和稳定性得到很好的提高。因此我们设计 MAS 体系结构的原则就是使得 Agent 之间的交互能够简单,快速来保证系统的优越性能。图 3-4 是标准多 Agent 体系结构的框图。

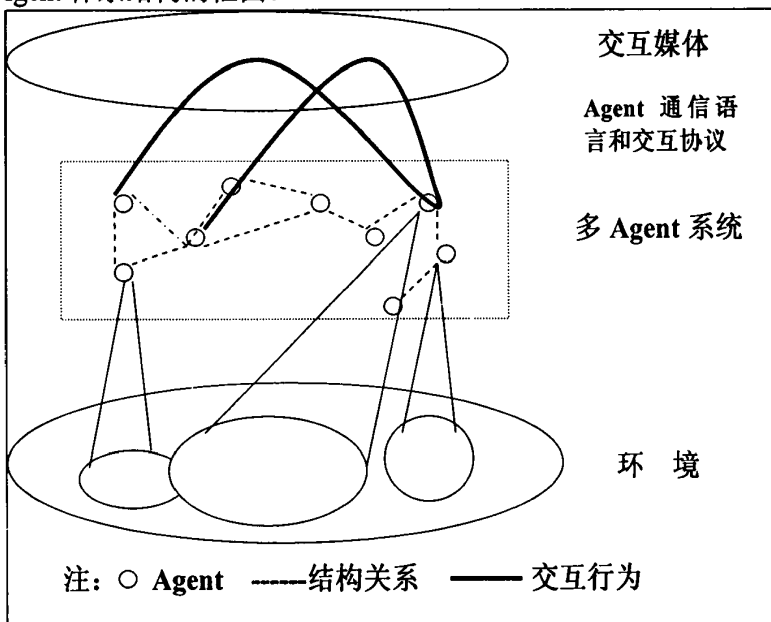


图 3-4 多 Agent 体系标准结构框图。

Fig.3-4 The framework of multi Agent system

一般来说，多 Agent 系统的体系结构可以分为以下几种：

(1) Agent 网络体系结构

Agent 相互之间是可以进行通信的。在这种体系结构中，agent 的相互通信和各自的状态都是先定义好的，每个 Agent 都应该知道自己的消息被发送到的目的地和发送的时间，在 agent 系统中可以和哪些 Agent 进行交互，自身具备的能力等等，整个体系结构看起来就像一个网络一样。

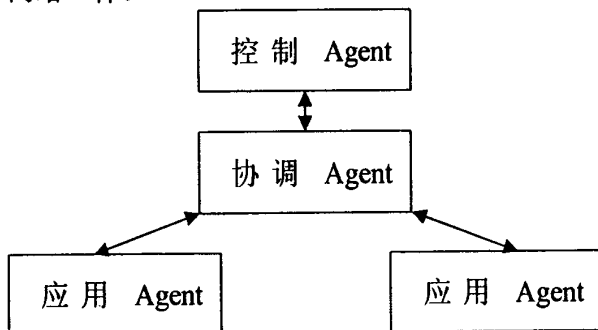


图 3-5 网络体系结构

Fig.3-5 The network structure

(2) Agent 联盟体系结构

联盟结构和网络结构不一样，它的工作是通过一个叫做协调者的 agent，联系着多个距离计较近的的 Agent，由它作为中间人进行交互，远程的 Agent 之间也是通过每个 Agent 群体的协助者进行信息交换和交互的。

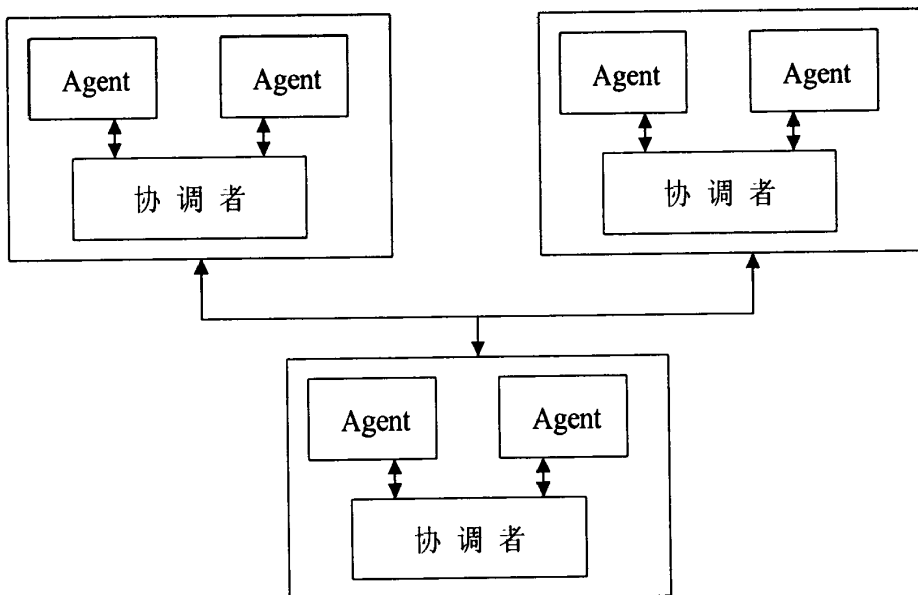


图 3-6 联盟体系结构

Fig.3-6 The union structure

(3) 分布自治体系结构

在 MAS 中的所有 Agent 都是自治的,彼此之间独立平等,如图 3-7 所示,所有 Agent 可以直接通信和交互,形成一种分布自治结构,这是现在最流行的一种 MAS 体系结构。

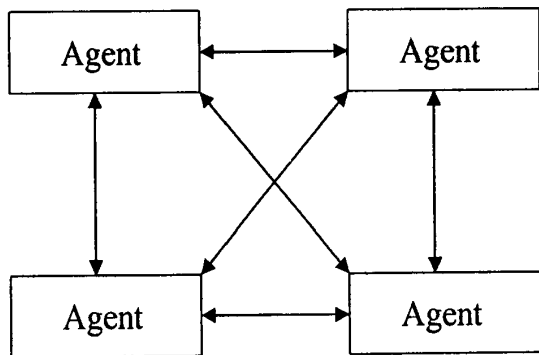


图 3-7 分布自治体系结构

Fig.3-7 The distribution structure

3.3.3 多 Agent 通信

在 MAS 系统中,每个 Agent 都是一个具有独立信念、目标和意图的个体,并且他们之间的作用是相互的。我们用通信的方式来解决它们之间的交互和相互理解的问题来达到各自知识的交互。在每个 MAS 应用系统中有一个共同的特点:我们必须使用有效的通信机制来解决 MAS 中 Agent 的信息交换和交互问题^[50]。目前对于 Agent 通信的研究主要集中在如下两个方面。

1) Agent 通信语言

由于 Agent 之间的通信是依靠消息的传递完成的,这种消息必须有充分的定义和直观的语义,这就需要我们定义一套标准化的 Agent 通信语言。KQML 和 FIPA ACL 是典型的 Agent 通信语言,这种语言的功能是描述 Agent 之间的通信内容本身。使用这种语言的 Agent 可能由不同的设计者设计,并且本身具有很强的行为能力,使用与应用无关的通信设施进行沟通,以完成一些复杂的任务。

在另外一些场景下,比如模拟自然社会,这个时候可能不需要如此“强”的 Agent,而更关心它们交互行为本身。于是便有了描述 Agent 交互场景的语言,这种语言描述交互场景本身而不单只是描述交互的内容,区别于上面的通信语言。其中一个比较成功的例子是日本京都大学的 Toru Ishida 的 Q 语言。Q 语言主要的功能是描述 Agent 之间的交

互场景，可被用以模拟真实社会中的一些群体行为。Q 语言以及其相关环境设计中就力求使得语言易于理解可编写，使得具有领域知识而缺乏计算机专业知识的领域专家可以容易通过使用 Q 语言进行系统建模，而模型的实现则需要计算机专家。类似于软件工程中 UML 的作用，架构师设计软件的架构，而编码专家实现具体系统。

2) Agent 消息机制

MAS 系统中通信的基本单元是消息，各 Agent 之间通过发送和接受彼此的消息来进行交互，为了保证交互的正常和高效进行，我们需要为 MAS 提供一套消息传输的机制。目前对于通信机制的研究主要可以分为两大类：一类是基于消息传递的实现机制，如 PVM（并行虚拟机）；另一类是基于远程过程调用（RPC）的实现机制，如 CORBA。在 MAS 中，MTS 提供了 Agent 之间进行传输消息的机制(FIPA Agent Message Transport Service Specification)。图 3-8 是 FIPA MAS 消息传输参考模型。

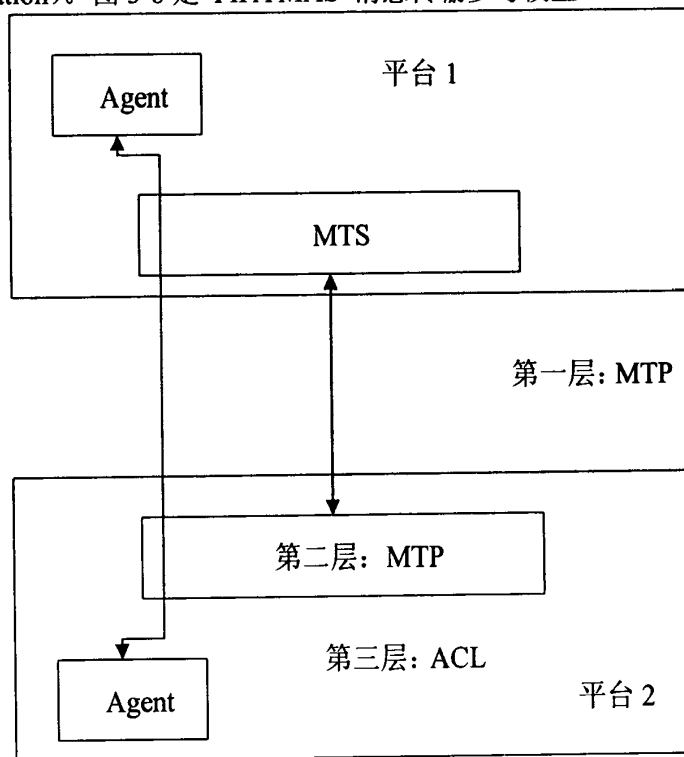


图 3-8 MAS 消息传输模型

Fig.3-8 The model of message transport in MAS

FIPA 消息传输模型由三个层次组成：首先是 MTP（Message Transport Protocol：消息传输协议），它的作用是对 ACC（Agent Communication Channel：消息传输通道）进行物理传输的控制；然后是 MTS（Message Transport Service：消息传输服务），它是由平台自身提供的，支持相同或者不同平台之间 Agent 的消息进行传输；最后一层是 ACL，表示 FIPA ACL，它代表了消息所包含的内容。

即使平台不同，MTS 的提供者都是一样的，它就是通信通道（ACC）。在消息中，

主要由两个方面组成，一个是消息的信封，它提供了消息传输的目的地的地址和自身的地址，还有就是消息内容，它包含的消息的主体信息。在传输的过程中，ACC 会从本地平台上的 AMS(Agent Management System)和 DF(Directory Facilitator)获得消息。获得消息后，对消息进行解析，读出消息信封，即消息的地址，以此来确定消息的接受者，或者当解析不了信封时，将消息传递给接受者所在的平台，消息的传送由其他平台的 ACC 完成。

3.3.4 多 Agent 协调合作机制

在 MAS 环境中，我们需要对多个 Agent 的资源整合和目标调整进行协调合作来达到最终的目标任务。由于每个 Agent 都是一个具有知识，意愿和目标行为的独立个体，协调他们将会使得系统工作的效率得到很大的提高，因为单个的 Agent 执行任务的能力是有限的，多个 Agent 协调合作可以增强系统解决问题的能力，使系统的灵活性得到改善^[51]。

多 Agent 协调合作 (Multi-Agent Coordination and Cooperation) 是指对具有不同目标的多个具有各自独立行为能力的 Agent 进行目标和资源的整合和协调自身的行为，并且相互合作来达到目标的工作方式，它最大化地实现了每个 Agent 的目标。一般而言，多 Agent 的协调合作有如下三种实现方法：

(1) 基于对策论的协调合作

这种方法和传统的对策论相似，包含无通信协商和有通信协商两种情况。无通信协商是指不需要 Agent 之间通信交互，根据自身情况按照对策论选择适当的行为。有通信的协商可以得到协商解。

(2) 基于控制中心的协调合作

这种方法与传统的操作系统方法类似，在 Agent 系统中，为每个 Agent 群体设置一个“代理者”，这个群体中的所有 Agent 都可以与这个“代理者”通过申明一组描述的信息进行通讯和交互，而“代理者”还可以与其他 Agent 群体中的“代理者”进行交互，以此来实现基于控制中心的 Agent 协调合作方式。

(3) 基于社会规划的协调合作

这种方法中每个 Agent 都遵循一定的协商策略，社会规则。以此作为协商的基础的方法。Agent 的行为由于受到这些规则的限制保证了其余的 Agent 的必须具有的行为模式，过滤了一些有冲突的行为，以实现整个 Agent 系统的社会行为的协商。

协调合作一般可以分为显式和隐式两种，多 Agent 系统以前的研究都是基于显式的协调方法，并且产生了许多有意义的结果，形成了多 Agent 协调研究的一个方向。显式协调是指 Agent 的设计要满足对可能的发生的交互进行推理的条件，能够和其他

Agent 进行协商。这种方法目标明确、协调的过程直截了当，我们用它来协调 Agent 交互过程出现的各个 Agent 直接动作行为不一致的情况。这种协调方法的缺点是在协调的过程中需要花费足够的时间，因此在一些讲究实时性的环境下会有弊端。隐式协调是指 Agent 的设计遵循某一局部的行为准则，并且以明显的协调方式完成系协调。它是近期才提出的一个新的方法，为协调的研究开辟了一个新领域。隐式协调的缺点主要存在于社会规划能力的自动生成问题，违反社会规则时的自动处理问题等。显式协调与隐式协调有各自的优缺点，在不同系统中我们对系统应用进行权衡并且利用社会规则去实施协调合作，让 Agent 的行为向着有利于整体性能的方向进行，提高 MAS 系统的性能。

3.4 本章小结

本章主要介绍了 Agent 体系的基本理论和多 Agent 体系的结构。对 Agent 的特性和体系结构进行了总结，并分析了不同体系结构的优缺点。在多 Agent 系统中，首先介绍了集中基本 MAS 系统的结构体系，对他们的优缺点进行了分析。然后分析了在多 Agent 系统中的几个核心的问题，如 Agent 之间的通信语言的制定，Agent 之间传递消息的机制和多个 Agent 之间如何协调合作来达到共同的系统目标。通过多以上理论的认识，为建立 Agent 系统的模型打下了理论基础。

4 基于 Agent 多目视觉人脸特征提取系统

4.1 系统设计概述

在 MAS 系统中，由于各个 Agent 的自治性和智能性以及通过他们之间的合作和调度提高了系统整体的性能，同时也改善了多 Agent 系统解决问题的能力，系统的灵活性也得到了很大提高。在 MAS 中主要的研究问题是 Agent 的创建和多 Agent 之间的交互，Agent 的建立规定了 Agent 的意念，目标和行为属性，它是一个独立的自治个体，而协作则联合了多数 Agent 的力量，在解决实际问题上使得 MAS 具有更大的潜力。

本文方法的基本思想是利用多目视觉从感知阶段解决因为角度和光线等原因带来的人脸误识别问题。我们用 Agent 系统对多目视觉系统进行协调，利用 Agent 的智能性和自治性使多目视觉系统的每个摄像头相互协调，相互合作去完成目标。Agent 系统的设计要符合简单，高效的原则，最终智能系统的目的要体现在多目视觉的灵活性和对误识别问题的解决上面。下面的章节，我们将逐步介绍多目视觉 Agent 系统的各个关键的技术。

首先，我们要确定 Agent 系统的开发工具，这里我们选用 JADE 平台，4.2 节将给出 JADE 平台详细的介绍和使用方法。整个 Agent 系统的流程框图将在 4.3.1 中给出。在 4.3.2 中我们将详细描述各个 Agent 的创建和其属性，4.3.3 节将会描述多 Agent 之间如何交互以及通信方式和消息的传递标准。在人脸特征提取上，由于现有的 ASM 算法很多，并且有很多开源的库，在这里我们选用 C 平台结合 OpenCV 库来实现。在 Agent 进行交互的时候，我们需要得到交互的参数来实现 Agent 的各个行为，参数的属性和要求将在 4.3.3 中介绍，这里我们需要的是人脸的姿态参数作为交互的信息。在 4.3.4 节，我们将详细介绍如何进行人脸的特征提取和姿态估计。由于 JADE 是基于 java 平台的应用程序，我们如何将 C 平台上的代码成功应用到 java 平台上成了一个问题，JNI 标准为这个问题提供了一个可行的方案，在 4.3.5 节，我们将详细介绍 JNI 标准的原理以及实现的流程图，以及如何将这个标准应用到我们的 Agent 系统中来实现人脸特征提取和姿态估计的代码的共享和复用。

4.2 Agent 平台 JADE

在本系统中，我们采用 JADE 平台来开发 Agent 系统^[52]。JADE (Java Agent Development Framework) 是一套软件开发框架，它由两个主要部分组成：一个是遵循

FIPA 规范的 Agent 平台和一个开发 Agent 的软件包。JADE 的目的是使得多 Agent 系统的应用程序的开发变得简单，它的框架全是由 Java 编写，并且由许多 Java 包组成，这些 java 包为我们提供了现成的函数实现和抽象接口。在这里我们选择 Java 作为编程语言是由于它的很多有吸引力的特点所决定的，尤其是它在分布式异类环境下进行的面向对象编程的优势；对象序列化是这中间最明显的特点，还有比较方便的就是它的 API 映射和远程方法调用(RMI)。JADE 的目标是通过系统服务和主体集的规范去简化 MAS 的开发过程。

4.2.1 JADE Agent 体系结构

在 JADE 中，有 FIPA 定义的标准 Agent 平台模型，如图 4-1 所示：

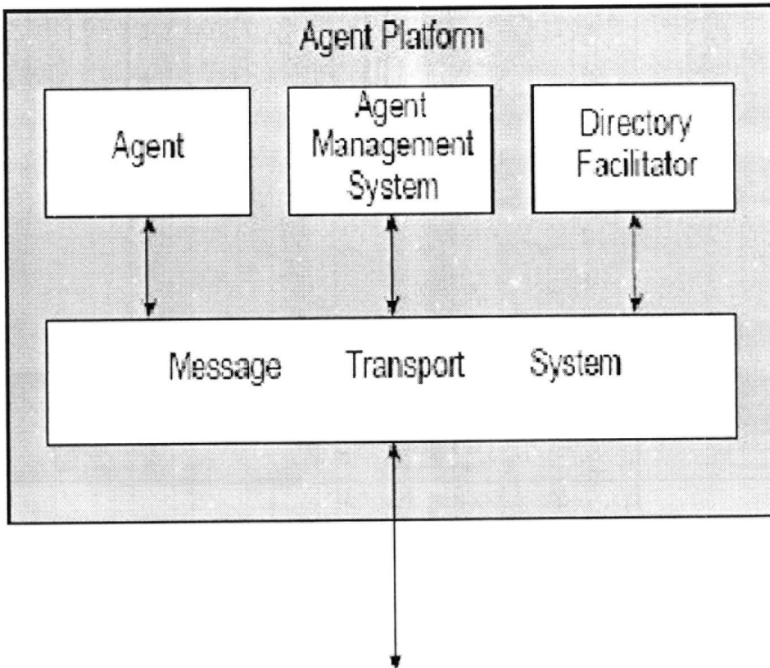


图 4-1 FIPA 标准 Agent 平台的参考结构

Fig.4-1 The structure of Agent based on FIPA standard

Agent 管理系统 (Agent Management System) 是用来对整个 Agent 平台的访问和控制进行监督的 Agent。每一个平台只有一个 Agent 管理系统。它维护着每一个 Agent 的 ID 的目录 (AID, directory of Agent identifiers) 和状态，并且提供生命周期和白业服务。目录服务器 (Directory Facilitator), 这个 Agent 在平台中提供黄页服务。消息传输系统 (Message Transport System), 也就是 Agent 通信信道 (ACC, Agent Communication Channel), 它是一个软件组件，控制了在整个平台中的消息交换，不管是发送到远程还是远程平台发送过来的。

一个任务的模型，并且所有的任务都是同时进行的。在 Agent 的属性中，它所拥有的能力都是依靠一个或者多个行为来实现的，并且由控制中心自动管理和调度。每个 Agent 在它运行的过程中都会有一个生命周期，如图 4-3 所示。

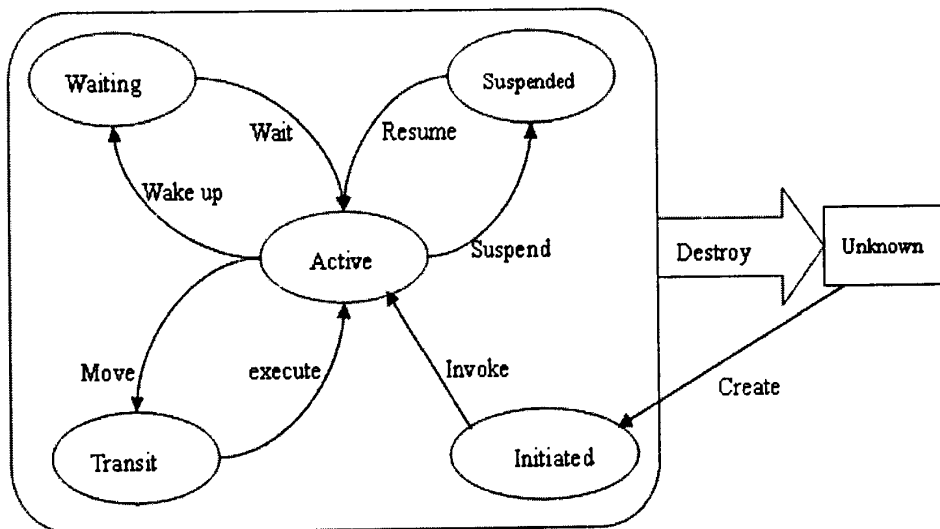


图 4-3 FIPA 定义的 Agent 生命周期

Fig.4-3 The life cycle of Agent defined by FIPA

根据 FIPA 规范，我们所说的 Agent 平台生命周期的意思就是，一个 JADE Agent 可以处于上图状态中的任何一个，图 4-3 描述的细节如下：

初始状态 (Initiated)：开始生成一个 Agent 的对象，这个对象还没有注册到 Agent 管理系统上，它没有名字和地址，并且不可以和其他 Agent 进行通信。

活动状态 (Active)：此时 Agent 对象已经注册到 Agent 管理系统上，它有了有名字和具体的地址，一切 JADE Agent 的特点它都具有。

挂起状态 (Suspended)：这时 Agent 对象运行暂停。Agent 线程此时挂起，它不再有执行 Agent 行为的能力。

等待状态 (Waiting)：Agent 对象此时处于等待状态，它在等待一些触发唤醒的条件。Agent 线程在监视器上处于休眠状态，当满足一些条件的时候，比如接受到某条唤醒消息，它就会停止休眠。

删除状态 (Deleted)：当我们已经确定 Agent 是废弃的时候。Agent 线程将会停止了 Agent 行为的继续执行，此时 Agent 已经从 Agent 管理系统上移除了。

转换状态 (Transit)：一个 Agent 变化到另一个新的位置时，就会进入转换状态。那些要被发送到新位置的消息将会被系统缓存起来。

Agent 类中有一些公共的方法是专门用来实现各种状态之间转换，这些方法是从 FIPA 规范的状态自动机中获得了一个相应的名字。比如，doSuspend()方法，它是把活动状态的 Agent 转换到悬挂状态，而 doWait()方法是把活动状态的 Agent 转换到等待状

态等等。

我们必须了解,只有当 Agent 处于活动状态的时候,它才被允许执行它自身的行为。当 Agent 的某个行为调用了 doWait()方法,这时不光是这个行为调用了 doWait()方法,并且整个 Agent 的行为都处于阻塞的状态。

当 Agent 有不同的外部时间需要实现的时候,它就应该具有同时执行多个任务的能力。在 JADE 中,Agent 都是被设计成一个单独的可执行线程,它的任务已经模块化,并且作为 Behavior 对象去实现的。JADE 没有提供一个多线程 Agent 的框架,因此,当我们需要 Agent 能执行多个行为的时候,我们就需要定义多个实例化的 Behavior 的子类,然后将它们都添加到 Agent 的行为执行列表中去。在进行 Agent 编程的时候我们必须继承 Behavior 这个类,我们有两个方法来管理一个或者多个 Agent 行为的就绪任务,addBehavior(Behavior)用来在任务列表中添加一个行为,removeBehavior(Behavior)则是从任务表中删除一个行为。当 Agent 添加一个行为时,其实就是在 Agent 中产生了一个新的可执行线程的方法。一个 Agent 行为的执行流程图如图 4-4 所示。

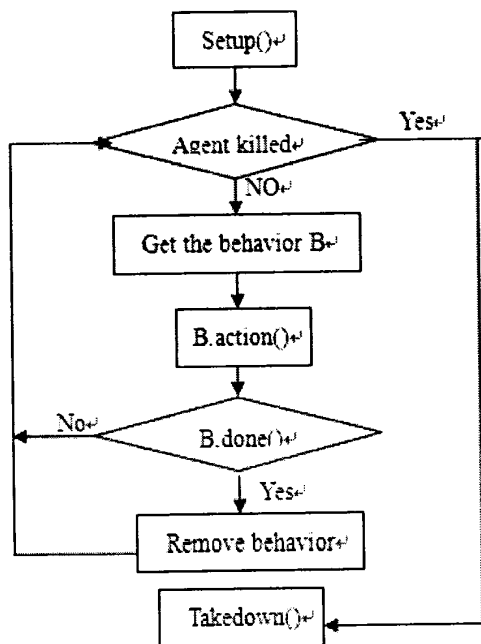


图 4-4 一个 Agent 行为的执行流程图

Fig.4-4 The flow of how an Agent execute its behavior

调度程序是由 Agent 的基类实现的,并且它对程序员是不可见的,在就绪的队列中所有将要执行的行为遵循着循环非抢占的调度策略。当执行一个 Behavior 类,除非它释放了控制权,一般发生在 action()方法结束,不然它是不会终止的。如果还没有放弃控制权,那么调度程序将会安排下一个循环的任务执行。那个行为将会被阻塞,知道等到唤醒的消息。Agent 调度程序会去执行所有的任务队列中的 Agent 行为的 action()方法,

当一个 `action()` 结束, `done()` 方法会用来检查这个行为是否已经完成任务。完成则从队列中删除这个行为对象, 没有完成则继续行为的执行。行为的运行就好像是线程之间的相互协作, 唯一不同的是它们没有堆栈可以去存储。

4.2.3 Agent 通信语言

在 Agent 通信中, 会使用 `ACLMessage` 类。它详细描述了 Agent 进行的 ACL 消息之间的相互交换。包含了 FIPA 制定的一些属性。所有的 Agent 当准备发射消息的时候都会首先创建一个新的 `ACLMessage` 对象, 并对其相关属性赋予相应的值, 然后我们可以调用 `send()` 方法将这个消息发送出去。相同的道理, 所有准备接收消息的 Agent 都会去调用 `blockigReceive()` 方法或 `receive()` 方法它们由 Agent 类实现。我们使用 `SenderBehaviour` 将发送属性行为添加到 Agent 的行为任务表, 并且用 `ReceiverBehaviour` 将接收属性行为添加到 Agent 的行为任务表中。只要 Agent 处于活动装, 它可以随时发送或者接收消息。FIPA 定义的 ACL 消息由通讯的参与者、述行语、消息内容等几个要素组成。其中述行语是消息必须包含的要素, FIPA ACL 消息的结构见图 4-5。

元素	类型
述行语(Performative)	通信行为类型
发送者(Sender)	通信参与者
接受者(Receiver)	通信参与者
回复给(Reply-to)	通信参与者
内容(Content)	消息内容
语言(Language)	内容描述
编码方式(Encoding)	内容描述
本体(Ontology)	内容描述
协议(Protocol)	会话控制
会话 id(Conversation-id)	会话控制
以...回答(Reply-with)	会话控制
对...回答(in-reply-to)	会话控制
回答不迟于(Pepley-by)	会话控制

图 4-5 FIPA ACL 消息的结构

Fig.4-5 The structure of FIPA ACL message

`ACLMessage` 的对象属性可以通过 `set<Attribute>()` 或者 `get<Attribute>()` 方法被访问, 属性都命名都是在参数名的后面。参数的类型是某些值的集合, 通过 `add<Attribute>()` 方法或者 `getAll<Attribute>()` 方法可以去访问这些参数, 其中 `add<Attribute>()` 方法向集合

中增加了一个值, 而 `getAll<Attribute>()` 方法首先遍历集合内的所有值, 然后返回一个迭代对象。当属性的值没有设置的时候, `get()` 方法的返回值返回为空。 `ACLMessage` 类还定义了用来描述 FIPA 行为的常量, 比如 `REPLY`, `INFORM`, `REQUEST` 等等。一个新的 `ACLMessage` 对象创建的时候, 它自己的一个常量值必须被初始化给 `ACLMessage` 类的构造器, 这样才可以确定发送或者接收的消息所表述的行为。

在某些应用程序中, 可能会需要通过 `ACLMessage` 去传送字节序列。最典型例子就是利用 Java 的序列化在两个 Agent 之间传送 Java 对象。 `ACLMessage` 类允许我们使用 `setContentObject()` 方法和 `getContentObject()` 方法去得到发送或者得到的 java 序列化对象值, 这两个方法会自动激活 Base64 编码。

4.3 Agent 系统设计

4.3.1 系统流程框图

对于一般的 Agent 系统, 它需要感知外界的环境, 并且对外部环境做出反应, Agent 自身的自治性和智能性使得对于不同的 Agent 能解决不同的任务^[53], 对于比较复杂和庞大的系统, 我们通过各个 Agent 之间的交互和合作建立的多 Agent 系统来解决, 通过一系列的交互规则, 各个 Agent 之间能最大程度地发挥自身的属性, 并且相互合作去达到共同的目标, 普遍的 Agent 系统的框架如图 4-6 所示。

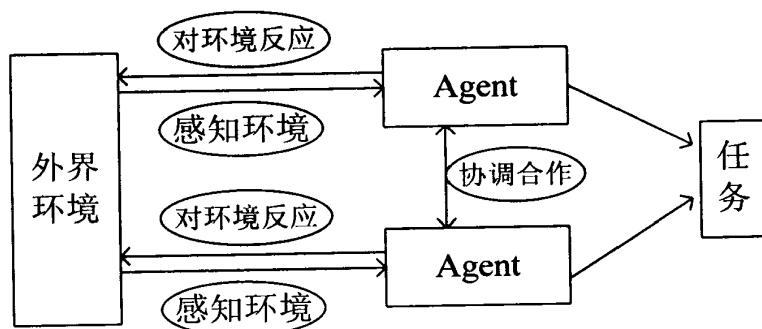


图 4-6 Agent 系统的一般框图

Fig.4-6 A normal Agent system

在本文的 Agent 系统中, 我们需要感知各个摄像头的环境属性, 以此得到关于各个角度的人脸信息。Agent 同时要对环境做出反应, 我们将计算感知到的人脸信息并给定规则计算出需要的人脸参数作为下一步交互的基础。本文中, 我们将使用中心控制结构的 Agent 系统, 首先我们会创建一个对所有平台内 Agent 进行控制, 调度和管理的中心

Agent, 它将会引导其他的 Agent 完成系统的最终目标。Agent 之间需要通过一定机制来进行交互, 对于前面计算得到的人脸参数, 我们需要一定的交互规则对这些参数进行一定的定义和整合。每一个感知的 Agent 都会得到一个标准定义的参数, 此时中心控制的 Agent 会通过消息机制得到这些参数, 并根据一个判断条件最终会选出相比较而言对于人脸特征比较好的人脸参数, 此时发送消息给中心控制 Agent 的那个 Agent 感知到的图片信息就是我们需要的人脸特征信息, 我们将这些信心保存起来, 作为下一步人脸识别的基础。整个系统的设计的流程如下面的图 4-7 所示。

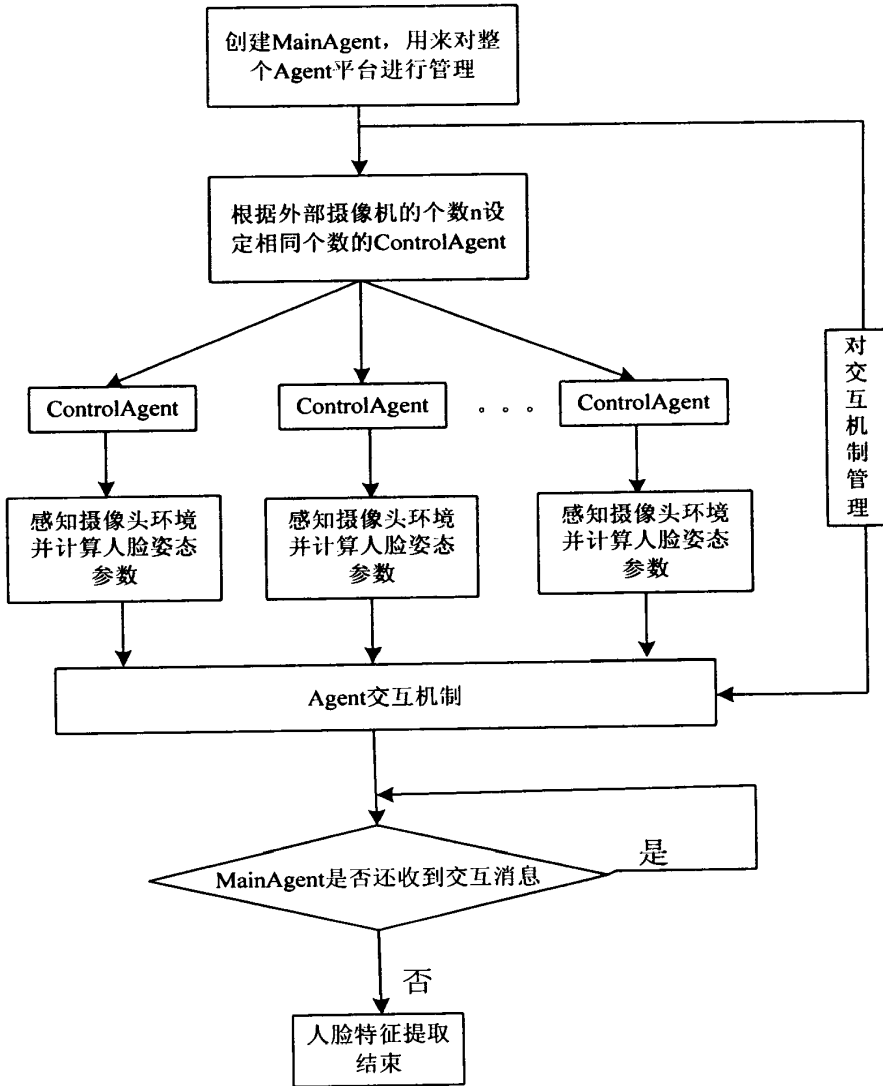


图 4-7 系统流程图

Fig.4-7 The flow of our system

4.3.2 构造各类 Agent

首先,我们要对需要使用的各类 Agent 进行构造。在本文提出的 Agent 框架下,需要构造一个对整个 Agent 平台管理,调度的中心控制 Agent,我们称之为 MainAgent,它具有一系列的属性和行为,MainAgent 的行为都是在它的 action()方法中实现的。

MainAgent 的 action()方法可以表示为一个三元组,即

$$\text{MainAgent.action} ::= \langle \text{Create}, \text{Control}, \text{Receive} \rangle$$

(1)Create(创建属性):根据环境中摄像头的个数创建相应个数的 ControlAgent,并告诉每个 ControlAgent 所要感知的目标,即图片路径来感知人脸的环境。作为整个系统的控制者,主 Agent 根据系统的需求创建一定数目的多功能的 Agent。

(2)Control(控制属性):多 Agent 系统的协作能力是共同解决复杂问题的关键,它能最大程度地发挥每个 Agent 的能力。MainAgent 的主要任务就对多 Agent 的交互行为进行管理,引导他们最终能够一起完成目标。在本方法中,主要对多 Agent 之间的一个竞争机制进行控制。

(3)Receive(获取属性):系统最终的目的是要得到最佳的人脸特征,因此最后我们需要将得到的人脸特征获取并储存起来。在各个控制 Agent 进行交互的时候,会激发竞争机制,竞争的目标就是得到最佳特征。当各个 ControlAgent 竞争结束,人脸姿态最优的图片已经选出,此时将最优人脸的特征传递给 MainAgent,完成特征提取工作。

我们使用一个具有感知和计算属性的 Agent 来对环境进行感知并进行人脸特征参数的计算,这个 Agent 我们称之为 ControlAgent,通用 ControlAgent 的行为属性是在它的 action()方法中实现的。ControlAgent 的 action()方法同样可以用一个三元组来表示,即:

$$\text{ControlAgent.action} ::= \langle \text{Perception}, \text{Competition}, \text{Send} \rangle$$

(1)Perception(感知属性):Agent 最基本的属性是感知,首先我们需要 ControlAgent 感知目标环境区域,得到人脸信息并进行人脸特征的提取,然后需要对环境做出反应,这里我们设定一定的规则,根据这个规则计算出人脸姿态的参数作为竞争机制的基础。

(2)Competition(竞争属性):多 Agent 之间进行交互时需要遵循一定的交互机制来完成信息交换。在本文的系统中,根据需要我们给定了各个 ControlAgent 之间的交互规则为竞争机制,即各个 ControlAgent 将得到的人脸姿态参数进行交互,由主 Agent 给出竞争的最终目标值,然后根据竞争判决的条件决定哪一个参数才是最终的目标值,即最佳的人脸特征。

(3)Send(发送属性):最终最佳的人脸特征是要由 ControlAgent 发送 MainAgent 的,因此每一个 ControlAgent 都要有一个能发送人脸特征的属性,当某 ControlAgent 竞争胜

利时，会触发发送属性，向 MainAgent 中传递自身的人脸特征参数。

4.3.3 Agent 交互规则

当每个 Agent 得到自己环境的人脸姿态的参数时，它会和其余的每个 Agent 进行交互，在交互的过程中，会触发一系列的 Agent 行为，各个 Agent 会去竞争成为最优的人脸姿态目标。一般而言，正面人脸包含人脸大部分的特征信息，而有角度偏转的人脸会失去一部分的人脸特征信息，因此在我们的竞争机制中，我们规定人脸姿态最优的标准为相对正面在按个坐标下角度偏转最小的人脸。对于每个 Agent 而言，它们会感知环境得到不同角度的人脸特征参数，因此他们估计得到的人脸姿态也是不一样的。在 Agent 竞争机制中，我们最终将得到各个 Agent 中相比较而言人脸姿态最优的目标。因此我们定义三条人脸姿态的参数作为竞争机制所需要的因素，将人脸划分为三维的 x, y, z 坐标系，三个参数分别为在三个坐标下偏转角度，称之为 Pitch, Yaw 和 Roll，如图 4-8 所示，三个参数的详细计算将在 4.3.4 节给出。

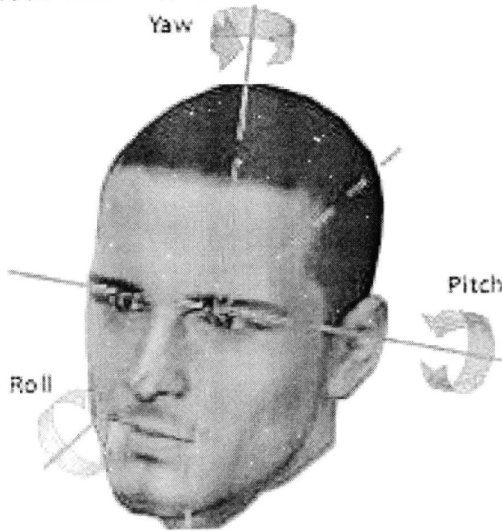


图 4-8 三维人脸坐标图

Fig.4-8 A face in a 3D coordinate system

在本文的 Agent 系统中，主 Agent 会向其余各个 Agent 提供一个人脸姿态的目标值来供大家竞争，只有 Agent 中最优的人脸姿态会成为最终的目标值。根据前面所定义的三个人脸姿态的参数，我们可以得到人脸在三个坐标下的偏转角度。人脸姿态的目标值即人脸在三个坐标中偏转角度的绝对值的和。这里为了计算数值的方便，我们计算每个角度的欧拉角。人脸姿态的目标值 Value 的公式为：

$$Value = abs(\frac{180 \times V1}{\pi}) + abs(\frac{180 \times V2}{\pi}) + abs(\frac{180 \times V3}{\pi}) \quad \dots (4-1)$$

其中，abs()函数表示取绝对值，V1,V2,V3 分别表示三维人脸 x, y, z 三个方向的欧拉角。

当 Agent 间进行交互时，竞争规则的判决函数为：

$$Value = \begin{cases} temp, & \text{if } (temp < Value) \\ Value, & \text{else} \end{cases} \quad \dots (4-2)$$

其中 temp 为交互时各 Agent 间相互传递的自身计算得到的人脸姿态值。

Agent 之间通信的消息结构使用的 ACL 语言可以很方便的实现 Agent 之间的交互和信息的传递。用 sendMessage()和 receiveMessage()来进行消息的发送和接受，在发送人脸特征给 MainAgent 时，我们使用一组 java 序列化对象进行传递。消息接受和发送遵循排队机制，都是顺序等待进行消息的处理。当 ControlAgent 竞争完毕，没有消息再发送给 MainAgent 中时，此时人脸特征提取结束。

4.3.4 人脸姿态参数

这一节我们将详细描述如何得到在 Agent 交互时所需要的人脸姿态参数的计算过程，首先需要人脸在三个坐标下相对于正面人脸的偏转角度，由这三个角度值的绝对值之和可以得到交互的人脸姿态参数。

为了估计人脸姿态旋转角度，需要已知人脸法线方向和人脸凝视方向^[54]。人脸法线方向即为鼻尖到人脸平面的垂直线即 $\overline{P_2P_1}$ ，人脸凝视方向就是人眼平视的直线方向即 $\overline{P_2P_3}$ ，如图 4-9 所示。人脸法线和人脸的凝视方向都在对称的平面内，他们之间的夹角为 β ，统计发现 β 的取值一般为 10 度。因此我们计算得到人脸的法线法向就可以得到人脸的凝视方向。

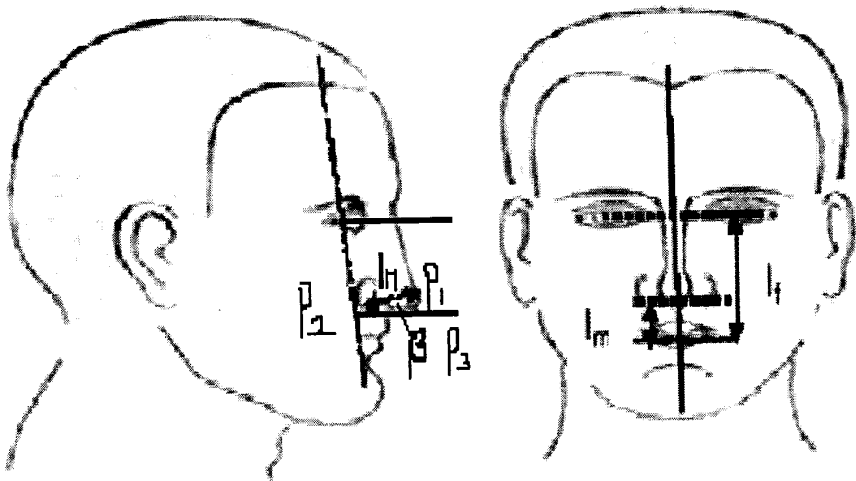


图 4-9 三维人脸模型

Fig 4-9 The model of 3D face

为了得到人脸法线方向，就需要的得到 $P_1(x, y, z)$ 和 $P_2(x, y, z)$ 的坐标，它们的 x, y 轴坐标值可由二维图像中的相应点得到，对于 P_1 和 P_2 ，有

$$\begin{aligned} x_1^2 + y_1^2 + z_1^2 + (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 \\ = x_2^2 + y_2^2 + z_2^2 \end{aligned} \quad \dots (4-3)$$

由正投影的对应关系，可以得到下式

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} * \cos \sigma = z_2 - z_1 \quad \dots (4-4)$$

式中， σ 是三维人脸的法线与 Z 轴的夹角，它可以可由式 (4-5) 计算得到^[55]。

$$\begin{aligned} R_n^2(1 - \cos \theta)(\cos \sigma)^4 + (m_n - R_n^2 + 2\cos \theta R_n^2)(\cos \sigma)^2 \\ - \cos \theta R_n^2 = 0 \end{aligned} \quad \dots (4-5)$$

式中， m_n 是二维图像的鼻子的尖部到根部距离和原点到嘴部中点距离之比，如图 4-9 所示，即 l_m / l_f 。 R_n 则为三维空间中上述两个距离之比即为 l_n / l_f 。经过统计可以发现， R_n 为常量，亚洲人的比例一般为 $R_n = 0.5$ ，而欧洲人则为 $R_n = 0.6$ 。 θ 为二维人脸中法线方向与对称轴之间的夹角。计算出 σ 的值后，代入式 (4-3) 和式 (4-4) 就可估计出 $P_1(x, y, z)$ 和 $P_2(x, y, z)$ ，即可以得到人脸法线的方向，也就得到了人脸凝视的方向。

由于任意姿势人脸相当于正面人脸绕 z 轴旋转 α 角度，再绕 Y 轴旋转 β 角，再绕 x 轴旋转 φ 角而得到。若相应姿势下的人脸凝视方向已知，将其分别绕 x, y, z 轴旋转 φ' ， β' ， α' 后，即可回到原来的正面姿势人脸。因此我们在 4.3.3 中定义的三个标准 Pitch, Yaw 和 Roll 可以表示为：

$$Pitch = -\varphi', Yaw = -\beta', Roll = -\alpha'$$

为了方便，我们一般计算出各个角度的欧拉角，以此作为竞争的因素来选出最优的人脸姿态目标图片。

4.3.5 JNI 标准

JNI(Java Native Interface)即 JAVA 本地调用。它允许 Java 代码和其他语言写的代码进行交互。JNI 设计的初衷是为了本地已编译语言，尤其是为了是 C 和 C++ 语言而设计的，但是它并不局限于这两种语言，对于其他语言，我们只需要调用约定受支持就可以了。最普遍的 JNI 应用就是连接 C 平台和 java 平台实现代码共享如图 4-10 所示。

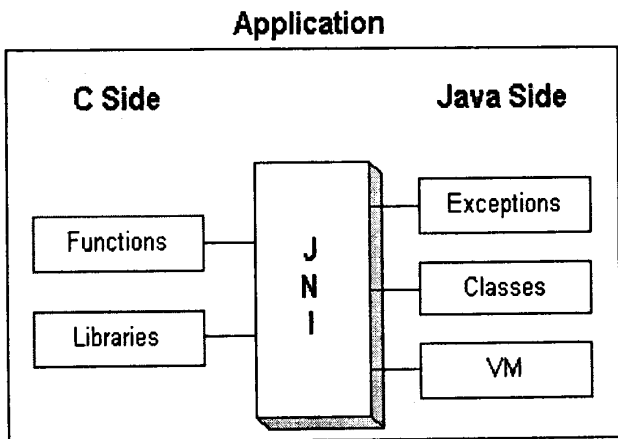


图 4-10JNI 实现 C 和 java 平台资源共享

Fig.4-10 The share of C and java realized by JNI

使用 java 与本地已编译的代码进行交互的时候，通常会使平台的可移植性丧失。然而有些情况下这样做是可以被接受甚至是必须的，比如，使用一些旧的库和操作系统进行交互或者为了提高程序的性能的时候就可以使用。JNI 标准保证了本地的代码可以工作在任何 Java 虚拟机下。

我们在使用 JNI 时最常问到的是 JAVA 和 C/C++之间如何传递数据，以及数据类型之间如何互相映射。这些基本类型的的映射定义使用使得数据的处理变得简单，就是在 C 或者 java 平台下进行函数实现一样。JNI 下的数据类型映射如图 4-11 所示。

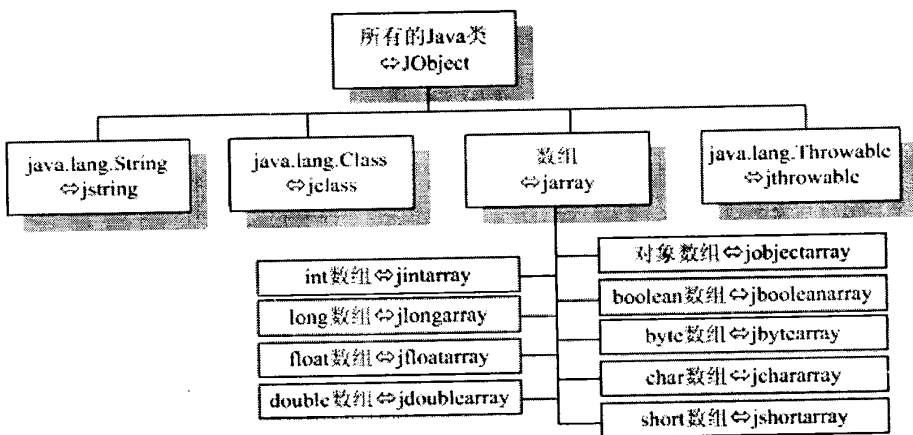


图 4-11 JNI 下的数据类型的映射图

Fig.4-11 The map of data structure in JNI

数据的映射关系明确了之后就可以实现 java 和 C 平台的交互编程，一般的 JNI 实现步骤如下：

1. 创建我们要在 java 中使用的类(*.java)的本地方法声明。

2. 使用 javac 命令编译源文件(*.java), 可以得到 (*.class)文件, 然后再使用 javah 命令来生成 C 头文件(*.h),在这个头文件里面包含了本地方法的函数原型。
3. 在 C 平台下对函数的功能进行 C 语言的实现。
4. 对 C 函数创建工程, 编译成一个动态链接库*.dll 或者*.so。
5. 运行程序。

整个步骤的流程图如图 4-12 所示。

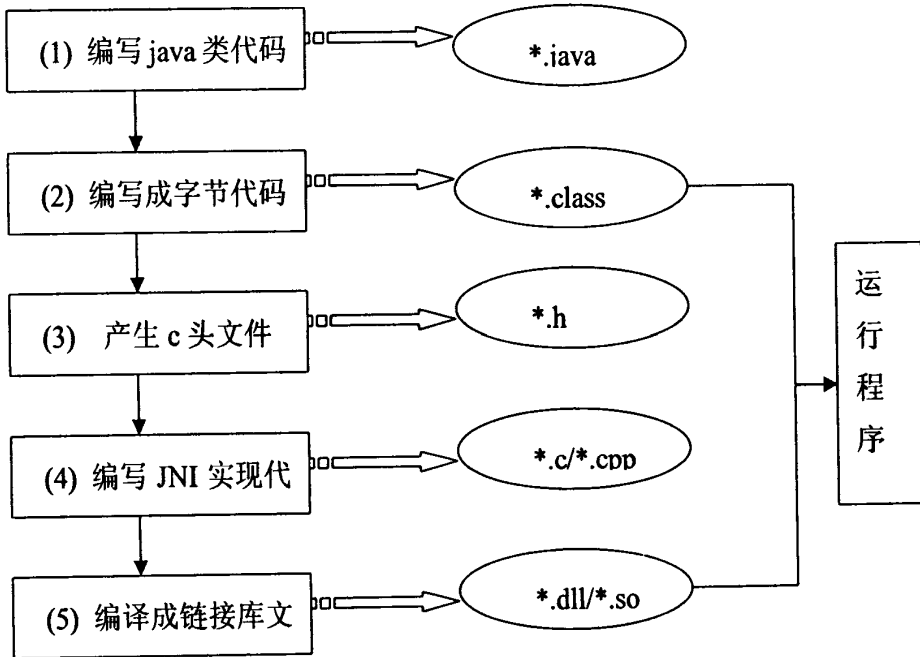


图 4-12 JNI 流程图

Fig.4-12 The flow of JNI

在我们的 MAS 系统中, 所有的 Agent 都创建在 java 环境中, 在 java 中要调用的 C 语言方法我们称为本地方法(native function),因此在 ControlAgent 中, 我们首先声明要得到的人脸姿态参数的方法 HandleFace()为 native, 即 private native double HandleFace (); 然后用 javah 命令编译成 HandleFace.h 文件, 在 VS 中可以用 C 语言结合 openCV 库实现 HandleFace()的功能, 其中各种数据类型的映射在图 4-11 中已经给出。将工程编译成动态链接库文件 HandleFace.dll。在 ControlAgent 中以 static 块进行加载, 即 System.loadLibrary("HandleFace")。这样就可以在 ControlAgent 中调用 HandleFace 函数得到人脸的特征和人脸姿态的参数了。

4.4 系统实现

4.4.1 JADE 平台实现

在 JADE 中,我们要实现各个功能 Agent 的创建,每一个 Agent 都是继承于 Agent 类,即类的声明为 `public class MainAgent extends Agent{}`和 `public class ControlAgent extends Agent`。MainAgent 是对整个平台管理和控制的它是在系统开始时就生成的,因此在运行 JADE 时就生成它,在 Netbeans 中使用命令行 `jade.Boot -gui MainAgent:MainAgent` 可以生成 MainAgent。而 ControlAgent 的创建是在 MainAgent 中进行的,在一个 Agent 中创建其他的 Agent 需要得到容器的一些信息和遵守一定的规则。由于它们都是在主容器中,因此使用图 4-13 中的命令生成 ControlAgent。首先要得到主容器的信息,循环的目的是要生成三个 ControlAgent, 在生成的时候会初始化给每个 Agent 它的名字和参数。

```
AgentContainer c=myAgent.getContainerController();
for(int i = 0; i < 3; i++)
{
    Object[] arguments = new Object[1];
    arguments[0] =title[i];
    try
    {
        AgentController s=c.createNewAgent("camera"+i,"ControlAgent",arguments);
        s.start();
        System.out.println("handle agents camera"+i+" has been created");
    }
    catch (Exception e)
    {
        System.out.println("Create Agent Error!");
    }
}
```

图 4-13 ControlAgent 的生成命令

Fig.4-13 The command to create a ControlAgent

当 Agent 都创建好后,我们需要为他们制定一系列行为属性和交互机制,每个 Agent 的行为都是由 behavior 类实现的,通过 `addBehaviour()`把需要的行为添加到行为人为队列中去。而 Agent 之间的交互是通过消息机制来实现的。Agent 之间通过 ACL 消息时进行通信和信息的交换。当 Agent 得到感知人脸的姿态参数后(姿态参数的计算由 OpenCV 库实现,并通过 JNI 标准实现 java 平台和 C 平台的代码共享),会和其余的 Agent 竞争

成为目标姿态值，这里是和主 Agent 之间交互完成的，ControlAgent 和 MainAgent 之间消息传递的实现见图 4-14，它包含了消息的接受者定义，消息的类别属性和消息的主题 java 序列化内容。

```

Receivedata mg = new Receivedata(x, filename);
AID r = new AID ("MainAgent@" + myAgent.getHap(), AID.ISOUID);
ACLMessage aclMessage = new ACLMessage(ACLMessage.INFOFORM);
aclMessage.addReceiver(r);
try
{
    aclMessage.setContentObject(mg);
}
catch (IOException e)
{
    System.out.println(e.toString());
}
aclMessage.setLanguage("JavaSerialization");
myAgent.send(aclMessage);

```

图 4-14 消息传输定义

Fig.4-14 The definition of the message transport

4.4.2 系统界面

JADE 中为了 Agent 开发的方便，维护了一个带有 UI 界面的 Agent 框架。我们为主 Agent 创建一个系统界面，当系统运行时主 Agent 也会启动，界面也会启动。首先这个 UI 界面是基于 JFrame 类的，即 `public class MainAgentGui extends JFrame{}`，然后在主 Agent 中创建一个实例，即 `myGui = new MainAgentGui(this)`，使用 `myGui.showGui()` 命令就可以启动我们自己定义的界面了，如图 4-15 所示。在界面中，我们定义了几个简单文本框，图形框和按钮，文本框主要是用来输入图片的路径，即 Set file1 name 中的 p1.jpg，Set file2 name 中的 p2.jpg 和 Set file3 name 中的 p3.jpg，还有一个是显示最好姿态人脸的图片名称，即 The best facial picture is 中的 p1.jpg。按钮是开始特征提取，即通过 Agent 框开始对多个 Agent 交互来达到选取较好人脸特征的目标。图形框是用来显示最佳姿态人脸的特征提取结果。JADE 同时还为 Agent 系统提供了一个图形化界面，如图 4-16 所示。在这个界面中，我们可以发现我们所创建的 Agent 都在容器中，我们可以直接对容器中的 Agent 进行操作，如增加和删除行为，和其余 Agent 进行通信交互，还可以直接增加和删除容器，Agent 等等。

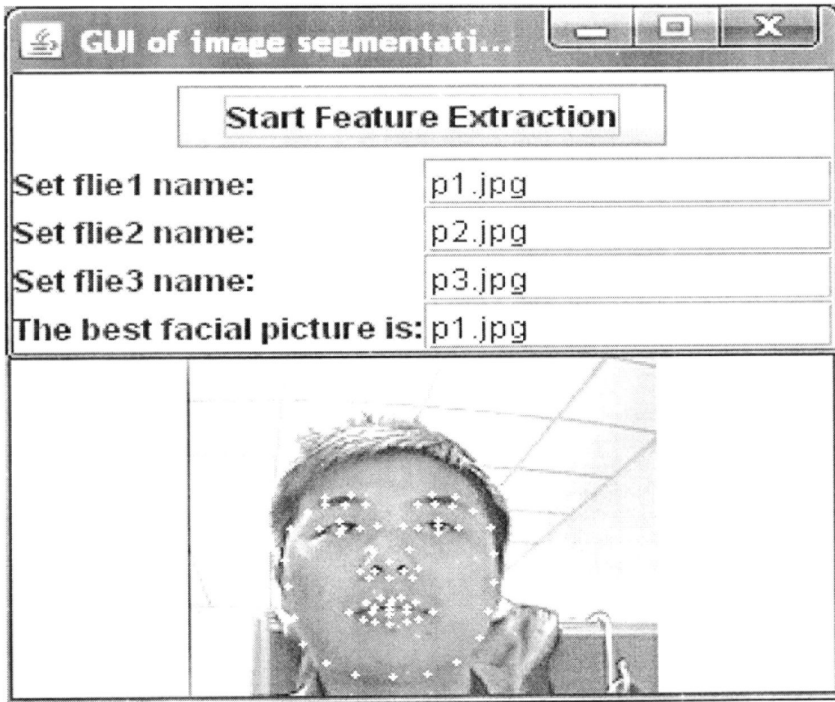


图 4-15Agent 系统界面

Fig.4-15The Agent system interface

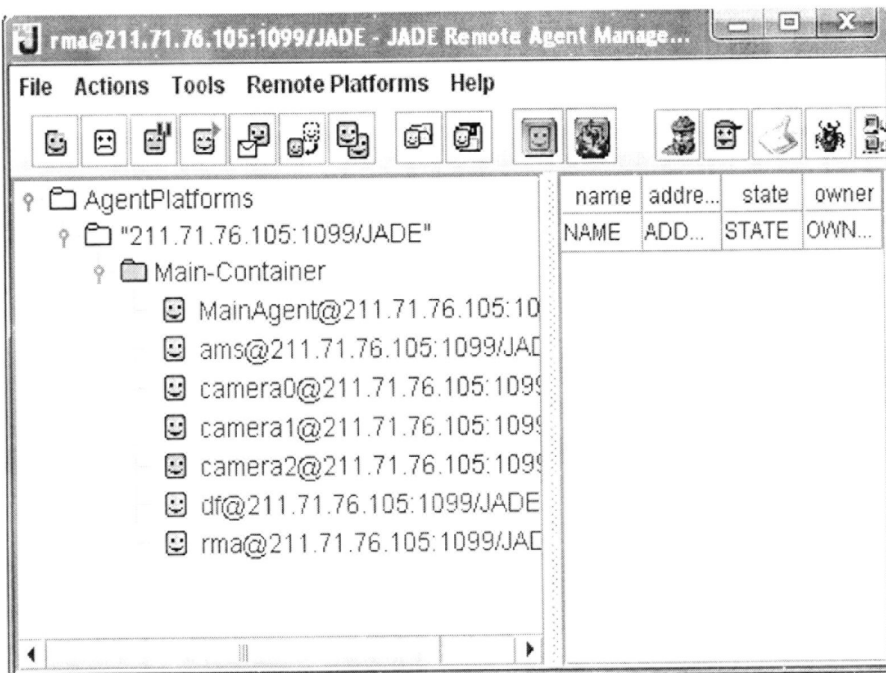


图 4-16 JADE Agent 界面

Fig.4-16 The interface of JADE Agent

4.5 本章小结

在这一章中，我们主要介绍如何使用各种平台和技术实现基于 Agent 的多目视觉人脸特征提取系统。首先详细介绍了开发 Agent 的平台 JADE，描述了在 JADE 下 Agent 怎样实现自己的行为 and 怎样与其他的 Agent 进行交互。然后介绍了整个系统的大致流程图，各个不同功能的 Agent 如何构造，他们之间如何交互，交互的规则等关键问题。我们利用人脸姿态估计得到的参数作为各个 Agent 之间竞争交互的基础。在平台的可移植性下，利用 JNI 标准的特性实现 C 平台下的代码功能在 java 平台的共享和复用。本章还介绍了 Agent 系统下的关键代码并展示了系统的实验界面。

5 实验结果与分析

5.1 实验设计

为了验证本文提出的基于 Agent 的多目视觉人脸特征提取方法，我们用多目视觉采集多组人脸图像，每组为三幅图像，如图 5-1 显示了其中一组图片，为了便于认知，我们给他们取名字分别为人脸 1，人脸 2 和人脸 3。然后用 Agent 系统对系统进行调度和管理，在遵循我们提出的一系列 Agent 交互规则的约束下，系统可以得到一组人脸图像中姿态比较好的图片，然后储存这幅图片的人脸特征作为人脸识别的基础。

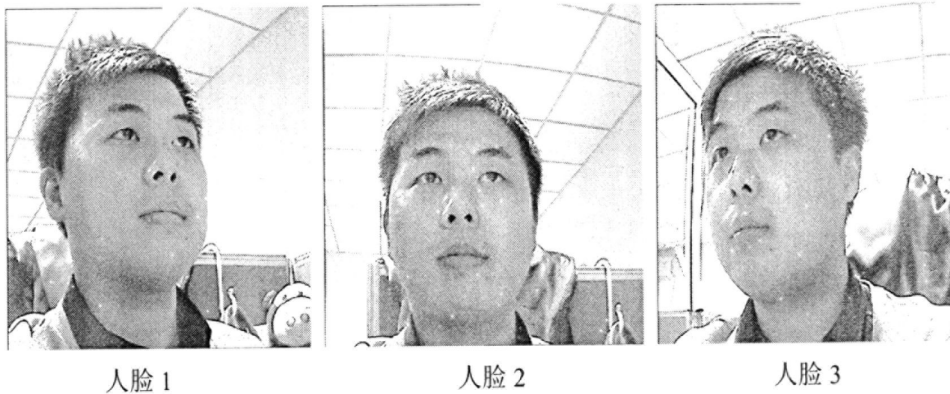


图 5-1 多目视觉采集的图像

Fig.5-1 The pictures of multi camera system

接着我们进行图像识别率对比试验，首先使用多目视觉中的任意一个摄像头提取的人脸特征进行人脸识别率计算，然后用多目视觉在 Agent 交互下得到的人脸特征进行识别率计算，两者的识别结果相对比可以验证本文方法的有效性。Agent 平台的另一个特性就是它的并行处理，也就是在时间处理上给系统带来的改善。这里我们为了验证系统在处理时间上的优势，将分别计算多目视觉中每一个人脸特征提取所需要的时间和 Agent 系统对多目视觉协调进行人脸特征提取所需要的时间，这两者的结果的分析将验证处理时间的优势。

5.2 实验结果和分析

多目视觉使得我们从感知角度获得了更多关于人脸的信息，接下来就是如何去处理应用这些信息去提取到比较理想的人脸特征。在本文的 Agent 框架工作下，对各个图片

的人脸特征进行提取，对图 5-1 的三幅人脸图片，实验 Agent 框架去感知三幅图片，并对图片环境进行反应，利用前面提到的人脸姿态参数计算方法，可以计算得到的三幅人脸姿态参数的值如表 5-1 所示。MainAgent 最后从各个 ControlAgent 在竞争机制下得到的最优人脸姿态(Value = 33.31478)的图片,为人脸 2。这里我们为了更直观的观察人脸姿态，将每张图片人脸区域和正常人脸三维人脸坐标轴画出，如图 5-2 所示。可以发现由于角度的原因人脸 1 和人脸 3 对人脸特征的提取并不理想，相比较而言，人脸 2 中人脸角度偏转的角度要较小，故 MainAgent 会将人脸的人脸特征参数存储起来，作为下一步人脸识别的基础。

表 5-1 各人脸姿态参数值

Table 5-1 The pose parameter value of all faces

属性	值			
目标	Pitch	Yaw	Roll	Value
人脸 1	-9.76715020	18.48283350	21.072994	49.32297650
人脸 2	0.484889	16.411161	16.418734	33.314784
人脸 3	10.282576	16.457791	19.563668	46.304034

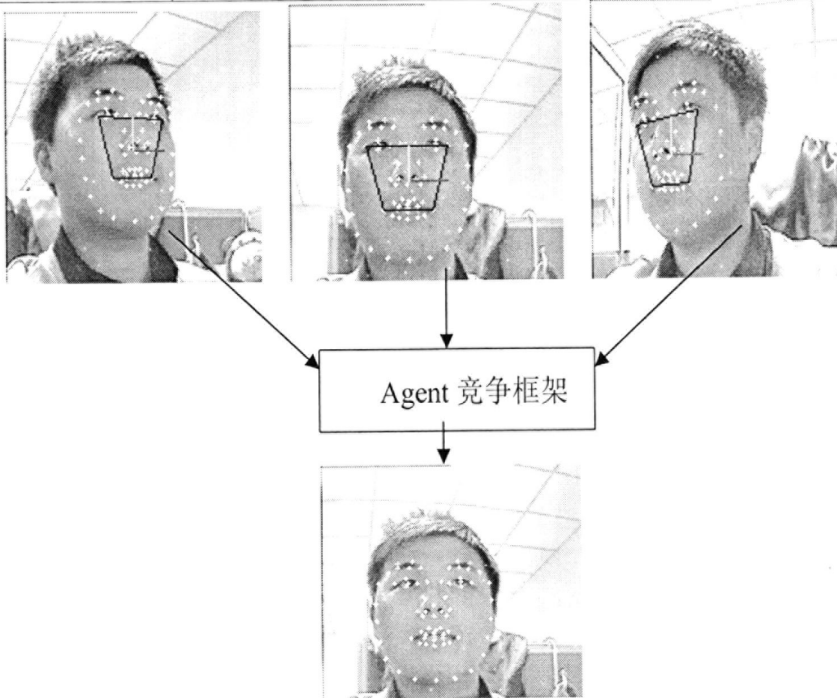


图 5-2 Agent 框架提取结果

Fig.5-2 The extraction result of Agent framework

为了验证通过多目视觉 Agent 框架提取的人脸特征对解决角度问题带来识别率问题有效性，我们分别使用多目视觉系统采集 40 组不同角度的人脸图像，用 ORL 人脸库进

行识别率计算，然后对其中的一个摄像机采集的图像提取的人脸特征也通过 ORL 人脸库进行人脸识别率计算。两者的识别的结果如表 5-2 所示。我们可以发现通过多目视觉 Agent 系统之后识别率有了很大的提高，说明了本文方法对解决人脸识别角度问题的有效性。

表 5-2 单目角度和 Agent 平台识别率

Table 5-2 The recognition rate of one camera and Agent platform

人脸库	测试图像	10 个	30 个	40 个
ORL	单目角度 3 幅	0.9000	0.8556	0.7500
	Agent 平台 3 幅	0.9333	0.9222	0.8500

由于 Agent 平台是一个并行平台，每个 Agent 都是一个独立的个体，因此所有的 Agent 都是同时运行和进行处理的。在本文的方法中每个 Agent 即相当于是一个独立的线程，在多线程的处理环境下，系统的运行速度有了很大的提高。表 5-3 是每幅图像进行人脸特征提取所需要的时间和在本文 Agent 平台下对三幅图像人脸特征提取所需要的时间。当分别对三幅图像进行人脸特征提取时一共需要 $1534+1668+1822=5024\text{ms}$ ，而在 Agent 并行平台下对三幅人脸特征进行提取并经过竞争机制处理得到一幅最好姿态的人脸特征只需要 1965ms ，其中已经包含了各个 Agent 进行交互的处理时间。很显然 Agent 平台提高了多目视觉进行人脸特征提取的处理速度。

表 5-3 每幅图片和 Agent 框架的特征提取时间

Table 5-3 The time of feature extraction of one camera and Agent platform

目标	人脸 1	人脸 2	人脸 3	Agent 平台
处理时间 (ms)	1534	1668	1822	1965

5.3 本章小结

本章通过实验验证了通过多目视觉 Agent 框架提取的人脸特征对解决角度问题带来识别率问题有效性。首先用多目视觉采集多组图片，然后通过 Agent 框架得到系统选出的姿态比较好的人脸图片，储存这个图片的人脸特征。使用 ORL 人脸库对选取的人脸特征进行识别率计算，并且和单目提取的人脸特征计算得到的识别率进行对比，可以证明通过 Agent 系统后的提取的人脸特征对识别率有了很大的改善。实验还验证了 Agent 并行平台对系统处理时间的改善。实验证明了本文提出的方法的有效性。

6 总结与展望

人脸识别技术是当前模式识别和人工智能领域一个重要的研究课题，它的研究跨越了图像处理，模式识别，计算机视觉和人工神经网络等等学科。在安全应用上它广泛应用于国家安全、公安系统和城市的公共安全等各个方面。特征提取是人脸识别研究中的最基本问题之一，是人脸识别中复杂且极具挑战性的环节，对人脸的识别率起着至关重要的作用。在这一领域的研究中，提出了许多经典的算法，如线性鉴别分析方法，基于子空间的主分量分析方法，独立成分分析方法以及在基于核技巧的非线性特征提取方法等。对于特征提取问题的深入研究在很大程度上促进了人脸识别领域的发展。

现有的很多人脸特征提取算法都具有较强鲁棒性，但是他们大部分都是从数学方法上改善了特征提取的质量来提高识别率。本文研究的重点是利用多目视觉在感知阶段最大程度地获取人脸的信息，并通过构建一个 Agent 框架对多目视觉系统进行调度和控制。在各个 Agent 相互合作中通过一系列的行为属性提取出比较好的人脸特征。首先我们对传统的人脸特征提取方法进行深入地探讨，主要研究了基于 ASM 的特征提取方法，并对其经典算法进行实现，通过实验总结这种算法的优点和弊端。然后提出了一种利用多目视觉解决因为角度和姿态等原因引起的人脸信息匮乏的方法，利用多个摄像机从不同的角度去感知人脸的环境，极大丰富了所得到的人脸信息，为下一步人脸特征提取奠定基础。接着提出了一种基于 Agent 的多目视觉人脸特征提取框架。利用 Agent 的协作性和智能性对每个摄像头进行代理，每个 Agent 通过感知环境得到要代理的人脸姿态的参数，并对环境进行反应，通过一定的规则计算出人脸姿态参数，而多个 Agent 之间通过竞争机制去成为最优的人脸姿态目标，由最优的人脸姿态图像得到最终的人脸特征，从而满足整个 Agent 系统对人脸特征较好提取的要求。最后通过试验表明本文方法能较好地每组图片中提取出较好人脸姿态的图片，并从中提取人脸特征，解决了由于角度问题带来的误识别问题，并且在处理时间上也有很大的优势。

本文在说明了人脸特征提取的目的及意义的基础上，展望了特征提取的前景和面临的挑战。将各种不同的方法综合运用，发挥各自的优势来进行特征提取将成为这一领域的发展趋势。本文深入研究的基于 Agent 的多目视觉人脸特征提取方法具有重要的理论意义和实际应用价值。由于现在对基于 Agent 的特征提取框架研究只是处于起步阶段，需要做的事情还很多。

(1)本文提出的基于 Agent 的特征提取框架大部分还停留在定性的基础上，如何在一定的数学理论上建立更形式化的模型，并且将知识库和控制策略更好地融合在一起需要进一步的研究。

(2)本文提出 Agent 交互机制中，只是简单地从多幅人脸图片中选出一幅姿态最好，

也就是特征点提取最好的，但是多目视觉得到的广泛的人脸信息没有得到充分的应用。因此如何提出一种人脸信息融合算法，将多幅人脸中有用的信息集合到一起是一个很大的课题。

(3)本文提出的基于 Agent 的特征提取框架更侧重于一种 Agent 应用的思想，没有系统地对不同功能的 Agent 进行详细构造。如何对框架的 Agent 体系结构更加详细的分析，对各个 Agent 性能如何分配也是值得深入研究的问题。

参考文献

- [1] R.Chellappa, C.L.Wilson, S.Sirohey. Human and machine recognition of faces. a survey[J]. Proceeding of the IEEE. 1995.83(5).705-741.
- [2] 山世光.人脸识别中若干关键问题的研究[D].北京.中国科学院计算技术研究所. 2004.1-30.
- [3] M.H.Yang, D.J.Kriegman, N.Ahuja. Detecting faces in images. a survey[J]. IEEE Trans. Pattern Analysis and Machine Intelligence. 2002.24(1).34-58.
- [4] Cootes T F, Taylor C J. Locating Faces Using Statistical Feature Detectors[C]. Proceedings of 2nd International Conference on Automatic Face and Gesture Recognition. Killington. 1996.204-209.
- [5] Young, Fukunaga. The reliability of linear feature extractor[J]. IEEE Trans. Computers. 1971.20(9).967-971.
- [6] Kirby M, Sirovich L. Application of the KL procedure for characterization of human faces[J]. IEEE Trans. Pattern Analysis and Machine Intelligence. 1990.12(1).103-108.
- [7] Turk M, Pentland A. Eigenfaces for recognition[J]. Journal of Cognitive Neuroscience. 1991.3(1).71-86.
- [8] J.F.Cardoso. High-Order contrasts for independent component analysis[J]. Neural Computation. 1999.11(1).157-192.
- [9] M.S.Bartlett, J.R.Movel. Face recognition by independent component analysis[J]. IEEE Trans. Neural Network. 2003.13(6).1450-1464.
- [10] Fisher, R.A.. The use of multiple measurements in taxonomic problems[J]. Annals of Eugenics. 1936.7.178-188.
- [11] Foley D H, J.W.Sammon. An optimal set of discriminant vectors[J]. IEEE Trans. Computers. 1975.24(3).281-289.
- [12] X.S.Zhuang, D.Q.Dai. Improved discriminant analysis for high-dimensional data and its application to face recognition[J]. Pattern Recognition. 2007.40(5).1570-1578.
- [13] F.R.Bach, M.I.Jordan. Kernel independent component analysis[J]. Journal of Machine Learning Research. 2002. 3.1-48.
- [14] V.N.Vapnik. 统计学习理论的本质(译本)[M]. 北京. 清华大学出版社. 2000.
- [15] S.Mika, G.Ratsch, J.Weston, B.Scholkopf. Fisher discriminant analysis with kernels[C]. IEEE International Workshop on Neural Networks for Signal Processing IX. Madison. USA. 1999.41-48.
- [16] B.Scholkopf, S.Mika, C.Burges, P.Knirsch, G.Ratsch,, A.Smola. Input space vs. feature space in kernel-based methods[J]. IEEE Trans. on Neural Network. 1999.10(5).1000-1017.
- [17] J.Yang, A.F.Frangi, J.F.Yang. KPCA plus LDA. a complete kernel fisher discriminant framework for feature extraction and recognition[J]. IEEE Trans. on Pattern Analysis and Machine Intelligence. 2005.27(2).230-244.
- [18] Y.Xu, J.Y.Yang, J.Yang. A reformative kernel fisher discriminant analysis[J]. Pattern Recognition. 2004.37.1299-1302.
- [19] 徐蓉, 姜峰, 姚鸿勋. 流行学习概述[J]. 智能系统学报. 2006.1(1).44-51.
- [20] H.S.Seung, D.D.Lee. The manifold ways of perception[J]. Science. 2000.vol.290.2268-2269.

- [21] J.B.Tenenbaum, V.deSilva, J.C.Langford. A global geometric framework for nonlinear dimensionality reduction[J]. Science. 2000.vol.290.2319-2323.
- [22] S.T.Roweis, L.K.Saul. Nonlinear dimensionality reduction by locally linear embedding[J]. Science. 2000.vol.290.2323-2326.
- [23] X.He, S.Yan, Y.Hu, H.Zhang. Learning a locality preserving subspace for visual recognition[C]. Proceeding of ninth International Conference on Computer Vision. France. 2003.10.385-392.
- [24] X.He, S.Yan, Y.Hu, Niyoge, H.Zhang. Face recognition using Laplacianface[j]. IEEE Trans. on Pattern Analysis and Machine Intelligence. 2005.27(3).328-340.
- [25] H.T.Chen, H.W.Chang, T.L.Liu. Local discriminant embedding and its variants[C]. IEEE Conf. Computer Vision and Pattern Recognition. 2005.846-853.
- [26] H.A.Rowley, S.Baluja, T.Kanade. Neural network based face detection[J]. IEEE Trans. on Pattern Analysis and Machine Intelligence. 1998.20(1).23-28.
- [27] Kass M, Witkin A, Terzopoulos D. Snakes: Active contour models[J]. International Journal of Computer Vision. 1987.1(4).321-331.
- [28] 肖秀春. 人脸检测与面部特征提取技术研究. [学位论文].长沙.湖南大学. 2004.
- [29] G.C.Feng, P.C.Yuen. Variance projection function and its application to eye detection for human face recognition[J]. Pattern Recognition Letters. 1998.19(9).899-906.
- [30] BRUNELLI R, POGGIO T. Face recognition: Features versus templates[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1993.15(10).1042-1052.
- [31] Cootes T F, Taylor C J, Cooper D H. Active shape models-their training and application[J]. Computer Vision and Image Understanding. 1995.61(1).38-59.
- [32] Yuille A L, Cohen D S, Halinan P W. Feature extraction from faces using deformable templates[C]. In. Proceedings of IEEE Computer Soc. Conf. on computer Vision and Pattern Recognition. Washington. DC. USA. 1989.104-109.
- [33] D.Y.Reisfeld. Robust detection of facial features by generalized symmetry[C]. In. Pattern Recognition. 1992.Vol.I. Conference A. Computer Vision and Applications. Proceedings.. 11th IAPR International Conference on. The Hague, Netherlands.
- [34] P.W.Hallinan. Recognizing Human Eyes. in SPIE Proceedings. Geometric Methods In Computer Vision.1991.
- [35] A.M.Pentland, B.Starner, T.View. View-based and modular eigenspaces for face recognition. in Proceedings CVPR 94. IEEE Computer Society Conference on 1994. Seattle. WA.
- [36] T.F.Cootes, C.J.Taylor. A mixture model for representing shape variation. In A. Clarke. editor, 8th British Machine Vision Conference. 1997.9.110-119.
- [37] T.F.Cootes, C.J.Taylor. Constraint Active Shape Models[C]. International Conference on Digital Object Identifier. 2001.1.748-754.
- [38] T.F.Cootes, C.J.Taylor, A.Lanitis. Active Shape Models. Evaluation of a Multi-Resolution Method for Improving Image Search[C]. British Machine Vision Conference. 1994.327-336.
- [39] G.Hamarnah, T.Gustavsson. Combining Snakes and Active Shape Models for Segmenting The Human Left Ventricle in Echo Cardio Graphic Images[C]. IEEE Computers in Cardiology 2000.115-118.
- [40] Fei Zuo. Fast Facial Feature Extraction Using a Deformable Shape Model with Haar-wavelet Based Local Texture Attributes. ICIP'04 International Conference on Image Processing. 2004(3).1425-1428.

- [41] Shuicheng Yan, Ce Liub, Stan Z.Lib, Hongjiang Zhangb, Heung-Yeung Shumb, Qiansheng Cheng. Face Alignment Using Texture-constrained Active Shape Models. *Image and Vision Computing*. 2003.69-75.
- [42] Jilin Tu, Zhenqui Zhang, Zhihong Zeng. Face Localization Via Hierarchical CONDENSATION With Fisher Boosting Feature Selection. *Computer Vision and Pattern Recognition*. 2004(2).719-724.
- [43] Feng Jiao, Stan Li, Heung-YeungShum, Schuurmans D.. Face Alignment Using Statistical Models and Wavelet Features. *Computer Vision and Pattern Recognition*. 2003(1).321-327.
- [44] Li Zhang, Haizhou Ai, Shengjiun Xin, Chang Huang, Shuichiro Tsukiji, Shihong Lao. Robust Face Alignment Based on Local Texture Classifiers. *ICIP 2005 IEEE International Conference on Image Processing*. 2005(2).354-357.
- [45] M.Minsky. *The Society of Mind*. Simon and Schuster[M]. New York. 1985.
- [46] 程显毅. *Agent 计算*[M]. 哈尔滨.黑龙江科学技术出版社. 2003.
- [47] 王琰. 基 MAS 环境的 Agent 交互与协作策略的研究[D]. 成都.电子科技大学. 2003,03.
- [48] R.A.Brooks. Intelligence without representation[J]. *Artificial Intelligence*. 1991.47.139-159.
- [49] 陈幼平, 赫惠清, 娄平等. 基于多 Agent 的运输任务分配系统德研究与开发. *设计与研究*. 2004.
- [50] 刘向军, 刘世平, 张洁. 多 Agent 系统通信与协作机制构造[J]. *机械设计与制造工程*. 2002.21(2). 40-42.
- [51] 程显毅, 陈小波. 基于多 Agent 的模式识别框架[J]. *智能系统学报*. 2006.1(2).89-93.
- [52] Fabio Bellifemine, Giovanni Caire, Dominic Greenwood. *developing multi-Agent system with JADE*[M]. West Sussex. WILEY. 2007.
- [53] 刘大有, 杨鲲, 陈建中. Agent 研究现状与发展趋势[J]. *软件学报*. 2000. 11(4).315-321.
- [54] A Gee, RCipolla. Estimating gaze from a single view of face[C]. *International Conference on Pattern recognition*. Vol94. 1994.A.758-760.
- [55] Akitoshi Tsukamoto, Chil-Woo Lee, Saburo Tsuji. Detection and pose estimation of human face with synthesized image models[C]. *International Conference on Pattern recognition*. Vol94. 1994.A.754-757.

作者简历

本人概况:

朱延飞, 男, 汉族, 1987年11月23日出生于江苏省扬州市宝应县。

教育经历:

- 本科: 2006.9-2010.7 中国地质大学(北京) 地球物理与信息技术学院, 获得 测控技术与仪器专业学士学位。
- 硕士: 2010.9-2012.7 北京交通大学 计算机与信息技术学院, 获得 信号与信息处理专业硕士学位。

撰写文章:

《Face Feature Extraction Based on Agents with Multi-camera System》


被《International Journal of Information and Computer Science》录用, 拟发表在第2012年第2期。

科研及项目经历:

[1] 基金资助: 受到国家自然科学基金项目(NO. 60973060)的资助。

独创性声明

本人声明所提交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京交通大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名： 签字日期：2012年6月20日

学位论文数据集

表 1.1: 数据集页

关键词*	密级*	中图分类号*	UDC	论文资助
特征提取, Agent 理论	公开	TP391.4	620	
学位授予单位名称*		学位授予单位代 码*	学位类别*	学位级别*
北京交通大学		10004	工学	硕士
论文题名*		并列题名		论文语种*
基于 Agent 的多目视觉人脸特征提 取				中文
作者姓名*	朱延飞		学号*	10120416
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京交通大学		10004	北京市海淀区西 直门外上园村 3 号	100044
学科专业*		研究方向*	学制*	学位授予年*
信号与信息处理		计算机视觉	2.5 年	2012 年
论文提交日期*	2012-6-15			
导师姓名*	阮秋琦	职称*	教授	
评阅人	答辩委员会主席*		答辩委员会成员	
	朱维彬		张树军 杨唐文	
电子版论文提交格式 文本 () 图像 () 视频 () 音频 () 多媒体 () 其他 () 推荐格式: application/msword; application/pdf				
电子版论文出版(发布)者		电子版论文出版(发布)地		权限声明
论文总页数*	69			
共 33 项, 其中带*为必填数据, 为 22 项。				