学校代号 10532　　学　　号 S1410W0678

分 类 号 TP391　　密　　级 普通

湖南大学
HUNAN UNIVERSITY

工程硕士学位论文

一种类人机器人手势识别算法及其实现

学位申请人姓名 卢兴沄

培 养 单 位 信息科学与工程学院

导师姓名及职称 李仁发 教授　唐涛 高级工程师

学 科 专 业 软件工程

研 究 方 向 人机交互

论文提交日 期 2017 年 05 月 10 日

学校代号：10532

学　　号：S1410W0678

密　　级：普通

湖南大学硕士学位论文

一种类人机器人手势识别算法及其实现

学位申请人姓名：　卢兴沄

导师姓名及职称：　李仁发 教授　唐涛 高级工程师

培　养　单　位：　信息科学与工程学院

专　业　名　称：　软件工程

论 文 提 交日 期：　2017 年 05 月 10 日

论 文 答 辩日 期：　2017 年 05 月 21 日

答辩委员会主席：　邝继顺 教授

A Hand Gesture Recognition Algorithm For Humanoid Robot And Its Realization

by

Lu XingYun

B.E. (Hunan University of Science and Technology) 2014

A thesis submitted in partial satisfaction of the

Requirements for the degree of

Master of Engineering

In

Software Engineering

in the

Graduate School

Of

Hunan University

Supervisor

Professor LI Renfa

May, 2017

湖 南 大 学

学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果由本人承担。

作者签名：　　日期：　　年　月　日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权湖南大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于

1、保密□，在_____年解密后适用本授权书。

2、不保密□。

（请在以上相应方框内打“√”）

作者签名：　　日期：　　年　月　日

导师签名：　　日期：　　年　月　日

摘　要

为了满足人类社会生产技术从机械化向智能化发展，智能机器人技术被作为一个突破口受到越来越多国家的重视，同时大量的商业团体将先进的智能机器人推向市场。为了满足机器人的“智能”升级，越来越多的传感器和其它硬件设备被搭载到智能机器人上。在这种背景下，智能设备的“先进”和人机交互方式的“落后”形成巨大的矛盾。手势交互作为一种既传统又新颖的交互方式适用于目前人机交互的需求。手势识别系统作为手势交互的核心问题，其设计与实现是一项重要的研究课题。通过对当前国内外对手势识别的研究工作分析发现有以下几点不足：其一，算法对实验环境的要求高，不能脱离特定的实验环境，很难满足现实的需求；其二，目前大量的手势识别算法还是基于传统的机器学习方法，这些算法的性能已经陷入瓶颈，而目前热门的深度学习方法为提高手势性能提供了可能；其三，绝大多数研究是在 PC 平台完成，没有考虑嵌入式平台的性能瓶颈，其算法对实验载体的性能要求很高，嵌入式平台很难满足这些性能需求。本文针对以上不足，进行了如下研究工作：

首先，设计并实现了一种基于肤色信息和运动信息相融合的手势分割算法。手势分割技术面临诸多难题，各种常用的手势分割方法都有各自的缺点，任一种很难满足复杂多变的现实情况。本文分析了两种手势分割技术的优点和缺点，以及两种方法结合的可能性，最后实现了一种将肤色信息和运动信息融合起来的手势分割方法，这种手势分割方法可以满足复杂背景条件下的手势识别。

其次，实现了一种基于卷积神经网络的手势识别算法。卷积神经网络具有直接读入图片提取特征的能力，可以避免人工设计手势特征所耗费的大量人力和需要的丰富经验。本文参考卷积神经网络算法在人脸识别领域的研究成果，根据手势和人脸的异同来进行改进，对卷积网络结构进行了重新设计，并用公共数据集来对网络进行训练，最后成功实现了卷积网络对手势的识别和分类。

最后，在 NAO 机器人平台上完成手势识别系统的移植，对移植后的手势算法进行优化，在嵌入式平台得到不错的识别效果。

关键词：智能机器人；手势交互；手势识别；手势分割；卷积网络；NAO 机器人

Abstract

In order to meet the social productive forces' demand for intelligent development, more nations have paid attention to intelligent robot technology. In the meantime, many business organizations get their intelligent robot products into the market. For purpose of upgrading those intelligent robots, the number of sensors and other devices is increasing. Human-computer interactions can't able to meet the demand of intelligent devices. Currently, gesture interaction as an original but innovative interactive mode which is fit to Human-Computer interaction. Gesture recognition as a very important issue of gesture interaction, the implementation of its system is also an important research subject. Based on the research results both abroad and at home, the method of gesture recognition have many defects, mainly includes the following aspects: First, These algorithms are so outdated. There are many traditional machine learning algorithms rather than these most popular Deep learning algorithms; Second, These algorithms needs fixed experimental environment. It is difficult to meet the needs of reality; Third, These algorithms are based on PC platform. They do not to account for embedded platform. To solve the above problems, we carried out the following research work:

First, combined with real requirement, this paper sets out design and realization of a method of gesture segmentation based on skin color information and movement information. In consideration of almost all of gesture segmentation method have their own defect, neither of them can fit to various real environment. This paper present a method, which combines skin color information and movement information, is of good robustness against various environment.

Then, aiming at the excellent performance of CNNs, it presented an improved CNNs algorithm to solve the difficulty of artificial design features, and redesign the network structure. Experimental results show the performance of recognition and classification are so well.

At last, we transplant the gesture recognition algorithm to Nao.

Key Words: Intelligent Robot; Gesture Interaction; Gesture Recognition; Gesture Segmentation; CNNS; NAO Robot

目 录

插图索引

附表索引

第1章 绪论

1.1 课题研究背景及意义

随着工业 4.0 时代的到来，传统制造业开始发生一场大变革，工业机器人将成为智能制造的主力军，同时智能机器人作为智能设备的代表也开始得到各国重点关注。从最早实验室研究，到如今开始进入商用应用阶段，机器人在制造行业、服务行业得到了广泛应用，甚至以“朋友”的身份进入普通消费者家庭，离不开其“智能”的升级。正是具备了这种“智能”，机器人与人类的交互才逐渐变得友好、自然。智能机器人，智能的核心是“计算机”，具有学习能力正是机器人能够胜任替代人类工作的重要原因。

人与智能机器人的交互实质是人与计算机的交互。人与计算机的交互主要有三种，从出现的先后顺序来看，最早是使用命令行，通过物理键盘在命令行界面上输入各种命令来控制计算机。在完成比较复杂的操作时，操作人员要与计算机进行频繁且大量的“交互”，需要耗费大量的精力来记住这些操作指令。因此该方式显得很复杂，只适合具备一定能力的专业人员。伴随着计算机软硬件的发展，鼠标这种全新的输入方式开始广泛使用，图形交互界面也应运而生。图形交互界面主要由窗口、菜单和图标等元素构成，能够输出更为丰富的静态或动态的图形图像，使人机交互方式更为友好，交互效率更高，使得普通人也可以成为计算机用户。但是随着虚拟现实、增强现实等技术迅猛发展以及可穿戴计算机的广泛应用，无论是鼠标、键盘的输入组合，还是可触摸屏幕的触摸交互越来越显示出局限性，因此图形交互界面这种人机交互方式已经逐渐不能满足用户的需求。尤其是人机交互方式的需求逐渐从“以机器为中心”转移到“以人为中心”，多通道交互已经变为人机交互领域的新焦点。多通道交互即把语音、手势、表情、眼动等多种方式蕴含的信息以某种合理方式混合后加入输入通道。多通道交互以贴近人类交流习惯为出发点，将“计算机”看作人，能让人与机器的交互变为人与“人”的交互。这种交互方式更符合人类的交流习惯，能使用户更加自然舒适的完成交互任务。这些新的交互方式是建立在新的交互技术之上，其中手势交互就是其中一个比较热门的人机交互方式。

手势交互离不开人类长期以来对手势语言使用的经验积累。手势语言是人们通过用手势动作和视觉进行交互的语言，拥有很长的历史。手势并非专为聋哑人所用，古人很早就开始通过手势来进行沟通，例如在公开场合隐晦地传递自己观点，在容易因震动而出现崩塌事故的山区进行交流，或者在战争中通过手语交流

躲避敌人的侦查。手势语言具有同书面语言和语音相当的表达能力，在一定场合手势交互具有其他交互方式难以企及的优点，因此可将手势作为人机交互的接口。手势交互这种种目前很热门的交互方式，其基础问题手势识别问题已成为了热门研究问题。

综上所述，本文从某些特殊场合：危险区域、需要安静的场合，水下、太空等不便直接操控的特殊场合的需求出发，将手势作为一种方便、快捷且可远距离控制的交互方式在人与智能机器人交互领域中进行研究，提出一种面向类人机器人的手势识别方法，并将其移植到类人机器人平台，为后续更深入的研究奠定了基础。

1.2 研究问题

手势识别技术发展到现在，主流的研究方向是基于机器视觉的手势识别研究。基于机器视觉的手势识别，先通过摄像机采集手势图像序列，然后对图像处理并分析，从而识别手势。该种方式识别率相对较低，实时性差，对计算机设备的计算性能要求较高。但是随着计算机性能的提高以及硬件价格的下降，硬件性能已经满足应用的需要，使得研究难度较大的基于视觉的手势识别开始成为研究热点。考虑到未来大量的智能设备将进入市场，人机交互的高通用性是一个很大的需求，基于视觉的手势识别技术更能满足这一需求，因而基于机器视觉的手势识别更具有研究价值。

通常基于视觉的手势识别实现的步骤为[1]：采集图像，输入图像经手势分割进行分离，定位出动态手势；然后使用手势分割算法提取出手势区域，根据手势模型从手势区域提取出手势参数；最后，根据将手势参数输入识别算法进行手势识别。其中有三个研究的热门问题，分别是：手势分割、特征提取和手势识别。

1.2.1 手势分割

手势分割的前提是手势定位。手势识别的有效区域是人的手掌区域，此外其他区域都是无效区域。复杂的环境因素将会对手势定位造成较大的干扰，如果没有一种好的方法将有效区域找出来，将会浪费大量的计算资源在无效区域上。目标检测和手势跟踪能够对手势进行粗略定位，快速得到“感兴趣”区域。目标检测的热门研究方法是背景减法，常见的背景减法有简单自适应背景减法和基于混合高斯模型的背景减法。简单自适应的背景减法在复杂背景下对运动的检测能力较差；基于混合高斯模型的背景减法抗干扰能力强，建模得到的背景稳定性更好。但是考虑到混合高斯模型在建模一定时间后，部分参数逐渐达到某种稳定状态，此时如果光照发生突然变化或者事物突然发生运动，混合高斯模型（GMM）的参数难以跟上场景中真实背景变化，从而可能检验结果为伪目标。针对上述问题，

周建英[2]等提出将滑动窗技术的短暂历史记忆特性与混合高斯模型相结合，通过设置不同滑动窗窗长和移动步长来灵活控制高斯模型对历史信息的遗忘速度，使建模得到的背景模型能够准确的反映真实背景的实时动态。针对跟踪算法在手势姿态变化、目标遮挡和外界干扰等因素影响下的性能较差，严权峰[3]等提出一种基于压缩感知的实时手势检测和跟踪算法，该算法将结合肤色模型的 Adaboost 算法得到的手势位置与压缩感知得到的跟踪结果进行一致性测量，最后得到最终手势位置信息，以实现手势跟踪自动初始化和跟踪错误后自我恢复。

手势分割是指将有意义的手势区域从包含手势的图像出提取出来的过程。但是在基于视觉的手势识别中，手势分割存在相当的难度。目前比较成熟的分割技术有基于肤色检测的手势分割方法、基于轮廓模型的手势分割方法和基于运动检测的手势分割方法。基于肤色模型的手势分割是最常见的手势分割方法，建立肤色模型首先要考虑颜色空间的选取。针对 RGB 颜色空间中各分量包含亮度信息，而 RGB 值容易受到环境光照影响而发生改变，冯志全[4]等提出可以选取肤色的亮度信息作为索引建立肤色模型。刘军[5]等提出将原始图像从 RGB 颜色空间转换到HIS颜色空间进行表示，再利用非参数化的颜色直方图得到人体肤色的聚类特征，最后根据肤色的 HIS 色彩范围对原始图像进行提取手势。

国外对于基于肤色检测的手势分割也有许多研究。文献[6]提出了一种基于 RGB 颜色空间的肤色建模方法，通过基于混合高斯肤色模型和基于直方图肤色模型相结合，获得了更好的肤色分割效果。文献[7]采用规定肤色范围的方法，对各参数给定一个阀值，在范围内判定为肤色目标，来建立肤色模型。文献[8,9]分别采用椭圆边界模型和高斯模型对肤色分布进行建模，这类利用边界模型的方法分割的时间复杂度很低，但分割的准确度也很低。文献[10]采用混合高斯模型建立肤色模型进行手势分割。

1.2.2 特征提取

特征提取是识别问题的必须步骤，根据问题的不同需要的特征也不尽相同。目前对于手势识别的特征提取研究，大多数方法还是采用人工设计手势特征。对于手势识别问题，常见的手势特征有：手指个数，指尖坐标和掌心坐标等。Lahamy[11]提出一种“U”型轮廓特征，但是在实验结果中发现大量识别错误是因为并非所有的手势都有明显的“U”型轮廓特征，该种特征的特点也使得手势类型数量受限。Z Ren[12]等针对手势轮廓提出一个新的概念 FEMD（Finger-Earth Mover's Distance），并通过提取手势边缘和掌心的相对距离曲线特征，用模板匹配的方法来对手势进行识别。李丹娇[13]等提出将 CSS 形状描述子与傅立叶描述子相结合，利用 CSS 形状描述子在曲率过零点反映局部形状信息和利用傅立叶描述子的低频系数描述的手势轮廓整体信息分别作为特征进行距离度量，

再将这两种距离加权计算得到一个新的距离来表征手势的差异程度。从上述文献可见，人工设计特征需要花费很多心思和做很多工作，其过程非常耗时耗力，且需要有丰富的专业知识和经验才能确定出能够用于正确分类的特征。近年来随着深度学习的兴起，卷积神经网络（CNN）等深度学习的方法在人脸识别、自然语言处理等领域取得巨大成功，深度学习方法也开始进入手势识别领域。深度学习带来了新的研究思路，从人工设计特征向特征学习转变，这是一种特征学习方法。它通过把原始数据经过一些简单但非线性的模型转变成更为高层次更加抽象的表达。例如卷积神经网络不需要人工设计特征，它能够将图像直接输入网络并在输出端得到分类结果[14]。

1.2.3 手势识别

无论采用何种视觉硬件系统，确定其背后的手势模型是手势识别问题的重点，手势模型的多样性决定了识别方法的多样性，传统的手势识别方法有：隐马可夫模型（HMM）、神经网络、基于时间规整法和多信息融合法等。近年来比较热门的识别方法是隐马可夫模型和神经网络模型。隐马可夫模型是一种基于概率统计的方法，虽然其在语音识别研究方面取得了巨大的成功，将其运用到手势识别上有着难以克服的缺点：初始化过程太复杂，跟踪和初始化需要分别进行，导致计算量特别巨大。其复杂具体表现为：每种手势分别建立 HMM 模型，使系统实时性很低。严焰[15]等利用隐马可夫模型对手势指令建模，并用 Kmeans 算法来提高手势识别性能。张静[16]等针对传统 HMM 的缺点采用 CRF 算法完成手势识别，识别率有较显著的提高。赵新龙[17]等利用基于 BP 神经网络的方法设计出一套编辑手势，可以实现对计算机草绘行为的准确理解和对草图快速编辑修改。

国外对手势识别有如下研究。Elmezain[18]利用 Baum-Welch 算法训练隐马可夫模型（HMM）最终完成手势识别。文献[19]使用了一种基于向量化的 HMM 模型，使得在足够的训练样本支撑下，能够处理不同长度的信息且具有很高的识别率。针对神经网络具有高度并行性和自适应性等优点，文献[20]提出 BP 神经网络，实现了数十种手势的识别。但是神经网络有很多缺点，如容易出现局部最小值，收敛速度很慢等。针对以上问题出现了一些改进的算法，文献[21]将 Chebyshev 网络应用到动态手势识别，文献[22]提出基于模糊神经网络拓扑结构的模糊特征值来区分不同的手势。

1.3 研究工作

通过对国内外手势识别系统的设计与实现分析，发现现有的手势识别系统研究方案存在以下几点不足：

(1)现有识别方案大多采用传统 HMM 或传统神经网络，而如 CNN 卷积神

经网络等深度学习的方法已经广泛应用于图像识别处理问题，其他新方法也大量涌现出来，旧方法已经不能满足性能需要。

(2)很多对手势分割的研究中，为了追求分割效果的稳定性，增加了许多限制条件，使得在这种情况下，手势分割算法的适应性很差。

(3)大多数手势识别系统都是以 PC 机甚至性能更好的计算机作为实验硬件平台，可利用的计算资源很多，因此往往没有考虑计算资源不足的情况：在嵌入式设备有限的计算资源情况下如何更好的实现手势识别。

针对以上不足，本文提出了基于 CNN 卷积神经网络的手势识别设计方案，围绕手势识别方法中手势分割和手势识别等问题，进行了相关的研究工作。最后在法国 Aldebaran Robotics 公司研制的 NAO 机器人上进行算法改进和测试，并最终移植到 NAO 机器人上，使其能够独立的完成手势识别并给予反馈。本文主要研究工作如下：

第一，分析了手势分割技术的难点，和常用的两种手势分割方法在实际应用中出现的缺点，针对这些缺点提出一种基于肤色模型和运动信息相融合的手势分割算法。

第二，采用现在热门的深度学习方法，构建一种手势识别系统。对卷积网络的结构进行了设计，构建出一个卷积神经网络对手势进行识别和分类。

第三，将完成的手势识别系统移植到 NAO 机器人上，使其能够成功的运行，实现 NAO 机器人的手势识别。

1.4 结构安排

本文的结构如下：

第一章 绪论

主要介绍人机交互发展趋势以及手势交互发展现状，分析手势作为交互方式的优势，国内外对手势识别研究的现状。最后简述本文研究工作及论文组织结构。

第二章 相关研究

介绍了进行手势识别研究需要涉及的主要问题及相关基础知识，深度学习理论和智能机器人的发展现状。

第三章 手势分割技术研究

分析手势识别主要问题之一的手势分割所存在的难点，提出一种基于肤色模型和运动信息相融合的手势分割算法，并对实验中出现的问题进行分析和改进。

第四章 基于 CNN 神经网络的手势识别研究

介绍了一种常用的运动跟踪算法，并将该算法与手势跟踪相结合进行了一些改进。提出了本文的卷积神经网络结构和设计方法，并完成了手势识别实验。

第五章 类人机器人手势识别系统的实现

介绍了实验的软硬件环境，实验载体 NAO 及其视觉系统。完成手势识别算法到 NAO 机器人上的移植。

结论

对本文所做的工作进行了总结，对今后的研究工作进行了展望。

第2章 相关研究

一个完整的基于机器视觉的手势识别系统主要包括图像采集、手势检测和跟踪、手势分割和手势识别。本章从构建一个完整手势识别系统的步骤出发，分别介绍所需要用到的相关技术。由于本文研究载体是类人机器人，因此在本章最末对智能机器人发展情况进行了介绍。

2.1 手势识别框架

基于机器视觉的图像识别通常有一个流程，如图 2.1，根据这个流程建立起手势分类模型。传统的手势分类模型如图 2.2 所示，而如图 2.3 采用卷积神经网络建立的分类模型将会简化很多，省去了人工特征的设计和提取，但是对需要分类的图像进行预处理仍是必要的。

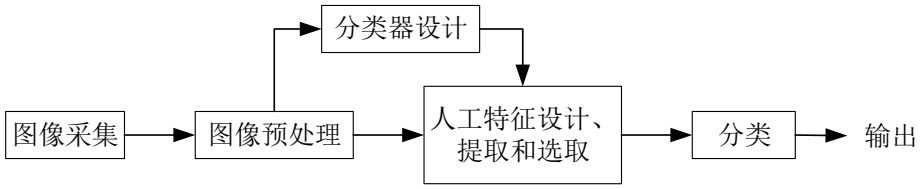


图 2.1 手势识别流程

图 2.2 传统手势分类模型

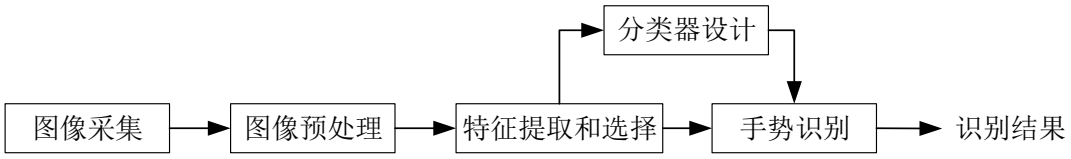


图 2.3 卷积神经网络分类模型

本文基于图 2.3 框架，分别从图像的预处理和识别模型入手，提出各自的解决的方案，这套方案所用到解决问题的方法单独来看也许是次优方法，但是将他们结合起来在性能有限的嵌入式设备手势识别的解决方案上会是最优方法。

2.2 基于视觉的手势识别技术

通过硬件设备采集图像。该步骤为手势识别的第一步，基于机器视觉的手势

识别通常是利用一个或者多个摄像机进行图像采集、捕获手势。基于视觉的手势识别技术根据手势识别系统的硬件复杂程度不同可分为：单目摄像头手势识别系统，多目摄像头手势识别系统。

2.2.1 基于单目摄像头的手势识别

单目摄像头手势识别系统利用图像的二维信息，提取手势的形状、颜色和运动特征。肤色检测法是单目摄像头条件下的主要方法，其关键是选择合适的颜色空间对肤色进行建模，从背景中分离出手势区域。单目摄像头手势系统的优点是面向二维图像，建模简单，数据处理量少，实时性高。其缺点是在非理想条件下，不同的光照条件对肤色模型的影响很大。因此在实际应用中，复杂环境条件下仅用二维信息难以准确划分手势区域。

尽管单目摄像头的手势识别存在不少问题，但是通过对算法的精心设计和改进，也能获得很好的识别效果。考虑到嵌入式设备计算性能有限，基于单目摄像头的手势识别方法是一种比较合适的方法。文献[23]在单目摄像头条件下提出将手势的跟踪与识别有机的统一起来，将手势识别的结果传递给跟踪部分，作为跟踪对象，手势跟踪的预测结果反馈给识别部分，通过预测下一帧手势出现的粗略位置大大降低识别步骤的计算量。该算法在单目摄像头的条件下在嵌入式平台有很好的性能。

2.2.2 基于多目摄像头的手势识别

基于多目摄像头的手势识别系统是将多个摄像头或者多种摄像头进行组合。与单目摄像头只能获取二维信息相比，多目摄像头就像人的双眼，可以通过左右两只眼睛获取的两幅图像的视差来计算确定距离，其优势是可以多获取一个深度信息。近几年随着大量比较廉价的深度视觉传感器进入市场（如 Kinect，Kinect 深度摄像机借助其搭载的深度视觉传感器获取深度信息，利用深度信息来消除肤色模型分割的缺点[24]），利用深度图像[25; 26]信息提取手势区域成为一种较为理想的方式，深度视觉结合普通 2D 视觉的研究方法也开始引起了学界的广泛关注。

(a) (b)

图 2.4 Kinect 深度摄像机

图 2.4 为 MicroSoft Xbox 的 Kinect 套件。Kinect 有三个镜头，中间的 RGB 彩色摄像头负责采集彩色图像。左右为 3D 结构光深度感应器的组件，用于采集深度数据。图 b 展示了深度感应器，其中包括 IR 发射器 IR 摄像头。Kinect 通过 IR 发射器投影随机的点阵，用普通的 CMOS 传感器来捕捉该点阵。当场景的深度发生变化时，摄像头捕捉的点阵也会发生变化，从而推断出深度信息。利用 Kinect 捕捉到的深度信息来提取手势区域，可以减少背景颜色的影响。同时手势的变动会引起手势在深度上的变化，利用这种变化能够更快的进行手势的定位和跟踪。但是同时产生了巨大的数据量，这对实时的手势识别来说是个巨大的挑战。

2.3 颜色空间

颜色空间本质是在不同标准下对彩色的说明。在不同需求下产生了许多不同的颜色空间，常用的颜色空间有 RGB、YUV、HSV 和 HSI 等。

（1）RGB 颜色空间

RGB 颜色空间是最常见的颜色空间，其标准制定是根据人体的视觉原理，将三原色光叠加产生不同的颜色感官。该颜色空间拥有三个颜色通道，每个颜色通道分配 0~255 共 256 个级别的灰度值，通过红（R）、绿（G）、蓝（B）三个颜色通道的变化和叠加可以得到各式各样的颜色，RGB 颜色空间可以用一个三维立方体来描述，如图 2.5 所示。因上述原因，使得 RGB 颜色空间成为了最为流行的颜色空间之一。虽然 RGB 颜色空间符合人类视觉原理，但是由于 RGB 表示法中，R、G、B 各分量中均包含了亮度信息，光照的变化会导致 RGB 值变化。在现实条件下存在很多外部因素，如光照。光照条件的变化常常是手势识别问题中肤色建模的主要干扰因素，因此在 RGB 颜色空间下对手势进行检测和分割的效果十分地不稳定。然而在现实情况中，大多数视频图像采集设备最终采集的是 RGB 值，颜色显示设备也是显示的 RGB 值。考虑到这一情况，解决光照变化敏感的最佳方法就是将 RGB 颜色空间的图像转换到亮度和色度分离的颜色空间。

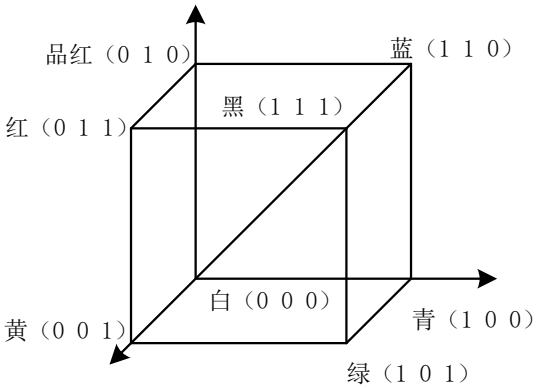
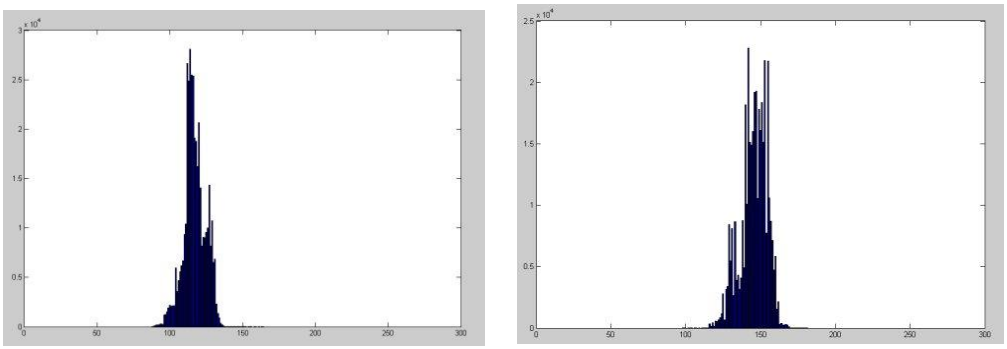


图 2.5 RGB 颜色空间的三维描述

（2）YCbCr 颜色空间和 HSV 颜色空间

常用于肤色模型的颜色空间有 YCbCr 和 HSV 颜色空间，将 RGB 与 YCbCr、HSV 进行比较可以看出，YCbCr 和 HSV 颜色空间在进行肤色分割时由于其肤色范围紧密，因而不易受到光照影响。

YCbCr 也是一种常见的颜色模型，Y 是亮度信息即亮度分量，Cb 和 Cr 皆表示色度信息，Cb 是蓝色色度分量，Cr 是红色色度分量。



(a)Cr 空间肤色分布　　(b)Cb 空间肤色分布

图 2.6 肤色在 YCbCr 颜色空间的分布

RGB 可与 YCbCr 进行转换，具体公式为[27]：

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \frac{1}{256} \times \begin{bmatrix} 65.738 & 129.057 & 25.064 \\ -37.945 & -74.494 & 112.439 \\ 112.439 & -94.154 & -18.285 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1}$$

即：

$$\begin{aligned} \mathrm{Y} &= 0.257 \times \mathrm{R} + 0.564 \times \mathrm{G} + 0.098 \times \mathrm{B} + 16 \\ \mathrm{Cb} &= -0.148 \times \mathrm{R} - 0.291 \times \mathrm{G} + 0.439 \times \mathrm{B} + 128 \\ \mathrm{Cr} &= 0.439 \times \mathrm{R} - 0.368 \times \mathrm{G} - 0.071 \times \mathrm{B} + 128 \end{aligned} \tag{2.2}$$

由上式可以看到，将 RGB 转化为 YCbCr 是线性转换，实现较为容易。

Garcia[28]在 YCbCr 颜色空间进行肤色建模，认为 YCbCr 颜色空间只用考虑色度信息，肤色分布与 Y 分量无关，仅与 Cb 分量和 Cr 分量有关，因而基于肤色在 Cb 和 Cr 的子空间的分布建立肤色模型。这种观点使得大量基于 CbCr 子空间建模的算法难以检测出肤色区域高光和阴影部分。后续的研究发现在 YCbCr 颜色空间中，在高亮度和低亮度区域中肤色色度（Cb、Cr 分量）与亮度 Y 存在非线性相关，而仅在高低亮度之间的区域，肤色色度与亮度线性无关。因此在 YCbCr 颜色空间建模要考虑亮度的影响，需要 YCbCr 三维空间建模。但是肤色在三维空间的形状和分布难以确定，雷明[29]提出可以将三维空间的肤色模型转化到两个二维空间上去建立，即在 Y_Cb 和 Y_Cr 子空间联合建立完整的肤色模型。

YCbCr 颜色空间有如下优点[29]：(1)在 YCbCr 颜色空间中，Y 代表了的亮度信息，而 Cb，Cr，分量不受亮度影响，可以有效地将 Y 分离。（2）Y，Cb，Cr，可以有 R，G，B 经过线性变换得到，具有较高的计算效率（3）YCbCr 颜色空

间中肤色聚类特性比较好。

HSV 即六角锥体模型（Hexcone Model），见图 2.7。该模型中颜色的参数分别是：H（色调）、S（饱和度）、V（明度）。HSV 空间是 RGB 空间的一种非线性变换，变换公式如下[30]：

$$\begin{cases} H_1 = \cos^{-1}\dfrac{0.5[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}} \\ H = \begin{cases} H_1, & (B \le G) \\ 360° - H_1, & (B > G) \end{cases} \\ S = \dfrac{\max(R,G,B)-\min(R,G,B)}{\max(R,G,B)} \\ V = \dfrac{\max(R,G,B)}{255} \end{cases} \tag{2.3}$$

它将相关性很强的 R、G、B 值转换为相关性较弱的 H、S、V 值，同时也能很好的解决 RGB 颜色空间亮度与色度未分离的问题。其中色调（H）为检测肤色的主要依据，在 HSV 颜色空间中，彩色图像的每一个均匀色彩区域都对应一个相对一直的色调（Hue），又因为 H 较不容易收到光线强弱的影响，使得可以单独采用色调来对彩色区域进行分割。

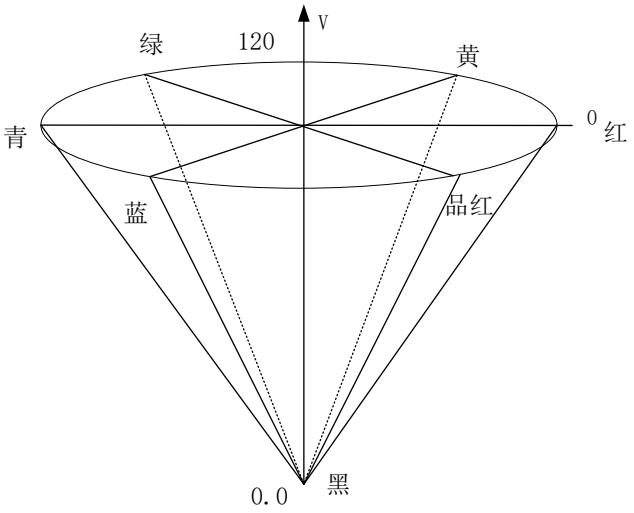


图 2.7 HSV 颜色空间模型

色调 H 可用角度度量，取值范围为 0°~360°。如图所示，红色为 0°，按逆时针计算，绿为 120°，蓝为 240°。另外补色：黄、青和品红，分别为 60°、180°和 300°。饱和度 S 表示颜色接近光谱色的程度，其实就是颜色的纯度，掺入白色光的程度，饱和度越高颜色越深而艳。明度 V 表示颜色明亮的程度，这与发光体直接相关，发光体亮度越大明度则越高。

HSV 空间的优点[30]：（1）符合人眼对颜色的感觉。RGB 颜色模型中任一属性发生改变时，R、G、B 三个坐标都要发生改变；而采用 HSV 颜色模型时只需改变与属性对应的坐标，这是因为在 HSV 颜色空间，三个坐标是相互独立的。（2）HSV 颜色空间是一种均匀的颜色空间，在线性标尺的作用下，彩色分量之

间的距离与坐标上点的欧几里德距离成正比。

HSV空间的缺点[30]：（1）将RGB颜色空间转换到HSV颜色空间需要将R、G、B 三个分量进行非线性变换，这种变换计算的复杂度不低。（2）样本肤色点在空间的聚类性不够明显。

2.4 目标检测与跟踪基础

2.4.1 目标检测方法

运动检测受到很多因素的影响，其中包括光照的渐变和突变、背景物体的移动、背景的变动、阴影、伪装、遮挡等。目前运动检测方法主要有帧间差分法(temporal difference)和背景差分法(background subtraction)两种。研究者可以根据问题的不同，使用目标特有的信息，选择合适的目标检测方法。

帧间差分法的原理很简单，场景中物体运动会使相邻帧出现明显的差别，截取视频中相邻两帧或多帧再进行相减来得到差分图，得到两帧图像的亮度差，再与阀值的对比来分析视频的运动特性，来确定是否有运动物体。将相邻帧之间像素在时间变化上的差分阀值用来提取图像中的运动区域，这种方法不易受到光线变化的影响，背景更新快、自适应能力好。但如果检测的图像中，运动目标尺寸较大、内部颜色较为一致或运动速度变化较快时，检测效果不是很好。

背景差法的原理是假定背景不发生变化，将要检测的帧和假定背景做差。所谓的背景不发生变化主要是指环境不发生变化，如光照条件不变。如果光照条件稍微有点变化，最终检测的结果会得到比较大的噪声，还需要对结果进行图像去噪。这种简单的背景差法只能在理想的环境下实现，在实际应用中很难有这样的条件，因而效果较差。许多学者针对上述缺陷做出了许多改进，背景差法的改进关键点在于背景的更新，要求一种恰当的背景动态更新策略来改善检测效果[31]。常见的方法有累加背景差法、单高斯背景模型、混合高斯模型等。

上述两种运动检测的方法适用于具有运动信息的目标，然而并不是所有的目标都具有运动信息，也就是说运动信息并不是目标的固有属性。当目标停止运动时，上述两种方法会将目标检测为背景。考虑到这一缺陷，需要一种新的方法来对静止的目标进行检测。考虑到本文研究对象是人手，肤色是人手的固有属性，且肤色检测的研究已经开展多年，相关理论已经比较成熟，且肤色也是手势分割的一个重要依据，因此可以通过肤色对目标进行检测。在现实环境中，由于光照变化 、背景复杂及其它不可预知的干扰。仅一个特征往往不能有效地定位出手，需要多种特征互补。多信息结合的方法，如肤色信息和运动信息相结合的方法能够提高目标检测的准确性。

2.4.2 目标跟踪方法

在手势检测过程中，手势在实际情况下是连续的动作，如果每次都从视频流中提取的整张图片进行搜索来定位手势区域，这会产生极大的计算负担，降低检测效率。摄像头捕捉到的都是连续手势视频，因此相邻视频帧存在相关性，我们可以利用这种相关性对目标进行跟踪来快速定位手势以确定手势区域，减少系统的计算负担。考虑到在本文的手势检测及后续的手势分割中，肤色信息都是一个重要的信息，因此在跟踪方法上可以考虑利用颜色特征作为跟踪特征的CamShift 跟踪方法。

CamShift 即均值平移算法,是 Meanshift 跟踪的改进算法。Meanshift 即均值偏移算法，由 Fukunaga[32]在 1973 年提出。最早是用于估计概率密度梯度函数，性能较好，但是由于没有相应的应用，而没有得到太多的关注。1995 年，Yizong, Cheng[33]率先将 Meanshift 引入计算机视觉领域，并对 Meanshift 进行了扩展，引起了人们广泛关注。1997 年 D. Comaniciu [34]等首次应用 Meanshift 算法将颜色特征用于图像分割及目标跟踪。 MeanShift 算法用于视频目标跟踪时，采用目标的颜色直方图作为搜索特征，不断迭代 MeanShift 向量，算法最终将收敛于目标的真实位置，这样就实现了对目标的跟踪。传统的 MeanShift 算法用于解决跟踪问题时有几个优势[33]：

（1）由于算法的计算量不大，在已检测到目标区域时可以做到实时跟踪。

（2）采用核函数直方图模型，在面对诸如边缘遮挡、目标旋转、变形和背景运动等问题有很强的适应能力。

如果背景简单，MeanShift 算法基本能准确地跟踪目标。但它没有模型更新机制，在跟踪过程中核函数窗款是保持不变的，这种情况下算法的适应性就比较差了。面对待测目标姿态改变导致目标大小同时改变时，算法的跟踪窗口无法适应这种变化，导致出现定位不准，甚至目标失踪。针对上述缺点 Bradski[35]提出 CamShift 算法，他利用均值漂移发现颜色概率分布图的质心位置，并根据质心搜寻具有类似颜色的物体，从而不断调整窗口大小来适应待测目标形状的变化。该算法针对 Meanshift 算法搜索窗口无法适应目标大小变化进行了改进，搜索窗口将在每一次搜索完成后自适应调整大小，解决了这一问题。

CamShift 跟踪算法基于颜色特征，因此各种颜色空间受光照亮度影响的不同会对 CamShift 的跟踪效果产生影响。在一个计算机视觉系统中，光照条件是影响性能的一个重要因素。区分肤色区域和非肤色区域的传统方法是采用颜色阀值，但是在变化的光照条件下颜色阀值并不能完全描述肤色的统计特征。为了使 CamShift 跟踪的效果更好，我们应该避免 RGB 这种对光照亮度变化比较敏感的颜色空间，而采用 YCbCr 等对光照强度不敏感的颜色空间。通过对多个颜色空间的聚类分析，发现采用 YCbCr 颜色空间会得到不错的区分效果，因此本文采

用 YCbCr 颜色空间。

2.5 深度学习理论

深度学习是机器学习领域的一个新研究方向，近年来在计算机视觉和语音识别等多个领域取得了突破性的进展。深度学习是建立在多层神经网络之上，运用各种机器学习的方法来解决图像、文本等各种问题的算法集合。其核心是特征学习，通过分层网络获取分层次的特征信息，来解决人工设计特征的难题。

深度学习的发展是建立在人类大脑认知原理的研究之上，特别是视觉原理，图 2.8 显示了人类视觉原理。研究发现人的视觉系统对信息处理是分层的，人类视觉原理如下：接收原始输入信号（瞳孔摄入像素），初步处理（大脑皮层神经元细胞发现边缘和方向），抽象（大脑判定眼前的事物形状）,然后进一步抽象（判断事物是什么东西）。人类的视觉系统对信息采取分级处理，低级的 V1 区提取边缘特征，V2 区抽象为形状或目标部分，一直向高层进行处理，最后得到整个目标或目标的行为。可见低层特征组成了高层特征，而视觉处理的过程是不断抽象的过程。

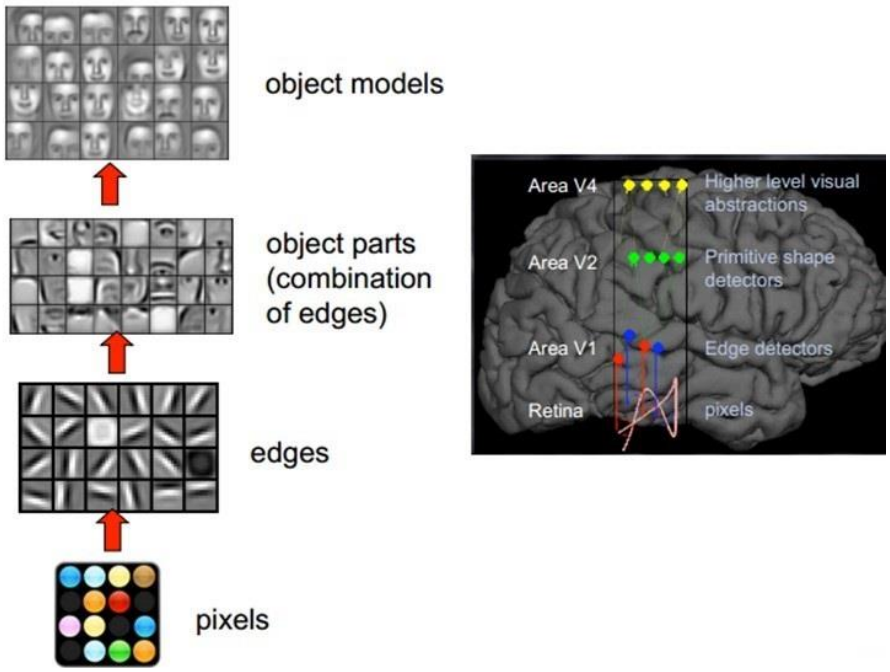


图 2.8 人类视觉原理

研究者从人脑的这个特性得到灵感，模仿人脑特点构造出多层的神经网络。神经网络的低层识别初级图像特征，若干低层特征组成往上一层的特征，最终在顶层就能得到分类。根据这一思想机器学习领域开始蓬勃发展，到目前为止共发生两次热潮。第一次始于上世纪 80 年代，人工神经网络 ANN 的反向传播算法（Back Propagation）的提出，BP 算法使得人工神经网络模型可以从大量的样本中学习规律来对未知的事情做出预测。此时的人工神经网络只含有一层隐层节

点的浅层模型。上世纪 90 年代，其他机器学习模型如支持向量机（SVM）、最大熵方法等相继被提出，因为它们理论不断完善且在各自的应用领域取得了巨大成功，而取代了人工神经网络在机器学习领域的地位，成为了研究的热点。但是无论人工神经网络，还是上述其他机器学习模型，在其结构上只存在一层隐层节点或者不存在隐层节点，因此他们被统称为浅层学习。第二次热潮始于 2006 年 Geoffery E. Hinton[36]提出：1）多隐层的人工神经网络具有优异的特征学习能力，学习到的特征对数据有更本质的刻画，从而有利于可视化或分类；2）深度神经网络在训练上的难度，可以通过“逐层初始化”来有效克服。此后几年内，深度学习迅速成为了机器学习领域的一个热门。

深度学习对深度神经网络的训练包含无监督训练和监督训练方法，在这两种学习框架下可建立不同的学习模型，其中比较有代表性的为深度置信网络（DBN）和卷积神经网络（CNN）。

2.5.1 深度置信网络

深度置信网络（DBN）作为无监督学习框架由 Geoffrey Hinton[37]在 2006 年提出，在此之前尽管大家都知道：在一定程度内，中间隐藏层越多，网络解决问题的能力越强。但是，经典的 BP 算法在面对多层的网络结构时性能急剧下降，因而没有一种很好的方法解决训练上的难度。DBN 模型提出通过逐层初始化来克服深度网络在训练的难度，从而初步解决了这一长期困扰人们的问题。

DBN 由多层神经元组成，这些神经元分为两类即：显性神经元（显元）和隐性神经元（隐元）。一层显元和一层隐元组成了 DBN 的组件：受限玻尔兹曼机（RBM），见图 2.9。

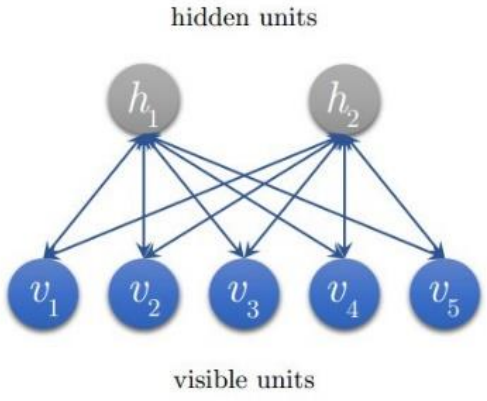


图 2.9 RBM 网络结构

受限玻尔兹曼机 RBM 是一种神经感知器，包含一个隐层和一个显层，显层与隐层的神经元之间为双向全连接。在 RBM 中，任意两个相连的神经元之间有一个权值 w 表示其连接强度，每个神经元自身有一个偏置系数 b（对显层神经元）和 c（对隐层神经元）来表示其自身权重，如下式，表示一个 RBM 的能量：

$$\mathrm{E}(v,h) = -\sum_{i=1}^{N_v} b_i v_i - \sum_{j=1}^{N_h} c_j h_j - \sum_{i,j=1}^{N_v,N_h} W_{ij} v_i h_j \tag{2.4}$$

在一个 RBM 中，隐层神经元h_j被激活的概率：

$$\mathrm{P}\left(h_j \middle| \mathrm{v}\right) = \sigma\left(b_j + \Sigma_i W_{i,j} x_i\right) \tag{2.5}$$

由于是双向连接，显层神经元同样能被隐层神经元激活：

$$\mathrm{P}(v_i|\mathrm{h}) = \sigma\left(c_i + \Sigma_j W_{i,j} h_j\right) \tag{2.6}$$

其中，σ为 Sigmoid 函数，也可以设置为其他函数。

以上是受限玻尔兹曼机的基本构造。

将若干个 RBM 连接起来则构成了一个 DBN，上一层的 RBM 的隐层即为下一层 RBM 的显层，上一个 RBM 的输出即为下一个 RBM 的输入。训练过程中，需要充分训练上一层的 RBM 后才能训练当前层的 RBM，直至最后一层。

2.5.2 卷积神经网络

卷积神经网络是一种多层神经网络，是一种深度的监督学习下的机器学习模型。卷积神经网络是人工神经网络的继续发展，传统的人工神经网络有一些难以克服的缺点：1）比较容易出现过拟合，参数难调整，需要一些技巧性的操作。2）训练速度慢，在层数比较少的时候与其他方法相比其效果并没有优势。

卷积神经网络作为一种深度学习的模型，除了拥有神经网络的特点，也具备深度学习的“深度。”卷积神经网络的权值共享结构与生物神经网络很相似，与人工神经网络相比降低了网络的复杂度，减少权值的数量。传统的识别算法一般都有复杂的特征提取和数据重建，卷积神经网络可以将图像直接作为输入，避免了这一过程。其卷积网络是一个专门设计的多层感知器，是一种特殊的网络结构，它可以对平移、按比例缩放等多种变形方式保持高度不变形。

多层感知器存在的最大的问题就是，它是一个全连接的网络，因此在输入比较大的时候，权值会特别多。比如一个有 1000 个节点的隐层，连接到一个 1000×1000 的图像上，那么就需要 10^9个权值参数（外加 1000 个偏置参数）。这个问题，一方面限制了每层能够容纳的最大神经元数目，另一方面也限制了多层感知器的层数即深度。多层感知器的另一个问题是梯度发散，即在深度增加的情况下，从后传播到前边的残差会越来越小，甚至对更新权值起不到帮助，从而失去训练效果。卷积网络面对多层感知器的上述缺点提出三个方法:局部感受野、权值共享和池化[38]。

局部感受野，其原理模仿人的眼睛。在人看东西时，目光总是聚焦在一个相对很小的局部。在普通的多层感知器中，隐层节点会全连接到一个图像的每个像

素点上，而在卷积神经网络中，每个隐层节点只连接到图像某个足够小局部的像素点上，从而大大减少需要训练的权值参数。

权值共享。如同人的某个神经中枢中的神经细胞，它们的结构、功能是相同的，甚至是可以互相替代的。在卷积神经网中，同一个卷积核内，所有的神经元的权值是相同的，从而大大减少需要训练的参数。

池化，即降采样。在卷积神经网络中，没有必要一定就要对原图像做处理，而是可以使用某种“压缩”方法，这就是池化，也就是每次将原图像卷积后，都通过一个降采样的过程，来减小图像的规模。

一个典型的卷积网络包括卷积层、池化层、全连接层。卷积层完成的操作即使运用局部感受野的原理，卷积层和池化层配合组成卷积组，逐层提取特征，最终通过若干个全连接层完成分类。

2.5.3 卷积神经网络研究现状

卷积神经网络在结构上与 BP 网络相似，且对网络的训练同样采用 BP 算法，因此具有与 BP 网络类似的缺点，这使得可能出现过拟合的情况，训练可能收敛到局部最优值。近年来许多学者针对上述提出不少改进方案，这些方案主要集中在网络结构优化和训练算法改进两方面。

在网络结构扩展方面，文献[39]提出一种多通道输入的改进方案。对输入通道进行了改进，采用多通道输入的方案。它提出在训练卷积网络之前，先对将要输入的图片进行多尺度超像素分割，然后将分割后的超像素序列、恢复超像素所需的上下文信息的空间结构矩阵和范围矩阵分别输入三个输入通道，通过这三个通道将这些信息输入用于卷积神经网络的训练。根据实验结果，这种对于输入通道改进的方案明显地提升了目标检测、显著性检测的效果。

根据卷积神经网络的网络结构，当图像输入卷积网络经过层层映射后通过全连接层输出特征提取的结果。文献[40]提出一种融合多层特征的方法，该方法不仅仅采样全连接层的映射输出，而是将图像在深度网络每一层的映射进行降维处理再融合，最后将融合结果作为特征提取结果输出。

卷积神经网络的每一层的卷积层的工作需要一个卷积核，对于卷积核的设置一般是随机初始化，然后通过 BP 算法对误差函数进行反向传播，再采用随机梯度下降法对卷积核进行调整直到网络收敛，这种方法往往会出现收敛到局部最优值而非所需要的全局最优值。文献[41]采用卷积神经网络进行人脸识别研究时，设置卷积核为加权 PCA 矩阵来实现隐层神经元间的映射，并将卷积网络结构设计为双层，对每层的映射结果都加以利用，最后采用生成码本的方式生成最终特征向量。该方法在 FERET、CAS-PEALR1 和 LFW 数据库上均取得良好的实验效果，对光照、表情变化表现出优异的鲁棒性。

在训练算法的改进方面，主要是引入新的非线性激活函数和对训练的无监督化进行探索。激活函数被用来调整卷积层的输出。卷积神经网络引入非线性激活函数，使得网络加入了非线性因数，避免了线性模型表达能力不够的问题。经典卷积神经网络中常使用 Sigmoid 函数或双曲正切函数（tanh）作为激活函数，但是它们只是调整了输出范围。随着近年来稀疏表示的兴起，许多研究发现人眼系统更倾向于图像的稀疏性描述，因此试图采用其他形式的激活函数。文献[42]提出使用纠正线性单元（Rectified Linear Units，ReLU）作为激活函数能够获得较好的稀疏性输出。同时有不少学者在尝试卷积神经网络的无监督化训练，无监督化训练可以避免需要大量有标签的训练数据的问题。文献[43]提出采用无监督的卷积神经网络来进行车辆类型分类。

以上许多研究者的实验表明，尽管 CNN 的类型不同，但其在单帧图像上和多帧图像上的识别率差别不大。这是否说明传统的深度卷积神经网络已不能适应学习运动特征的需求？可能需要针对这一问题专门设计一种基于 CNN 网络结构来显式地对时空域上行为信息建模[44]。

2.6 智能机器人

2.6.1 发展现状

机器人在当前的生产、生活中的应用越来越广泛，并逐渐开始替代人类在各种场合的作用。机器人技术是综合了计算机、电子、机械、信息、控制理论和仿生学等多领域多学科而形成的高新技术，集成了各个学科顶端的研究成果，代表着高新技术的发展方向，也是目前科技研究的热门方向。机器人的发展到目前为止，大概经历了三代：

第一代为可编程的示教再现型机器人，如 1959 年德沃尔和约瑟夫发明的世界上第一台工业机器人，图 2.10(a)，其特点是机器人可以在人们预先编入程序指令的情况下，按照该程序指令重复工作。这类机器人中比较典型的是工业机器人，如各种机械手臂，它们离不开人类的实时控制，灵活程度很低。

第二代为感知机器人，如 1965 年约翰霍普金斯大学物理实验室研制出的 Beast 机器人，见图 2.10(b)，其具有一定感知功能和自适应能力。这种机器人一般采取离线编程，可以脱离人类的实时控制。面对具体的任务作业可以根据不同的作业对象改变作业的内容，其特点是具备了一定的“感知”能力和初级判断能力。第二代机器人发展到现在，已经进入智能机器人阶段。智能机器人搭载了大量的传感器，能够将各种传感器获取到的信息进行融合，再进行判断并做出应对，有极强的环境适应能力。其特点是环境适应力强，学习能力强。机器人的发展到

此时，人工智能的概念已经比较成熟，但是其“智能”还是对人类对自身“智能”的模拟，其目的还是为了实现人类自身能够完成具体工作。随着神经网络模型和遗传算法的兴起，通过机器学习方法训练出的人工智能基本能够满足人类对“智能”的理解，但是这种智能依旧是“复制”而非“创造”。此时的智能机器人依旧不能脱离人类，依旧是人类的工具。

第三代为智能机器人，此时的机器人已经脱离了机器实体的限制，其关键点在于人工智能，如 2014 年通过了图灵测试的聊天程序“尤金古斯特曼”，见图 2.10(c)。第三代智能机器人具备的特点应该有：能够像人类一样的思考和活动，能够进行自我学习和成长。

(a) (b) (c)

图 2.10 三代机器人

2.6.2 智能机器人的应用

目前智能机器人在各行业有广泛的应用，其主要领域有工业、农业、医疗和家政等。

在工业发达的国家，工业机器人及自动化生产线成套设备已经成为高端装备的重要组成部分及未来发展趋势[45]。随着全球进入工业 4.0 时代，作为智能制造装备产业的重要组成部分，机器人产业将迎来巨大发展，工厂“机器换人”的现象将越来越频繁。在工业应用方面，智能机器人的显著特征是智能感知和智能规划。智能感知可以使机器人实时获取环境信息，智能规划使机器人对所处环境进行最优决策并做出相应操作。在传统的工业装配作业中，往往会因为工人操作失误，造成人员和财产的损失，而具备两大特征的智能工业机器人可以通过各种传感器和力控技术，避免产品损伤，也可以利用视觉传感器和伺服控制技术实现精确定位。

在医疗机器人应用方面，医疗机器人系统已经实现了商用化。目前在临床应用方面，机器人技术还只是医生医学技术的延伸，只能起到辅助作用，以减小人类因为经验或精力不足所带来的失误。但是在医疗服务机器人方面，机器人的“智能”得到很大的应用。在医院出现了挂号机器人，能通过与患者的交流引导

患者前往正确的科室，或者推荐合适的医生[46]。还有一类护理机器人，可以取代人类的护理人员，能够与患者交流，照顾患者的日常起居。另外在非物理形态的机器人上面，人工智能软件系统得到广泛应用。例如医学影像方面的应用，基于卷积神经网络的深度学习方法，可以更快更精准的处理医疗数据，通过对影像的处理能够直接展示出病灶，从而更好地诊断和治疗[46]。

在家用机器人方面，无论是家政服务还是作为家庭宠物，智能机器人得到广泛应用。家庭机器人对智能化的要求相对更高，主要有以下原因[47]：

（1）工作在复杂的半结构化环境。

（2）服务对象未经过专业训练。

（3）通常要同时与多个对象进行交互。

基于以上几点原因，家用服务机器人需要解决以下几个关键技术：室内自主导航技术，人机交互技术，物体识别技术。室内导航技术用于解决家庭环境复杂的因素，不同的用户家庭空间情况不同，需要一套成熟合适的技术来使家用机器人具有普适性。不同的交互场景需要不同的交互方式，人机交互技术需要解决传统的交互方式带来的不便，对于家用机器人来说需要引入一些新的交互方式如基于视觉、基于听觉的交互方式，这样对于没有经过训练的服务对象可以实现自然、高效的交互。物体识别技术主要是针对家庭环境的复杂性，传统的模式识别方法针对可能出现的物体设计特征，再进行特征提取的方案已经不能满足数以万计家庭的需求。家用机器人的物体识别需要提出一种新的方案，目前基于深度学习的自主特征提取算法是一个很好的研究思路。

2.6.3 类人机器人的优势

类人机器人并不是机器人相关学术领域的研究重点，科研机构和学者们对机器人是否是“人形”并不太关注，世界各地的尖端机器人研究团队的研究都是基于非人形的机器人。目前有大量最前沿的研究使用实验载体都是非类人机器人，如图 2.11。

图 2.11 非人形机器人

类人机器人对机器人技术的要求非常高，甚至远远高于非人形机器人。理论上机器人的作用是为人类服务，我们需要机器人在各种场景下代替人类，因此机

器人是否是人形应该不是很重要。那么为何要付出这么大的代价把机器人弄成“人形”？原因很简单，那就是客户喜欢。

1969 年日本机器人专家森昌弘提出“恐惑谷”效应，认为人类对类人物体的好感度是随着它的拟人程度变化的（见图 2.12），类人机器人在一定程度上更能被人类接受。

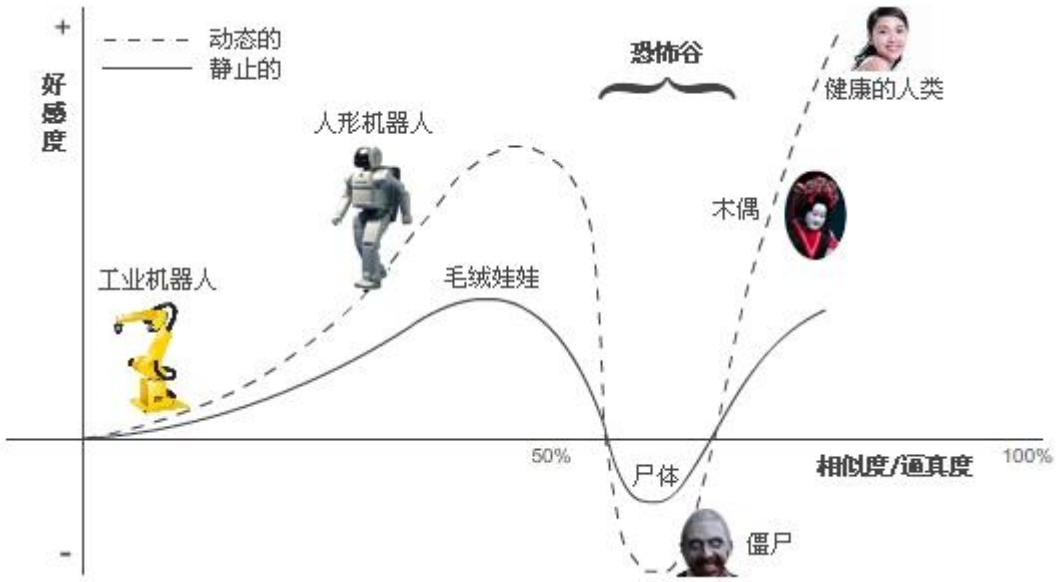


图 2.12 恐惑谷曲线

目前，类人机器人如 NAO 机器人已经被使用于儿童自闭症的治疗方法的探索中[48]，由此可见在某些领域类人机器人所具有的“对人类的好感影响力”有着非人形机器人不可替代的作用。

2.7 本章小节

本章以手势识别的流程为线索，首先对手部跟踪与定位的算法进行介绍，然后简要介绍手势分割的基础概念，接着介绍了国内外对于深度学习和卷积神经网络在图像处理领域的研究和应用，最后简要的介绍了智能机器人的发展情况和类人机器人与其他机器人相比在应用领域的优势，为后续章节相关知识和技术的引入做铺垫。

第3章 融合肤色信息与运动信息的手势分割研究

手势分割的目的是为了使手势与背景分离出来，为后继的特征提取做准备，是手势识别的重要步骤之一，分割的质量影响后继的特征提取及手势识别。本文采用卷积神经网络对手势进行识别，根据卷积神经网络的特性，可以直接输入图像，不用人工设计特征和进行特征提取。但是真实的交互场景中，我们对于手势在什么时候、什么位置出现都是未知的。在这种情况下，如果将视频流中的图像直接进行全幅扫描，将会产生巨大的工作量，浪费大量时间。因此在图像输入CNN之前，先进行手势分割，将会大幅提高识别效率。

对于手势分割，好的分割结果应该是尽可能多地把手势完整分割下来，同时又避免把手势与背景分割在一起。为满足手势识别系统实时性要求，手势分割要平衡分割效率和分割质量。

3.1 手势分割的技术难点

手势分割有两大难题，一是在复杂的背景环境中，多变的光照条件会使手势产生不同的高光和阴影，以及使背景物体颜色与肤色相近；二是，在三维空间手势活动很灵活，使得手势外形特征复杂多变。这些难题给进行准确的手势分割带来极大的困难。

进行手势分割实际上包括两个步骤：首先是手势定位，这是指从图像中确定手是否出现，并且确定手势所在区域。该步骤即是前文所述的手势检测和跟踪需要完成的工作。完成第一个步骤后，需要进行排除背景干扰，将手势区域从图像中分割出来。

3.2 常见的手势分割方法

基于视觉的手势分割方法主要有基于轮廓的手势分割方法、基于运动信息的手势分割方法、基于颜色信息的手势分割方法等。

基于轮廓的手势分割方法利用手的拓扑结构特征来对手势进行分割，但人的手势灵活多变，手部旋转弯曲会使手部轮廓发生改变[49]。并且手势形状存在深度凹陷区域，受这些区域影响，传统基于轮廓的方法无法收敛。这两个技术难题极大地影响了轮廓的准确性，使得手势分割效果并不理想。

常见的基于运动信息的分割方法主要是帧间差分法和背景差分法。帧间差分法提取出视频中的一段连续帧并进行差分运算，来消除手势运动产生对背景的影响，使得可以提取出精确的运动目标轮廓信息。背景差分法首先需要建立图像的

背景参考模型，再将图像序列中当前帧和背景参考模型进行对比来检测运动物体。背景建模技术的优劣对背景差分法的性能影响很大。运动中的光影变化和背景的动态变化是影响分割结果的两大因素，解决它们的方法正是背景差分法的热门问题。从帧间差分法和背景差分法的缺点分析可见，基于运动信息的分割方法有一个难以克服的缺点：容易受到其他运动物体的干扰。

基于颜色信息的手势分割方法主要是利用人类的肤色信息建立肤色模型，通过对比肤色模型中肤色和背景的差异来实现手势分割。但是在实际应用中复杂的背景环境让该方法很难得到应用。在实际环境下，光源亮度和位置变化、有色光源的色彩偏移等条件都会影响到肤色的影响。且手部反转弯曲形变会让光源角度和阴影发生改变。这些影响和改变都使手部区域肤色不一致，这使建立一个具有高准确度的肤色变得很困难，并且类肤色信息的干扰也是需要解决的问题。

3.3 基于肤色信息的手势分割

考虑到肤色信息在手势识别的多个环节都会被使用，现有的肤色检测和分割理论也比较成熟，利用肤色信息分割手势可避免重复计算，降低系统的计算负担，提高处理速度。本文决定采用基于肤色的手势分割方法，因此需要对肤色检测技术进行研究。肤色检测一般采用统计的方法。通过建立肤色统计模型进行肤色检测，主要包括两个步骤：颜色空间变换和肤色建模[50]。

不同肤色模型的建立基于不同的颜色空间。肤色信息主要由肤色模型来描述，而肤色模型的选取要由颜色空间的选取来决定，因此肤色建模的第一个步骤就是选取颜色空间。我们可以从两个方面考查某个颜色空间：（1）在该颜色空间中“肤色”区域的分布是否能够在给定的模型上体现；（2）颜色空间中的“肤色”与“非肤色”区域的重叠有多少。不管在什么样的颜色空间中，肤色模型大体上分为四种：区域模型、简单峰高斯模型、混合高斯模型和直方图模型。

3.3.1 颜色空间的变换

肤色在颜色空间的分布相当集中，但是光照和人种的不同会对其产生影响。通常设备采集的图像或图像序列是基于 RGB 颜色空间，RGB 颜色空间是亮度和色度没有分离的颜色空间，光照会对肤色造成很大的影响。为了减少肤色受照明强度影响，通常将颜色空间从 RGB 转换到亮度与色度分离的某个颜色空间，常见的有 YCbCr 和 HSV 颜色空间。在双色差或色调饱和度平面上，不同人种的肤色变化不大，肤色的差异更多的是存在于亮度而不是色度[50]。本文采用基于 YCbCr 颜色空间建立肤色模型，利用肤色模型进行肤色分割。

3.3.2 肤色建模

基于 YCbCr 颜色空间的高斯肤色模型被很多研究采用。在 YCbCr 颜色空间建立高斯肤色模型是因为在 YCbCr 颜色空间中，不同的肤色色度分量 Cb 和 Cr 的分布趋于一致，近似呈现二维高斯分布。传统的高斯模型在使用过程中，其参数是基于大量肤色统计得到的固定值[27]，而在肤色和光照变化较大的情况下，采用固定值的高斯肤色模型鲁棒性较差。Grimson[54]和 Stauffer[55]提出基于混合高斯模型的背景差方法，该模型中的参数能够针对复杂背景实现自动更新，能够有效地处理对于光照、物体运动速度以及出现的一些突变因素对模型的影响问题。

在混合高斯模型中，对图像中每个像素点建立 k 个高斯模型，对于 t 时刻像素的样本在值x_t，它的概率密度函数由 k 个多维高斯分布函数的概率密度函数加权和来表示[50]：

$$\mathrm{P}(x_t) = \sum_{i=1}^{K} \omega_{i,t}\eta_{i,t}\left(x_t,\ \mu_{i,t},\ \Sigma_{i,t}\right) \tag{3.1}$$

K 为高斯模型的个数，$\omega_{i,t}$为第 i 个高斯分布的权重；$\mu_{i,t}$为第 i 个高斯分布的均值；$\Sigma_{i,t}$为协方差矩阵。

GMM 是一种聚类算法，其中每个高斯模型都是一个聚类中心。算法思想是：不管数据样本呈什么分布，只要 K 值够大，那么 GMM 模型就会变得足够复杂，就可以用该模型来逼近任意连续的概率密度分布。分类标签 Y 是隐藏变量。其中数据点的分类标签 Y 可能有两种情况，第一种情况是分布标签为已知，此时直接利用极大似然估计就可以计算出来：

设样本容量为 N，属于 K 个分类样本数量分别是$N_1, N_2, \ldots, N_K$，属于第 k 个分类的样本集合是 L(k)。参数的表达式为[50]：

$$\omega_{k,t}=\frac{N_k}{N} \tag{3.2}$$

$$\mu_{k,t} = \frac{1}{N_k}\sum_{x\in L(k)} x \tag{3.3}$$

$$\Sigma_k = \frac{1}{N_k}\sum_{x\in L(k)}(x-\mu_{i,t})(x-\mu_{i,t})^T \tag{3.4}$$

另一种情况是 Y 为未知，设 N 个数据点，服从某种分布 P(x)，我们需要找到一组参数Θ，使得生估计的概率密度函数上的概率值最大，其概率函数表达为：

$$\prod_{i=1}^{N} \eta_{i,t}\left(x_t,\ \mu_{i,t},\ \Sigma_{i,t}\right) \tag{3.5}$$

该函数又被称为极大似然估计函数。

通常单个点的概率都很小，考虑到当样本点的数目 N 足够大时，上式连乘的结果在计算机计算中会造成浮点数下溢，因此采取取对数的方法，因此 GMM 的极大似然函数为：

$$\max \sum_{i=1}^{N} log\left(\sum_{i=1}^{K} \omega_{i,t}\eta_{i,t}\left(x_t,\ \mu_{i,t},\ \Sigma_{i,t}\right)\right) \tag{3.6}$$

一般求极值是通过求导的方式，而上式 log 函数中还包含求和，这使得求导非常复杂。通常采用 EM 算法来求解，EM 算法分为 E 步骤和 M 步骤两部分。EM 要求解的问题一般形式是：

$$\theta^* = \arg\max \prod_{j=1}^{|x|} \sum_{y\epsilon\gamma} \mathrm{P}(X = x_j, Y = y; \theta) \quad (3.7)$$

其中 Y 为隐含变量。

采用 EM 算法，其思路为：随机初始化一组参数$\theta^{(0)}$，根据后验概率 Pr(Y|X; θ)来更新 Y 的期望 E（Y），然后用 E（Y）代替 Y 求出新的模型参数$\theta^{(1)}$，反复按照这种方法迭代，直到θ趋于稳定。EM 算法的具体步骤分为以下两步：

第一步计算期望(Expectation，E-Step)假设模型参数已知的情况下求隐含变量 Z 分别取z_1, z_2,....的概率，表达式为式 3.10。

$$\gamma(\mathrm{i},\mathrm{k}) = \alpha_k \mathrm{Pr}(z_k|x_i; \pi, \mu, \Sigma) \quad (3.8)$$

α_k为权值因子，它表示在训练集中数据点属于类别z_k的频率，在 GMM 模型中z_k即是$\omega_{k,t}$。

$$\gamma(\mathrm{i},\mathrm{k}) = \frac{\omega_{k,t} N(x_i|\mu_{k,})}{\sum_{i=1}^{K} \omega_{i,t} N(x_i|\mu_j, \Sigma_j)} \quad (3.9)$$

第二步为 Maximization（M-Step），即最大似然的方法求出模型参数，此时我们认为$\gamma(\mathrm{i},\mathrm{k})$是数据点$x_i$由第 k 个高斯函数生成的概率。可以由公式 3.5、公式 3.6 和公式 3.7 可以推出[50]：

$$N_k = \sum_{i=1}^{N} \gamma(\mathrm{i},\mathrm{k}) \quad (3.10)$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N} \gamma(\mathrm{i},\mathrm{k})\, x_i \quad (3.11)$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} \gamma(\mathrm{i},\mathrm{k})(x_i - \mu_k)(x_i - \mu_k)^T \quad (3.12)$$

$$\omega_k = \frac{N_k}{N} \quad (3.13)$$

重复 E 步骤和 M 步骤直到算法收敛。

3.4 基于运动信息的手势分割

肤色并不是很独特，有不少日常物品的颜色与肤色范围重叠。文献[6]提到近肤色的衣物和棕色地板会对被肤色模型判断为肤色区域，针对同一图像序列出现脸部和手势，可以通过对深度信息进行处理，提取出精确的手势。考虑到本文是基于单目摄像头的图像采集，采集的图像在大多数情况下包含人体的其他部分，

如脸部和四肢，这些人体部分均具有与手势近似的肤色信息。在单目摄像头的情形下不具备采集深度信息的能力，因此还需要其他方法将手势分割出来。

在复杂背景中仅仅依靠肤色模型很难完美地分割出手势，但是使用运动信息，将手势从该场景中的多个物体从分割出来便成为可能。在类人机器人的实时交互场景中，人与机器人交互时，除了手部外的身体和其他躯干保持静止状态，因此对手势的运动信息进行检测是一个很好的切入点。

3.4.1 帧间差分法

帧间差分法是对相邻两帧图像进行差分，并设定一个阀值，认为差分结果大于阀值的像素点就是运动目标。帧间差分法检测运动目标的具体步骤为：首先对视频序列取其第 K 帧和 K+1 帧图片进行平滑去噪等图像处理，再做帧间差分法处理，用第 K+1 帧图片减去第 K 帧图片，得到二值图像，公式表达为[56]:

$$\mathrm{D}(x,y)=\begin{cases}1, |f_{k+1}(x,y)-f_k(x,y)|\geq T\\ 0, |f_{k+1}(x,y)-f_k(x,y)|<T\end{cases} \tag{3.14}$$

T 为预先设定的阀值。T 的设置很关键，T 值过大检测目标可能会出现空洞甚至漏检；T 值过小，会出现大量噪声。

从上式可以看出，如果待检测目标是匀速运动，帧间差分法检测出的运动目标会比较一致。但是如果变速运动则会检测不准，出现多检和少检。另外帧间差分法检测的物体是前后两帧相对变化的部分，无法检测到重叠的部分，也会导致检测的目标出现空洞[56]。我们可以用三帧差分法来解决这个问题。

3.4.2 基于三帧差分法的运动检测

三帧差分法是对原有两帧间差分法的改进，它将相邻的 k-1，k，k+1 三帧作为一组来进行差分，能够很好的检测出中间帧运动目标的形状轮廓。具体步骤与帧间差分法近似：用处理后的第 k 帧减去第 k-1 帧，得到二值图像$D_1(x,y)$，再用第 k+1 帧减去第 k 帧，得到二值图像$D_2(x,y)$，最后用$D_1(x,y)$和$D_2(x,y)$进行“与”运算得到三帧差分图像$D(x,y)$。其公式表达为[31]：

$$\mathrm{D}_1(x,y)=\begin{cases}1, |f_k(x,y)-f_{k-1}(x,y)|\geq T\\ 0, |f_k(x,y)-f_{k-1}(x,y)|<T\end{cases} \tag{3.15}$$

$$\mathrm{D}_2(x,y)=\begin{cases}1, |f_{k+1}(x,y)-f_k(x,y)|\geq T\\ 0, |f_{k+1}(x,y)-f_k(x,y)|<T\end{cases} \tag{3.16}$$

$$\mathrm{D}(x,y)=f(x)=\begin{cases}1, D_1(i,j)\cap D_2(i,j)=1\\ 0, D_1(i,j)\cap D_2(i,j)=0\end{cases} \tag{3.17}$$

帧间差分法的优点是计算量小且实时性高，考虑到嵌入式设备的性能限制和实时的交互需求，帧间差分法是一种比较适宜的运动检测方法。

3.5 重建手势区域

3.3 节和 3.4 节分别利用肤色信息和运动信息对手势进行分割，但是仅仅使用混合高斯肤色模型和三帧差分法很难得到让人满意的分割效果。例如利用肤色模型进行分割时，受到背景和光照影响，得到的二值化处理后的图像的手势区域边缘会大量存在大小不一的空洞、毛刺和不完整的轮廓，在确定的图像区域内部也会出现一些孤立的块状区域。使用三帧差分法也同样会出现同样的情况如图 3.3(a)所示。我们需要通过形态学方法对二值图像进行处理，再使用标记连通方法处理，最后重建图像区域 3.3(b)。

（a）三帧差分法处理得到的二值图

（b）图像重建后的二值图

图 3.1 图像重建

3.5.1 形态学方法处理

经过二值化处理的图像，手势区域边缘会存在大小不一的空洞以及具有毛刺或不完整的轮廓，产生大量的噪声。这与我们预期的结果不符，因此我们需要采用形态学的方法对图像去噪。对上述问题图像形态学常用的方法有膨胀和腐蚀。

（1）膨胀

膨胀就是求局部最大值的操作。从数学角度来说，膨胀就是将图像 A 与核 B 进行卷积，计算核 B 覆盖区域的像素点的最大值，并把这个最大值赋值给参考点指定的像素。这样会使图像中的高亮区域逐渐增长。

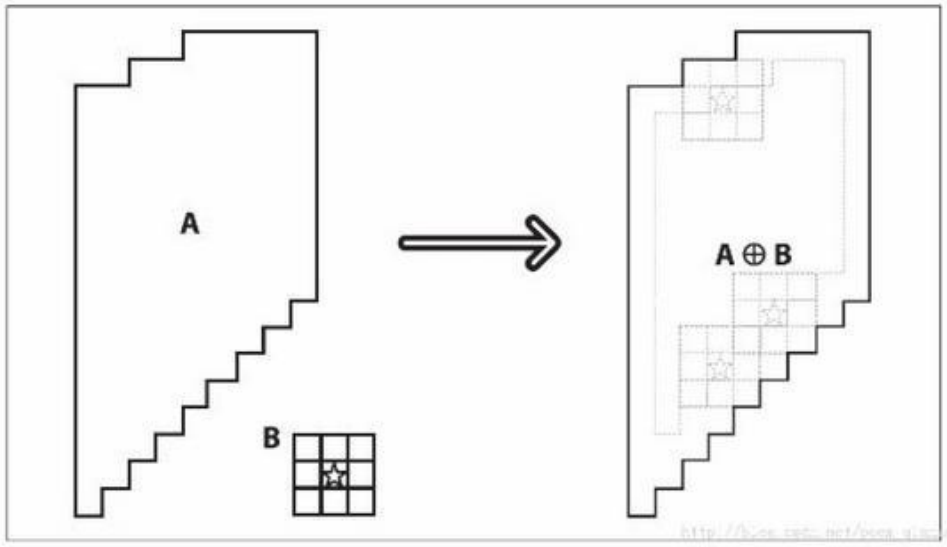


图 3.2 膨胀操作

（2）腐蚀

腐蚀是与膨胀相反的操作，是求局部最小值。将图 A 与核 B 取交集，如果核 B 与图 A 的交集完全属于图 A 的区域内则保存该位置点。所有满足条件的点构成图 A 被核 B 腐蚀的结果。

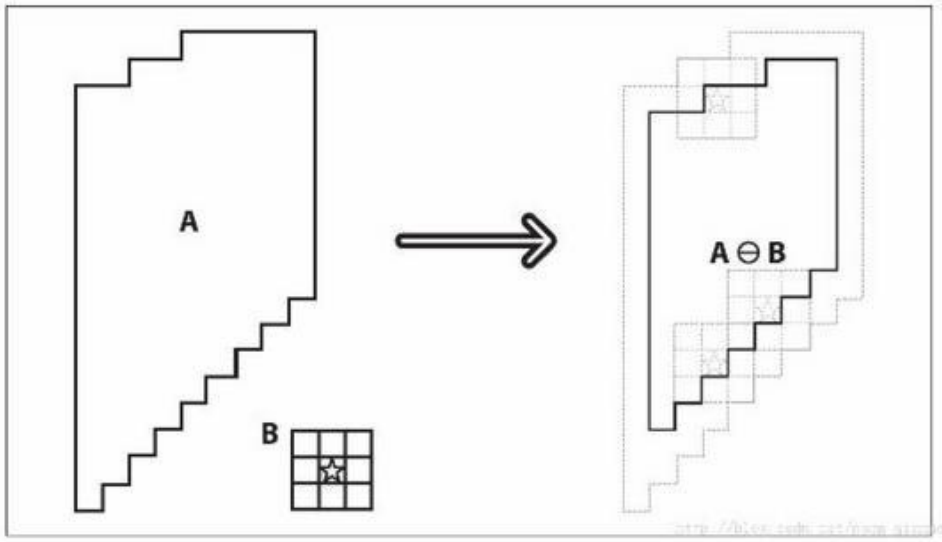


图 3.3 腐蚀操作

在对二值图进行形态学方法处理时，膨胀算法可以扩展二值图像中的亮色区域，腐蚀算法可以扩展二值图像中的暗色区域[57]。本文设置结构为3×3的核对二值图像进行先腐蚀再膨胀的开运算处理，去除二值图像中孤立的噪声点和手势区域边缘不平滑的凸出部分，同时对二值图像的空洞进行填充。

3.5.2 标记连通处理

经过形态学方法处理后的手势图像，除了有效的手势区域外，还存在一些孤

立的块状区域，这些区域的存在会对后续识别结果造成影响。本文采用标记连通法进行处理。具体的操作步骤为：取得形态学处理后的二值图后，将该二值图内的每个块状区域进行标记分别记为 1，2，3，…，N，N 为块状区域总数，然后依次计算每一个标记区域的面积 S，将这 N 个区域面积进行比较选出最大的区域面积S_{max}[58]。因为二值图中手势区域理应为最大面积区域，所以S_{max}即为手势区域面积，将最大的块状区域标记为Max。最后重新遍历二值图像，对每个已经标记的区域判断其是否属于Max，属于则记为 1，不属于则记为 0。最后得到连通后的手势区域。

3.6 融合算法

如前文 3.2 节所述，手势分割的方法有多种，但是每种都有自己缺点。基于肤色的手势分割方法在本文中难以克服脸部和其他躯干肤色的干扰，同时也受光线变化的影响较大。而基于运动信息的方法虽然能够克服上述缺点，但是在实际应用中难以避免其他运动物体的干扰。因此本文采用一种肤色信息和运动信息相结合的方法。

3.6.1 算法设计

融合算法的操作流程如下：首先在 YCbCr 颜色空间下采用混合高斯肤色模型提取视频流中的肤色区域；然后利用三帧差分法提取运动区域，将图像中类肤色区域去除；再将肤色区域和运动区域进行“与操作”来得到一个运动肤色区域；最后采用形态学方法及连通区域分析去除噪声，从而得到比较理想的手势图像。

融合算法流程图如下：

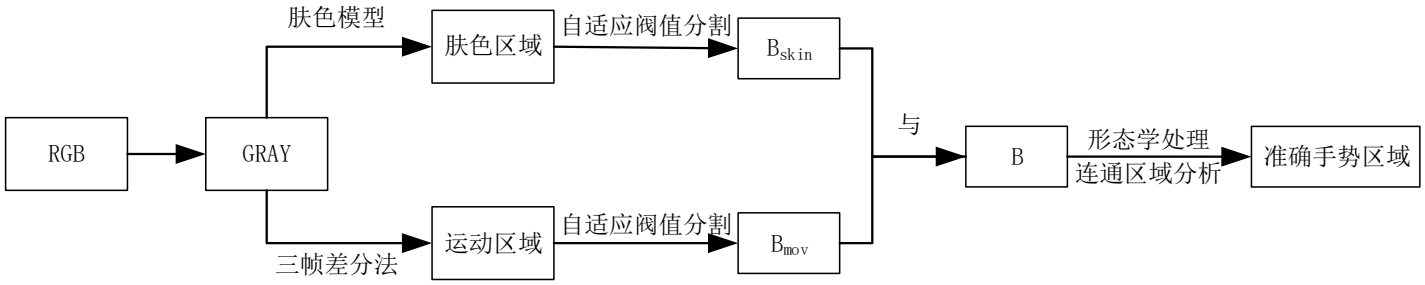


图 3.4 融合算法流程图

其中 RGB 是实时视频中截取的彩色图像序列；GRAY 是由 RGB 图像序列转化得到的灰度图像序列；B_{skin}和B_{mov}分别为经过肤色模型和三帧差分法得到的二值图像序列；B 是经过“与”运算后得到的二值肤色运动图像区域。

3.6.2 算法的不足与改进

（a）原始帧　（b）肤色模型分割　（c）三帧差分法分割　（d）形态学处理　（e）检测结果

图 3.5 算法运行效果图

以图 3.7(a)所示的运动图像为例，首先用混合高斯肤色模型对第 K 帧做肤色判断得到目标色块，如图 3.7 (b)所示。同时采用三帧差分法对连续的第 k-1、k、k+1 帧图片进行处理，得到运动目标的轮廓，计算方法见 3.4.1 节。此时得到的运动目标轮廓不连续且内部出现空洞，如 3.7 (c)所示。再对上一步得到的运动目标轮廓做形态学处理，轮廓经过多次腐蚀膨胀操作填充轮廓，得到较清晰的运动区域轮廓，如图 3.7 (d)所示。最后将肤色模型和三帧差分法分别得到的二值图像进行“与”运算后，再进行后续处理，得到分割的最终结果，如图 3.7 (e)所示。

本文提出的融合算法的目的是为了分割出手势区域，但是实验结果并不理想，最终的分割结果中脸部区域还是包含在内。考虑到融合算法中三帧差分法的目的是把非手势区域的类肤色区域剔除，问题可能是出在三帧差分法的处理上。

在实验中发现将三帧差分法用于手势检测时，可能会出现以下两种问题：

（1）在视频流中可能会存在手势在某一位置短暂停留而丢失[56]。由三帧差分法的算法原理可知，可能会因为检测不到运动目标在后续融合算法中，不能通过运动信息将类肤色区域去除。

（2）当人体处于摄像头视野中时，面部和躯干等部位的微小运动可能会被误检测为目标色块而被分割出来，从而对后续融合算法造成干扰。

经过对实验结果的分析，确定问题（1）（2）都可能导致上述实验结果。针对（1）问题需要对三帧差分法进行一些改进，针对（2）问题则是需要选定一个最佳的阀值 T。本文采用文献[56]的改进方案解决上述问题。

算法改进如下:第一步进行判断，将上一次算法分割得到的目标区域色块数记为N_{k-1}，将当前帧检测到的目标区域色块数记为N_k，将N_k和N_{k-1}的大小进行比较，当$N_{k-1} > N_k$，则说明当前帧至少有一个色块丢失。根据分析，目标区域色块减少可能是手离开摄像头视野范围、手势短暂停留、手被遮挡等多种原因造成，根据该判断结果进行下一步处理。第二步，将当前帧肤色模型处理结果与上

一帧融合算法处理结果做按位或运算来最大可能地保存肤色可能存在的区域，在对或运算后的区域用肤色模型做肤色检测，最后将得到的结果作为当前帧的最终结果。

在帧差法中，阀值 T 表征为运动检测的灵敏度。T 的设置是否合理，关系到人体微小的运动是否被误检，针对问题（2）进行阀值 T 选取实验。本文对阀值 T 的选取进行了大量对比试验，以求选取最合适的 T。实验分别选取 T 值为 1~20 进行实验，根据硬件设备 NAO 机器人的摄像头提供帧率为 30f/s（具体数据见第五章）视频摄取。实验结显示，当阀值 T 为 1-7 时，三帧差分法均分割出人脸，其中当 T 为 5 时分割出最大面积的人脸区域。当阀值 T 为 8-15 时，手势区域和人脸区域出现部分漏检，当 T 大于 15 时，出现分割失败。其中 T 为 10 手势分割效果最好。

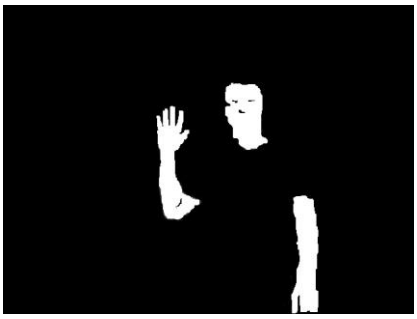
图 3.6 阀值 T 调整后分割示意图

3.7 实验结果与数据分析

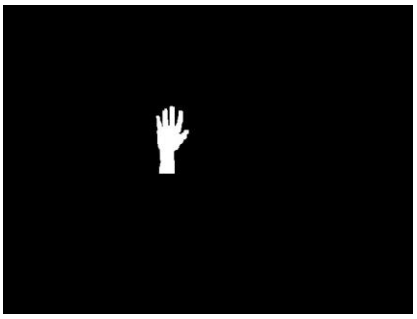
3.7.1 分割效果

为了验证本文提出的融合肤色信息与运动信息手势分割算法，分别对文献[7]采用的规定肤色范围、文献[8]采用的椭圆肤色模型、文献[10]采用的混合高斯肤色模型和本文的方法进行对比实验。

(a)原始帧图像　(b)规定肤色范围模型分割效果　(c)椭圆肤色模型分割效果



(d)混合高斯模型分割效果



(e)本文算法分割效果

图 3.7 传统分割方法与本文方法效果对比

图 3.9(a)是视频流中截取的原始帧，图 3.9(b)是采用文献[7]的方法利用规定肤色范围建模得到的效果图，图 3.9(c)是采用文献[8]的方法利用椭圆肤色模型建模得到的效果图，图 3.9(d)是采用文献[10]的方法利用混合高斯肤色模型建模得到的效果图，图 3.9(e)是本文提出方法分割效果图。由上述分割效果图可知，规定肤色范围和椭圆肤色模型这两种方法只能简单的提取肤色特征，如果背景中存在大面积类肤色区域，分割效果将会不理想，本实验中水壶、门和纸盒等背景均被误分割为目标；基于混合高斯模型建立的肤色模型能够有效的分割出肤色区域，但是在实际应用中，待分割图片中会存在手势以外的人类肢体，仅靠肤色模型难以排除这些干扰。

3.7.2 分割时间

本实验性能测试在 Intel core i5 处理器，CPU 运行频率为 2.30GHZ，内存为 8G，Ubuntu14.04 32 位操作系统的笔记本电脑上完成。上述各种方法分割时间如表 3.1，每种方法分别进行 5 次完整分割，最后求取平均值。

表 3.1 四种分割方法的时间比较

ms

方法类别	1	2	3	4	5	平均值
规定肤色范围	6	7	6	6	7	6.4
椭圆肤色模型	7	7	8	8	9	7.8
混合高斯模型	33	37	35	40	36	36.2
本文算法	45	43	47	45	43	44.6

由表中分割时间可知，单从分割时间看文献[7]和文献[8]的方法最优，本文算法中融合运动信息和肤色信息与文献[10]中只使用肤色信息的方法相比由于增加运动信息的处理，导致分割时间增加。但从分割效果来看本文算法能够准确将手势提取出来，排除类肤色区域的干扰，能够为后续步骤节省大量的计算资源，从而减少计算时间。

3.8 本章小结

手势分割是实现手势识别的关键和前提条件，好的手势分割能够很大程度减小手势识别阶段的难度，和减小后续步骤的工作量，这在嵌入式设备上显得尤为重要。针对目前各种手势分割方法的不足，本文分别研究了两种常用的手势分割方法，评估了它们的优势和不足，并针对不足提出了一种将肤色信息和运动信息相结合的手势分割方法，通过实验对该方法出现的问题进行分析和对算法进行改进。实验结果表明，该分割算法在复杂背景的条件下能够有效的分割出目标区域。

第4章 基于 CNN 神经网络的手势识别研究

在对手势进行分割处理后还有两个关键步骤：特征提取和手势分类，这两个步骤实质上是图像分类技术的应用。虽然目前存在很多优秀的图像分类算法，但是考虑到 CNNs 在图像分类领域已经取得了巨大成功，且学术界和工业界都倾向采用 CNNs 进行图像分类，本文决定采用 CNN 来完成手势识别系统中的手势分类任务。本章将对构建手势识别系统中所需要用到的手势跟踪技术和卷积神经网络的设计进行具体介绍。

4.1 手势跟踪技术

嵌入式设备计算资源有限，通过手势跟踪能够快速定位手势区域，减少后续的计算量，因此采用手势跟踪技术是一个很好的方法。手势分割中利用运动信息的方法的实验中发现，人手进入或离开摄像范围需要一定的时间，而相邻帧有一定相关性，因此可以根据这两个特点采用相关的跟踪算法来进行手势跟踪及预测。

图 4.1 为采用手势跟踪技术对运动中的手势进行跟踪，目前常见的目标跟踪算法有 KCF 跟踪算法、Kalman 滤波跟踪算法和 CamShift 跟踪算法等。

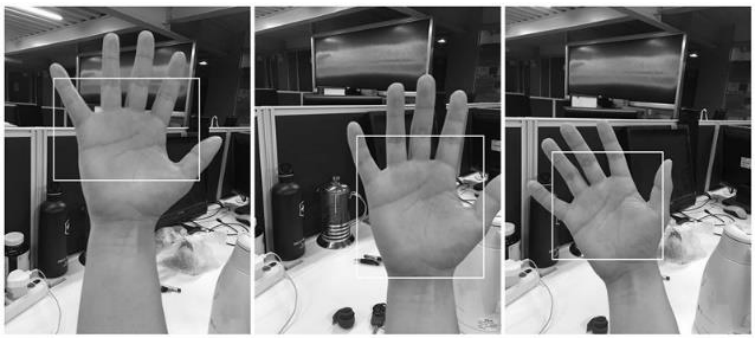


图 4.1 对视频流中手势进行跟踪

本文采用 CamShift 手势跟踪算法，CamShift（Continuously Adaptive Mean-shift）即连续自适应的 Meanshift 算法。Meanshift 算法是针对单张图片来寻找最优迭代结果，CamShift 算法则对连续的视频序列进行处理，对每一帧图片都采用 Meanshift 算法寻找最优迭代。其算法思想为：对视频序列的所有图像都做 Meanshift 运算，并将上一帧的结果(搜索窗口的中心位置和窗口大小)作为下一帧 Meanshift 算法的搜索窗口的初始值，一直迭代下去。

CamShift 算法流程图[59]如下：

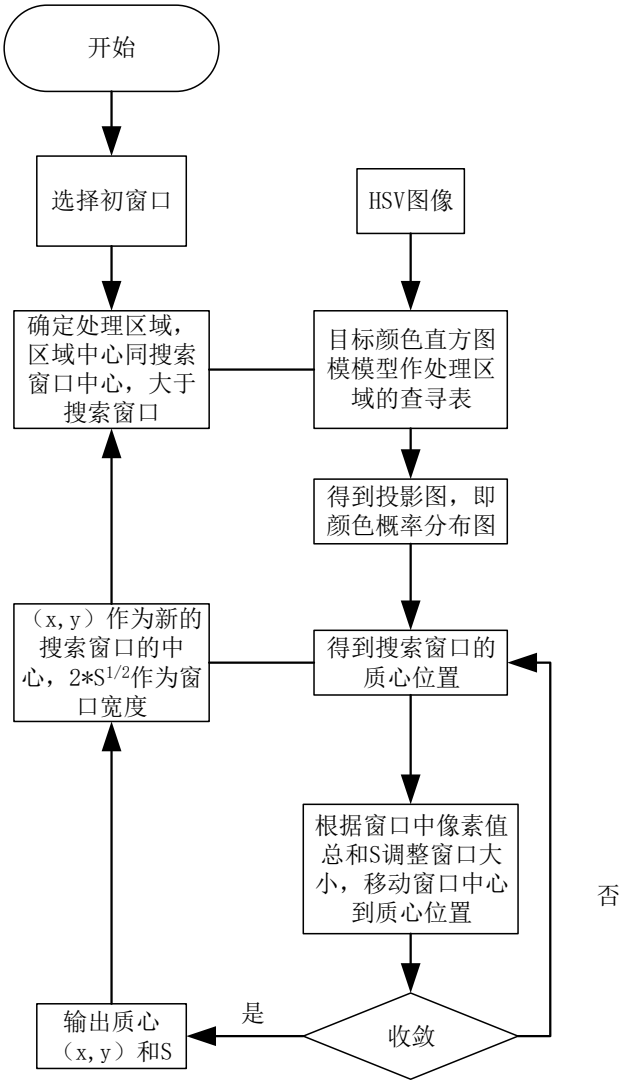


图 4.2 Camshaft 算法流程

在实验中直接采用 CamShift 算法对手势进行跟踪遇到了以下一些问题：

(1)CamShift 算法需要人为设定一个手势的初始状态，以便定位手势，初始化跟踪目标区域。

(2)在光照条件突然发生变化时，跟踪效果变差。

(3)当目标运动过快或者目标出现短暂遮挡时，跟踪会失败，搜索窗口会收缩并停留在一个很小的区域，即使目标重新在图像序列中稳定的出现，跟踪也不会自动恢复。

问题(2)是因为只在初始化时计算一次颜色直方图，之后目标颜色概率模型就不会再更新。文献[60]指出 CamShift 算法仅用颜色信息对每帧图像进行处理，可以通过在计算颜色概率分布图时引入运动信息。本文采用文献[61]的改进算法，将 CamShift 算法和帧间差分法结合，先对相邻两帧图像的 H 分量进行差分运算，将得到的帧间差分图像进行二值化，在进行一系列滤波去噪处理，最后用其包围矩形初始化 CamShift 算法的搜索框实现自动跟踪。文献[61]同时提出了一种解决(3)的方法，通过计算目标的尺寸来判断是否跟丢目标，将第 i 帧目标的

面积与阀值进行比较来判断目标是否丢失。如果目标丢失，则扩大搜索窗范围到全屏，结合帧间差分法的结果，当目标重新出现时，主动调整搜索窗，实现跟丢后重新自动跟踪。

4.2 卷积神经网络

整个手势识别系统分为两个阶段，第一阶段（如图 4.3）对手势图像进行预处理，完成背景减除和肤色分割，得到手势二值图。第二阶段为识别阶段，由一个训练好的卷积神经网络完成识别和分类。

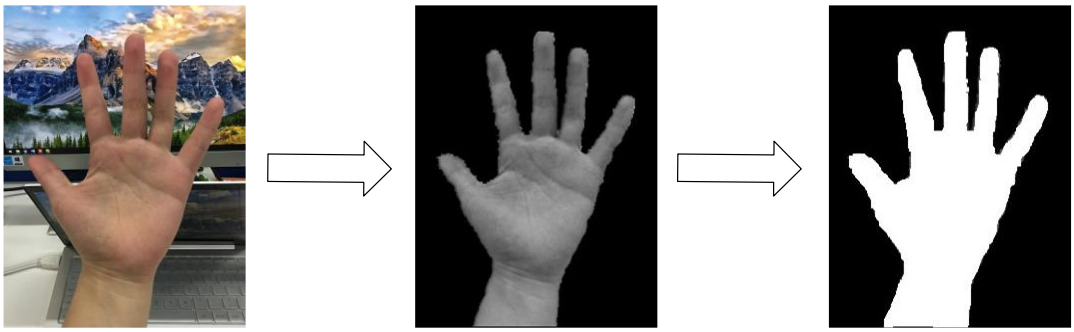


图 4.3 手势识别第一阶段

本文采用卷积神经网络来对多种手势进行分类来达到识别人类手势的目的。手势识别系统完成第一阶段工作后，将得到的二值图输入卷积神经网络，由训练完成的卷积神经网络来完成对手势的分类和识别。

卷积神经网络的网络结构包括卷积层、降采样层和全连接层。卷积层对输入图像和卷积核进行卷积求和，通过使用非线性激活函数得到输入信号的非线性表示。降采样层的作用是降低数据维度，虽然减少了许多数据，但特征的统计属性仍能够描述图像。总体来说提升了鲁棒性，有效的避免过拟合。

4.2.1 网络结构设计

卷积神经网络包含两种特殊的神经元层，即卷积层（C）和降采样层（S），整个卷积神经网络都是由卷积层和降采样层交替出现，最后和全连接（F）连接构成，并最后在输出层（O）给出结果。

本文设计的 CNN 网络如图 3.12 所示，共八层，分别为I1，C2，S3，C4，S5,F6,F7 和 O8。I1 为输入层，C2和C4是卷积层，S3和S4是降采样层,F6 和 F7 是全连接层，F6 作为前馈神经网络的输入层，F7 是神经网络的隐层，O8 是输出

层。网络结构如图 4.4 所示。

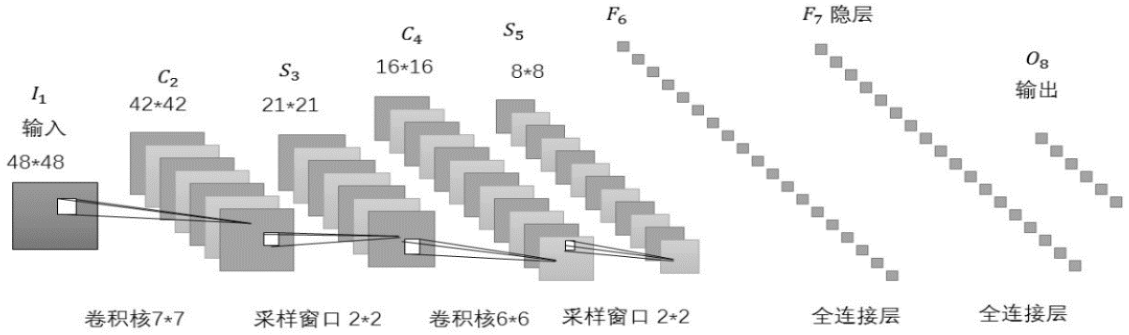


图 4.4 网络结构

其中每层的操作如下：

（1）输入层 I1 是已经归一化并二值化的48×48二值图像，可直接输入网络。

（2）C2 层是输入的二值图像经过卷积得到结果，本次采用7×7的卷积核对输入图像进行特征抽取得到 6 张大小为42×42的特征图。

（3）S3 层是对 C2 层进行的采样，采样窗口大小为2×2，得到大小为21×21的 6 张特征图。

（4）C4 是对 S3 的卷积操作，卷积核为6×6，最后得到 12 张大小为16×16的特征图。

（5）S5 为采样窗口为2×2采样，得到的 12 张大小为8×8的特征图。

（6）F6 层是由上一层 12 张大小为8×8的特征图有序展开成的长向量，作为全连阶层网络的输入。

（7）F7 层是神经网络的隐层，与 F6 层全连接，同时也与 O8 层全连接。

4.2.2 训练算法

传统的多层感知神经网络可以表示为[37]：

$$Y_n = W_n X_{n-1} \tag{4.1}$$

$$X_n = F(Y_n) \tag{4.2}$$

其中W_n是权值，其列数为X_{n-1}的维数，而其行数为X_n的维数。F 代表神经元的激活函数，将激活函数作用于神经元的输入上。Y_n为该层神经院的总输出或是输入的加权和向量。对上式采用如下求导链式规则[37]：

$$\frac{\partial E^p}{\partial y_n^i} = f'(y_n^i)\frac{\partial E^p}{\partial x_n^i} \tag{4.3}$$

$$\frac{\partial E^p}{\partial w_n^{ij}} = x_{n-1}^j \frac{\partial E^p}{\partial y_n^i} \tag{4.4}$$

$$\frac{\partial E^p}{\partial x_{n-1}^k} = \sum_i w_n^{ik} \frac{\partial E^p}{\partial y_n^i} \tag{4.5}$$

将上式变为矩阵形式：

$$\frac{\partial E^p}{\partial Y_n} = F'(Y_n)\frac{\partial E^p}{\partial X_n} \tag{4.6}$$

$$\frac{\partial E^p}{\partial W_n} = X_{n-1}\frac{\partial E^p}{\partial Y_n} \tag{4.7}$$

$$\frac{\partial E^p}{\partial X_{n-1}} = W_n^T\frac{\partial E^p}{\partial Y_n} \tag{4.8}$$

其中最简单的学习方法是梯度下降算法，其中 W 的迭代方式如下[37]：

$$W_t = W_{t-1} - \eta\frac{\partial E}{\partial W} \tag{4.9}$$

在最简单的情况下，学习率η是一个常量，在复杂的情况下学习率η在变化的。在某些情况下，η为对角矩阵的形式，或者是损失函数的逆 hessian 矩阵的估计值。

卷积网络本质上是一种输入到输出的映射，它不需要知道任何输入与输出之间精确的数学表达式，而是通过大量学习输入与输出的之间的映射关系，用已知的模式对卷积网络加以训练。卷积网络的训练过程中不断改变参数，网络便逐渐具有了输入到输出的映射能力。

卷积神经网络的训练一般采用随机梯度下降法，其训练过程主要分为前向传播和反向传播两个阶段，每个阶段有两个步骤。

第一阶段，前向传播阶段：

a)从样本集中取一个样本(X, Y_p)，将 X 输入网络；在这个阶段，信息从输入层经过逐级变换，传送到输出层，此过程中不断进行计算。

b)计算相应的实际输出O_p。

第二阶段，反向传播阶段：

a)计算实际输出O_p与相应的理想输出Y_p的差；

b)按极小化误差的方法反向传播调整权矩阵。

4.2.3 激活函数的选择

在上一节提到典型的多层感知神经网络的公式表达[37]：

$$Y_n = W_n X_{n-1} \tag{4.10}$$

$$X_n = F(Y_n) \tag{4.11}$$

上式中，F 代表激活函数。通过激活函数能够把“激活的神经元特征”保留并映射出来，这是神经网络可以解决非线性问题的关键。如果无激活函数，即激活函数为$\mathrm{f}(x) = x$。此时在每一层输出都是上一层输入的线性函数。实验表明，此时无论神经网络有多少层，输出都是输入的线性组合。在这种情况下，其效果

只与只有一个隐层时的效果相当，简而言之线性模型的表达力不够，需要加入一些非线性因素，因此引入非线性函数作为激活函数，这样使得深层神经网络不再是输入的线性组合，而可以逼近任意函数。

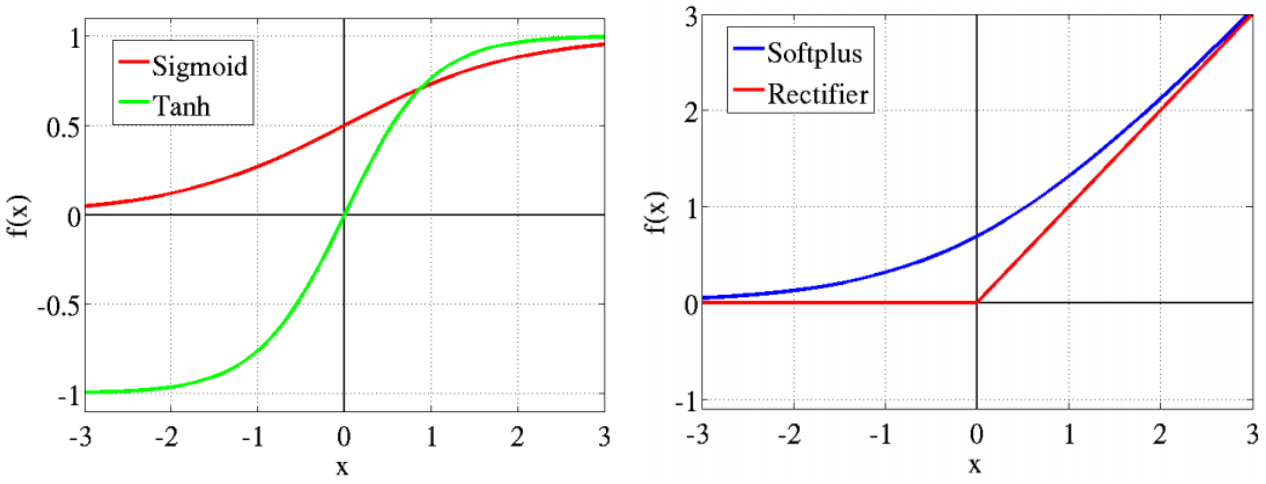
一般常见的激活函数有如下四种形式：Sigmoid、tanh、ReLu 和 Softplus。对于激活函数的选择，最早采用的是 Sigmoid 函数或者 tanh 函数。

Sigmoid 激活函数又称 S 曲线，其形式为$f(x)=\frac{1}{1+e^{-x}}$，其图形如下图所示。Sigmoid 函数输入一个实值的数，然后将其压缩到 0~1 的范围内。大的负数被映射成 0，大的正数被映射成 1。Sigmoid 激活函数在历史上流行过一段时间因为它能够很好的表达“激活”的意思，未激活就是 0，完全饱和的激活则是 1。现在 Sigmoid 函数已经很少被使用了，这主要是因为它的两个缺点[42]：

（1）Sigmoid 函数容易饱和。当输入非常大或者非常小的时候，会出现饱和现象，这些神经元的梯度接近于 0。如果初始值很大，梯度在反向传播的时候需要乘上一个 Sigmoid 的导数，会使得梯度越来越小，将导致网络变的很难学习。

（2）Sigmoid 的输出非 0 均值。这会导致后层的神经元的输入是非 0 均值的信号，而对梯度产生影响：假设后层神经元的输入都为正(例如，$f=w^Tx+b$ 中$x>0$)，那么对 w 求局部梯度则都为正。这样在反向传播的过程中 w 要么都往正方向更新，要么都往负方向更新，导致有一种捆绑的效果，使得收敛缓慢。

Tanh 激活函数与 Simoid 激活函数类似，其数学表达式为$f(x)=\tanh(x)$。不同的是它把实值输入压缩到-1~1 的范围，因此它基本是 0 均值的，也就解决了上述 Sigmoid 缺点中的第二个，所以实际中 tanh 会比 Sigmoid 更常用。但是它还是存在梯度饱和的问题。



(a) (b)

图 4.5 激活函数模型

近年来出现很多研究采用近似生物神经的激活函数：Softplus 激活函数和

ReLU 激活函数，模型如图 4.5(b)。

本文在激活函数中采用 ReLU 激活函数，其数学表达式为：$f(x) = \max(0, x)$ 。

采用 ReLU 函数解决了 Sigmoid 函数难以解决的两个问题[42]：（1）采用 Sigmoid 等函数，反向传播求误差梯度时，求导计算量很大，而 Relu 求导非常容易。（2）对于深层网络，Sigmoid 函数反向传播时，很容易就会出现梯度消失的情况（在 Sigmoid 接近饱和区时，变换太缓慢，导数趋于 0），从而无法完成深层网络的训练。另外 ReLU 函数会使一部分神经元的输出为 0，造成网络的稀疏，并且减少了参数的相互依存关系，缓解了过拟合问题的发生。

4.3 实验结果及分析

4.3.1 手势数据集处理

本文设计的手势识别方法采用 Marcel-Train 标准手势数据库对卷积神经网络模型进行训练，并验证。Marcel-Train 数据库共有 6 种不同的手势，见图 4.6，每种手势取自 10 个不同的人，包含了 3 种不同背景（亮、暗、复杂背景）下的图像，其中共有 4872 幅图像。本文提出的算法在上述手势数据集上进行训练和验证。

手势 A　　手势 B　　手势 C　　手势 D　　手势 E　　手势 F

图 4.6 手势类别定义

图 4.6 展示了需要被识别的手势。首先将每幅图片转换到 YCbCr 颜色空间，然后用本文的肤色模型进行手势分割。然后将这些经过上一步处理后的图片分别转化为48 × 48像素的灰度图和二值图（见图 4.7，图 4.8），这样可以帮助 CNN 减少计算时间。

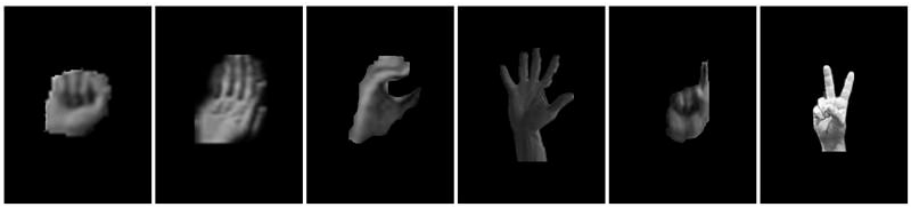


图 4.7 转化为灰度图

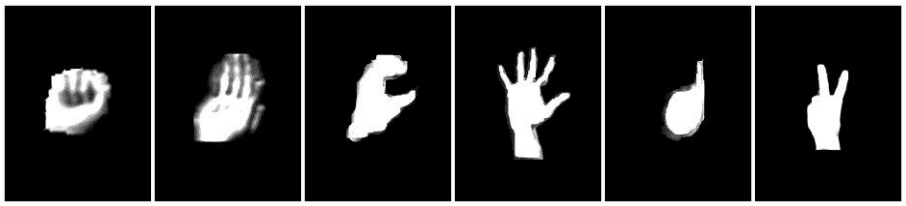


图 4.8 转化为二值图

为了证明算法的有效性，本文设置对比实验，采用未经手势分割的数据集（如图 4.6），经过手势分割并转化为灰度图像的数据集（图 4.7），经过手势分割并转化为二值图像的数据集（图 4.8）分别对卷积网络进行训练，然后对实验结果进行分析。

4.3.2 对比实验与分析

采用手势数据集来训练卷积神经网络，实验结果发现在一定训练次数范围内，训练次数与手势识别率呈正相关关系。从表 4.1 可以看到，当训练次数较少时，手势识别率较低。随着网络训练次数增加，识别率明显开始上升。

表 4.1 训练次数与识别率关系

训练次数	500	1000	2000	3000	5000	10000
平均识别率%	85.5	93	94.3	95	95.2	95.3

由实验数据分析可知，网络需要经过多次训练后才能得到较好的识别效果，其原因是受手势扭曲程度和拍摄角度等因素影响，最终的结果是阻碍网络的权值进入稳定点，因此在每次训练结束后，手势的均方误差依然很大[57]。通过增加网络的训练次数来降低均方误差，使权值进入稳定点。由数据分析，当训练次数达到 7500 次时，均方误差开始趋于稳定，当训练次数达到 10000 次时，网络权值达到稳定。

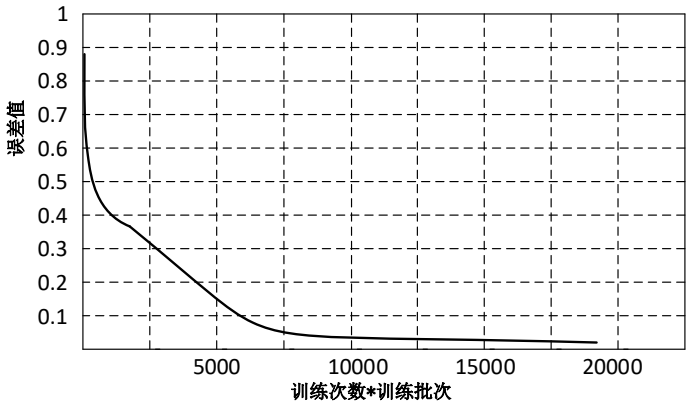


图 4.9 误差值和收敛曲线

本文为了验证手势识别算法的性能分别设置了横向与纵向两组对比实验。

实验一：纵向对比

将经过三种不同方式处理后的数据集对本文建立的卷积神经网络进行训练。表 4.2、表 4.3 和表 4.4 分别为经三种数据集训练网络后系统的手势识别率。

表 4.2 未经处理数据集的识别率

手势类别	A	B	C	D	E	F	平均识别率%
样本识别率%	98.3	97.8	99.3	99	93	96.5	97.3

表 4.3 转化为灰度图的识别率

手势类别	A	B	C	D	E	F	平均识别率%
样本识别率%	99.3	99	99.5	100	99	96.5	98.8

表 4.4 转化为二值图的识别率

手势类别	A	B	C	D	E	F	平均识别率%
样本识别率%	99	99	99.5	100	93	96.5	97.8

表 4.3 与表 4.4 是将手势数据集进行手势分割后分别转化为灰度图像和二值图像手后得到的识别率，表 4.2 为未经处理的数据集得到的识别率。将表 4.3 与表 4.4 数据相比，明显可以看到手势 E 在二值图情况下的识别率明显低于灰度图的情况。出现该情况的原因是，在数据集中，E 手势部分图片是一根手指，其他部分是两根手指。二值图像在特定的角度下，B 手势和 E 手势非常相似。但是在灰度图像的情况下，图像会有更多的细节，例如手指位置，这些细节可以帮助卷积网络更准确的进行分类。将表 4.2 与表 4.4 比较，可以明显的看出，手势分割可以帮助卷积网络排除背景的干扰，达到提高识别率的作用。

实验二：横向对比

在现有的相关文献中，已有文献[10,38]采用卷积神经网络的识别手势，但是这些文献均采用自采集手势数据集的方式来训练卷积网络。为了验证本文算法性能，将对文献[10,38]方法进行对比。文献[10]对手势定位，精确得到手势位置后，在通过混合高斯肤色模型将手势图像转化为二值图，最后再输入卷积神经网络。文献[38]对图像进行简单的预处理后转化为二值图，然后将手势图像置于 120*120 大小区域中心，再统一缩放到 32*32，最后输入卷积神经网络。

表 4.5 手势识别性能比较

参考文献	平均消耗时间/幅/ms	平均识别率%
[10]	6	98
[38]	12	89
本文算法	5	98.8

表 4.5 显示了本章与其他工作的对比。由于采用公共数据集 Marcel-Train 标准手势数据库，该数据集中的手势都是经过初步分割，如图 4.6，图中基本已经排除了人体其他肢体干扰，也没有其他类肤色区域，基本能够满足文献[10]和文献[38]的要求。从实验数据可见本文与文献[10]相比每幅图片的识别消耗时间缩短了 1ms，平均识别率提升 0.8%，优势并不明显。由于本文的研究重点之一是提升手势识别算法在复杂背景下的适应性，因此增加了一个测试集，其中图片如图 3.8(a)，该测试集中均是未经过初步手势分割，实验结果见表 4.6。

表 4.6 复杂背景下的手势识别性能比较

参考文献	平均消耗时间/幅/ms	平均识别率%
[10]	85	86
[38]	117	74
本文算法	53	94

由表 4.6 中数据可以看到，与表 4.5 的数据相比各个算法的性能衰减了很多。这主要是测试数据未经过初步分割，因此需要对全图进行扫描。由第三章的实验结果可知文献[10]的手势分割速度是优于本文算法的，但是由于文献[10]并不能将类肤色区域在手势分割中去除，在后续的计算中需要消耗更多的时间。因此在复杂背景下本文的算法更具有优势。

4.4 本章小结

本章首先介绍了手势跟踪技术，以及本文手势识别系统的两个阶段。然后对卷积网络的结构进行了介绍，分析了卷积网络的训练算法和训练过程。接着讨论了几种激活函数的优点和缺点。最后对实验结果进行分析。

第5章 类人机器人手势识别系统的实现

本文针对类人机器人的应用需要而提出的一种手势识别算法，因此算法在类人机器人上的表现是一个非常重要评价因素。类人机器人的手势识别实验需要在PC平台和NAO机器人平台上先后进行。PC平台利用Naoqi框架的远程模块，可以方便地进行实验数据分析和算法调试。NAO机器人平台则是在将手势识别算法的移植实验完成后，让NAO离线运行已经打包成本地模块的手势识别程序来进行手势识别。本章将对实验载体进行详细介绍，并对已完成的手势识别算法进行移植实验。

5.1 实验环境

本文设计的是一个能够在类人机器人上实现的手势识别算法，首先在PC平台上完成算法的设计，然后将算法移植到类人机器人上并进行优化，最后实现机器人的手势识别。在整个实验中需要用到多个硬件、软件实验平台，其中软件平台包括：Caffe、OpenCV、NAOqi和Choregraphe等。而硬件平台包括：笔记本电脑、NAO人形机器人。

5.1.1 软件环境简介

本文涉及到的代码编写全部采用Python编程语言，采用Python主要有以下几方面的考虑：

（1）NAOqi机器人框架

NAOqi是一个基于Gentoo操作系统的跨平台分布式Gun/Linux操作系统，是Aldebaran Robotics公司专门为旗下的机器人设计的机器人框架，它能满足机器人如并行性、资源、同步等需要。

Aldebaran Robotics公司选择采用C++语言开发实时模块，使用Choregraphe和Python语言开发行为模块，与硬件调动相关的模块都是采用C++开发。Python语言因其嵌入式解释器被官方选中，而被推荐来进行应用程序的开发。NAOqi可以从python中调用C++的函数，并且官方文档提供了大量的python程序示范。总而言之，采用NAOqi提供的python接口可以很方便的进行NAO机器人的应用程序开发。

（2）Opencv计算机视觉库

Opencv是一个被广泛使用的开源计算机视觉库，它是由一系列C函数和少量C++构成的库函数，可以实现模式识别与图像处理领域的很多通用算法，可

应用于图像处理、运动分析、三维重建以及模式识别等领域的开发。Opencv 也是 NAO 机器人官方推荐的视觉库，利用 Opencv 提供的 python 接口可以直接将 Opencv 配置到 NAO 机器人的开发环境中。

（3）Caffe 深度学习框架

Caffe 是一个 Python 库，是一个深度学习框架。Caffe 中包含了大量的深度学习 Python 包，可以比较轻松的编写深度学习模型。

5.1.2 硬件环境简介

（1）电脑一台。运行环境为 Ubuntu14.04 32 位操作系统，Intel core i5 处理器，CPU 运行频率为 2.30GHZ，内存为 8G。

（2）NAO 人形机器人一台。

NAO 是一个升高约 58CM 的可编程仿人机器人，其关键组件如下：

1）传感器：2 个摄像头、4 个麦克风、1 个超声波距离传感器、2 个红外线发射器和接收器、1 个惯性板、9 个触觉传感器及 8 个压力传感器。

2）自我表达的器件：语音合成器、LED 灯及 2 个高品质扬声器。

3）一个 CPU（位于机器人头部），运行一个 Linux 内核，并支持 ALDEBARAN ROBOTICS 公司自行研制的专有中间件 NAOqi。第二个 CPU 位于机器人躯干内。

4）一个 55 瓦时电池，可为 NAO 提供 1.5 小时自主时间。

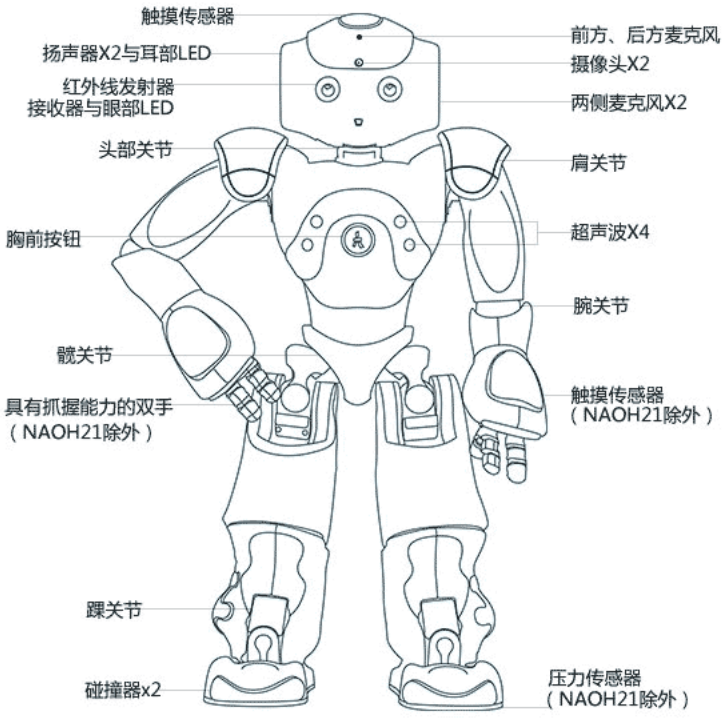


图 5.1 NAO 机器人硬件分布图

5.2 NAO 机器人平台

5.2.1 NAOqi 框架

NAOqi 是 NAO 机器人运行的主软件，NAOqi 框架是 NAO 编程的程序框架。在 Nao 机器人运行的应用程序实际上是 NAOqi 的可执行代理程序，当机器人启动时，这个代理程序会自动加在一个定义库的首选项文件，每一个库包含一个或多个模块，每个模块可以扩展多种方法，NAO Framework 如图 5.2 所示：

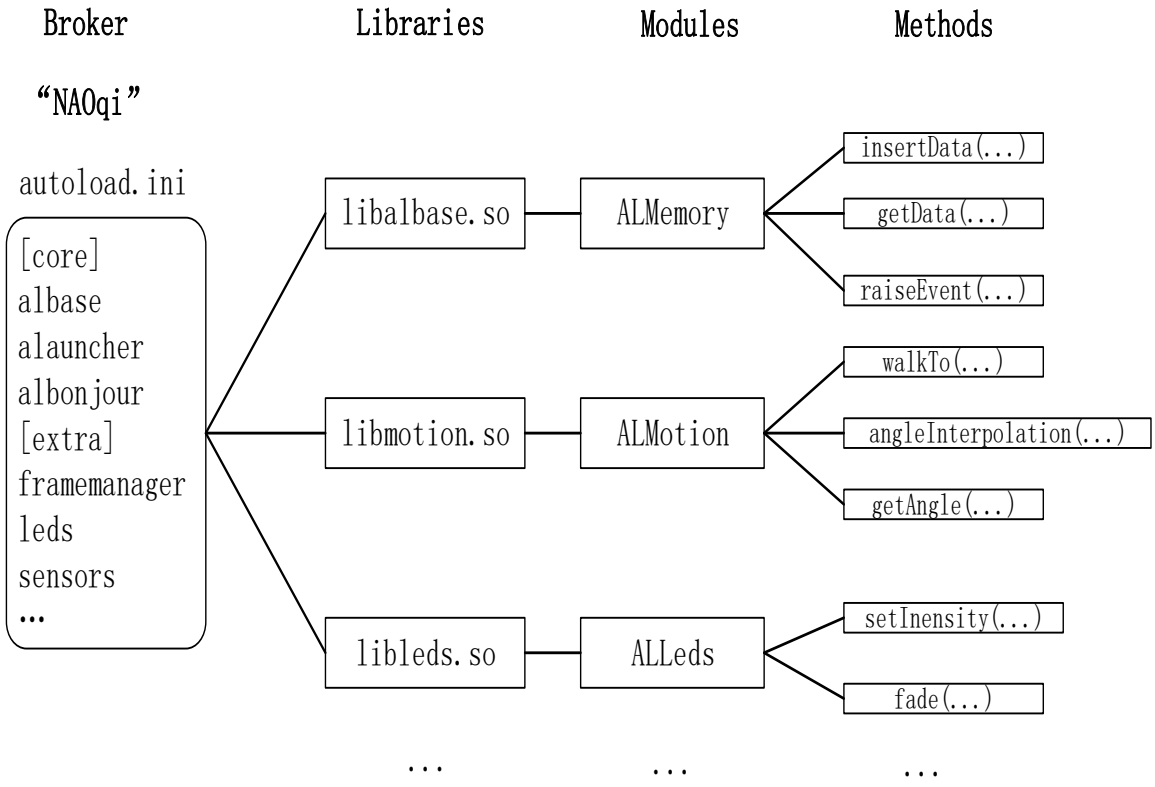


图 5.2 NAOqi 的架构

NAOqi 是 NAO 的主代理程序，作为可执行文档，监听机器人的端口和 IP 地址。Module 即模块，是库中的一个类，当机器人启动时，库随之加载，系统会自动实例化库中包含的模块。模块可分为远程模块和本地模块两种。如果是远程模块，这个模块会被编译成一个可执行程序，用于远程执行。如果是本地模块，这个模块会被编译成一个库，可在机器人上运行。每一个模块都包含若干个方法。当调用某个模块的方法时，需要创建这个模块的代理（Proxy），相当于类的实例，即对象。

5.2.2 NAO 机器人的视觉系统

NAO 机器人面部安装了两个相同的摄像头，两摄像头呈纵向排列如图 5.3，摄像头参数见表。

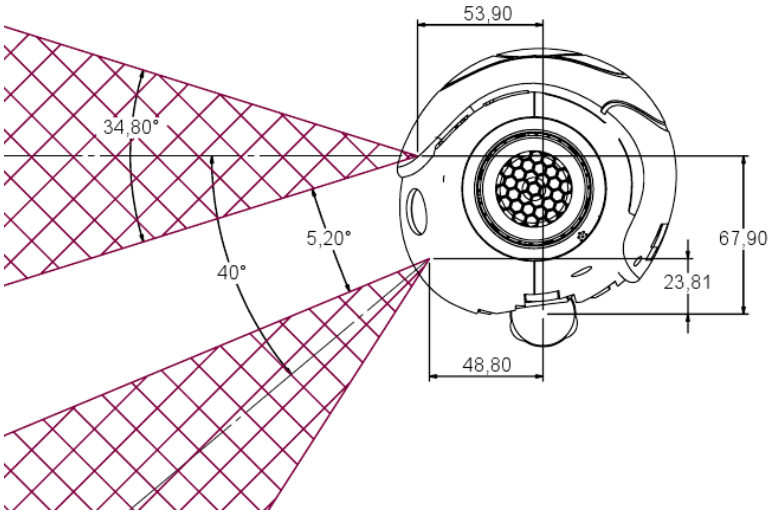


图 5.3 NAO 摄像头分布及参数

表 5.1 摄像头参数

	参数
摄像头输出	YUV422 颜色空间
视野	沿曲线走动
聚焦范围	鞠躬
聚焦类型	踢腿

由上图可见 NAO 机器人有两个摄像头，但是由于它们呈纵向排列，受视角影响，在 NAO 正视前方时，下摄像头是对着地面。根据官方文档介绍，这种设计是为了让 NAO 机器人在机器人足球赛的时候方便用来识别视野里面的足球。因此这种摄像头的排列方式对本文的手势识别并没有用处，我们只能利用上摄像头来捕捉手势。受硬件的限制，在不增加外接视觉传感器的情况下，NAO 机器人只能采用单目摄像头手势识别的方案。

5.2.3 体系结构

Aldebaran Robotics 公司为 NAO 的视觉系统专门设计了一套体系结构（图

5.3），其他视频源通过模拟件来使用这个体系结构。

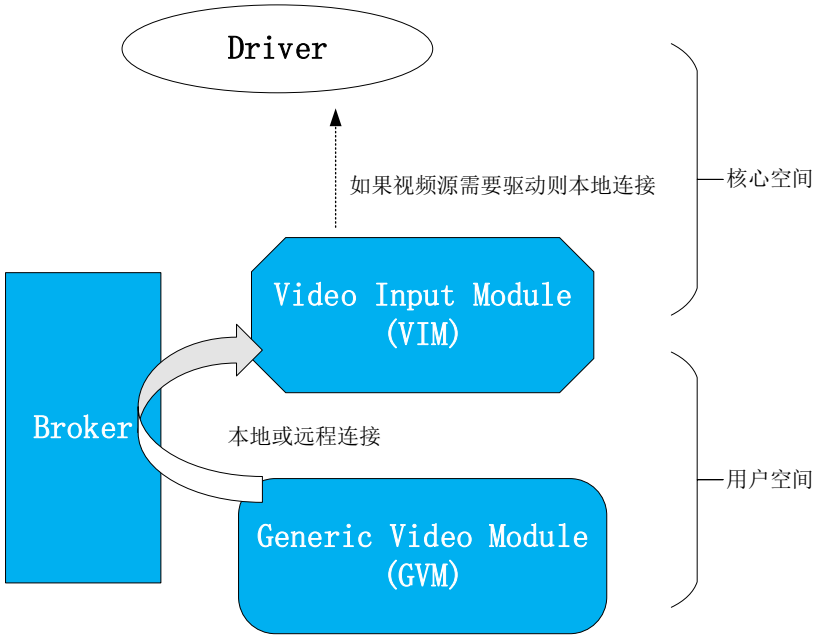


图 5.4 NAO 机器人视觉系统体系结构

（1）视频输入模块（VIM）

每个和视觉有关的模块都可以通过 VIM 访问视频流，通过 VIM 既可直接访问视频源的原始数据，也可访问根据选定的分辨率和颜色空间转换后的数据。VIM 主要有如下作用：

1）通过 I2C 总线打开摄像头器件。

2）在流模式下运行 V4L2 驱动程序（驱动程序将创建一个 n 个元素的循环缓冲区来抓去视频流）。

3）接收来自 GVM 的订阅，为每个订阅创建一个存储原始数据的 ALImage，以及一个存储转换数据的 ALImage。

4）当一个 GVM 要求图像时，选择要提供的图像。

5）当一个 GVM 注销时，删除相应的 ALImages。

6）当接到命令或不再有 GVM 订阅时，停止驱动程序，关闭视频器件。

（2）通用视频模块（GVM）

GVM 需要一个特定的图像格式来进行处理，因此 GVM 通过代理程序发送请求，注册 VIM 参数，参数包括：分辨率、颜色空间和帧率。VIM 会把视频流转换为所需格式（如果 GVM 需要访问的是视频源的原格式，可直接访问原始数据）。

5.3 NAO 机器人的手势识别算法移植

本节是将已经训练好的手势识别系统移植到类人机器人上，在进行移植前后针对 NAO 机器人的反馈进行了一系列算法优化，具体见 5.3.4 小节。

5.3.1 远程模块和本地模块

NAO 机器人可以本地或远程运行模块，本地运行模块是机器人上的 NAOqi 直接运行模块，每个模块都是 NAO 机器人的一个库。远程运行模块是 PC 连接上 NAO 后，通过 PC 上的 NAOqi 运行模块。开发时使用远程运行模块，模块可以通过任意 IDE 来进行调试（见图 5.4），并且连接至机器人可执行档的方式与使用一个本地库的方式完全相同。本文 NAO 的手势识别系统通过调用远程模块（即一个新的可执行档）的方式进行开发和调试，在这种情况下，算法的核心计算和数据处理都是在 PC 上完成，而 NAO 机器人只是提供硬件支持（包括摄取图像和做出反馈等）。当应用开发完成，就需要将该程序移植到 NAO 机器人的 NAOqi 上编译为 NAO 的应用程序。

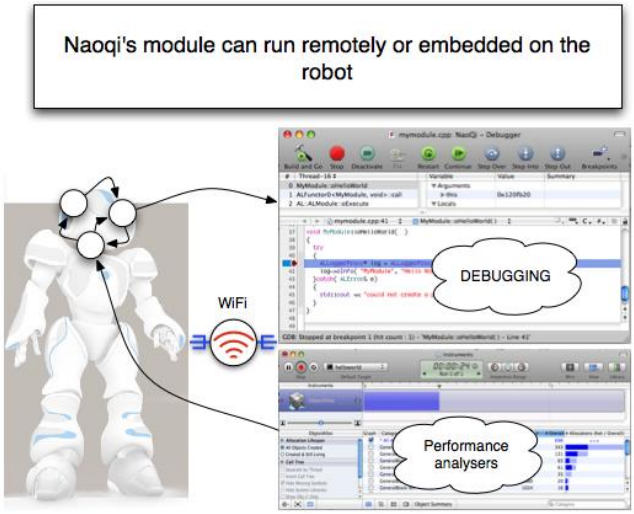


图 5.5 NAOqi 远程模块使用

5.3.2 使用 Cmake 交叉编译

开发完成的代码要想能够成为 NAOqi 的本地模块，需要通过 Cmake 进行交叉编译，编译的目的是为了保证编译时使用正确的优化标志，以及使用正确的库版本进行链接。

NAOqi SDK 中已经包含 Opencv，当使用 Opencv 库时，只需要在 CmakeList 中添加：qi_use_lib(mylib Opencv)。但是 NAOqi SDK 提供的 Opencv 版本和本文需要的 Opencv 版本不符，且本文需要调用 Opencv 的 python

接口，需要进行这方面的配置，因此需要进行重新编译。具体操作如下：

（1）清除老版本，从安装包进行 uninstall。

（2）通过 build 源代码安装需要版本的 Opencv：

1）获取源代码；

2）解压；

3）进入解压目录，创建 build 目录，并进入该目录执行：mkdir build cd build；

4）使用 Cmake 进行设置；

5）Build Opencv；

6）安装。

交叉编译步骤如下：

（1）把数据放入“/home/nao”。

（2）将 Aldebaran 提供的交叉工具链（cross-toolchain）保存在“/path/to/etc”。

（3）在一个交叉构建目录里配置 Cmake，并把“toolchain-geode.cmake”文件规定为工具链文件。

手势识别算法中用到的其他软件包如 Caffe 也要进行交叉编译。经过 Cmake 的交叉编译后，需要将新建模块加入到 NAO 机器人上的 NAOqi。

5.3.3 模块移植

将交叉编译得到的“lib***.so”文件复制到 NAOqi 的相关文件夹中，操作命令如下：

scp ./sdk/lib/naoqi/module nao@192.168.1.100: nao/naoqi/lib/naoqi/module. so

然后配置机器人启动文件，将 /opt/naoqi/preference/autoload. ini复制到 /home/nao/naoqi/preference下。最后修改 autoload.ini 文件后重启。

5.3.4 算法优化

根据 Aldebaran Robotics 公司提供的技术手册，NAO 机器人搭载两个 AMD GEODE x86 500MHz CPU，SDRAM 为 256 MB ；尽管目前类人机器人提供了强大的 CPU，但是在这类嵌入式设备上运行深度卷积网络来完成复杂的手势识别任务任然比较吃力。为了保证手势识别算法在机器人上有良好表现，需要对算法进行优化。

本文进行了一系列算法的优化，其中包括对 CNNs 模型的压缩。CNN 通常包含卷积层和全连接层，它们分别支配着整个网络的计算消耗和内存消耗，对卷积层和全连接层的压缩一直是模型压缩中的热门问题。Denton[62]等人指出全连

阶层的权矩阵能够通过奇异值分解来进行压缩，并且该压缩不会对预测精度产生重要影响。文献[63-66]等后续研究则分别提出矢量量化、散列技术、循环投射和张量分解训练等方法来对网络模型进行压缩。本文采用文献[67]的方法一次性对整个网络进行压缩，该方法分为三步：首先，选择变分贝叶斯矩阵分解模型排序；然后，对每层的核张量进行 Tucker 分解；最后进行微调来恢复准确度的累积损失。

5.4 NAO 机器人手势交互实验

与 NAO 机器人进行手势交互并不能像在 PC 平台上一样直接得到反馈的数据，因此本文希望通过 NAO 机器人的反馈来判断手势是否被正确识别，对于 NAO 机器人做出的反馈，本文设计一组特定的动作与之相对应。

对 NAO 机器人的运动行为进行设计并不是一件轻松的事，因为涉及到运动学方面的问题，所幸 Aldebaran Robotics 公司提供的官方开发平台 Choregraphe 中已经存在了许多动作，可以直接将这些动作利用起来。我们根据六种手势设置了 6 个动作指令，NAO 进行手势识别后，会做出相应的反馈动作。动作指令和反馈动作如表 5.2 所示：

表 5.2 动作指令和反馈动作

动作指令	反馈动作
手势 A	稍息动作
手势 B	双脚踱步摇晃
手势 C	向前直行
手势 D	沿曲线走动
手势 E	鞠躬
手势 F	踢腿

针对以上表格六种指令，分别进行实验如图 5.5 所示：

图 5.6 动作指令

NAO 机器人对手势 A 进行识别后，将会做出稍息的反馈动作，如下动作序列图 5.6 所示：

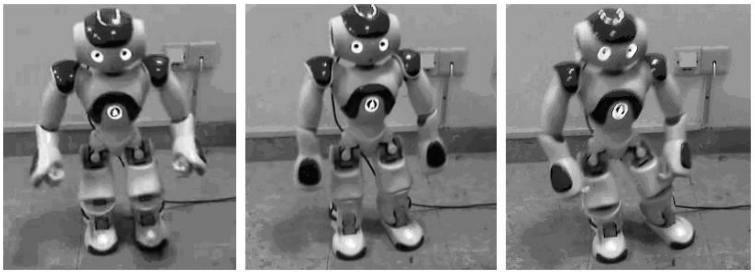


图 5.7 稍息动作

NAO 机器人对手势 B 进行识别后，将会进行双脚踱步摇晃，如下动作序列图 5.7 所示：

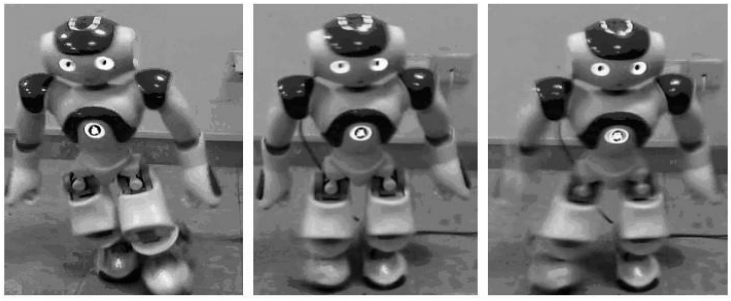


图 5.8 左右踱步摇晃

NAO 机器人对手势 C 进行识别后，将会向前直行，如下动作序列图 5.8 所示：

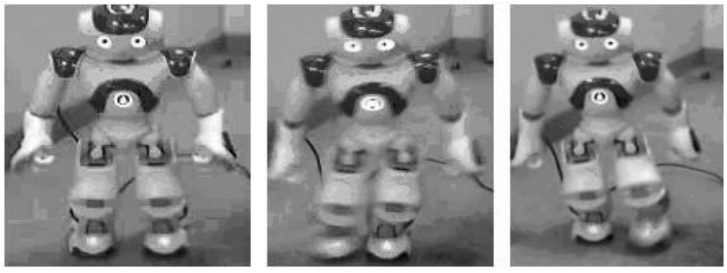


图 5.9 向前直行

NAO 机器人对手势 D 进行识别后，将会沿曲线走动，如下动作序列图 5.9 所示：

图 5.10 沿曲线走动

NAO 机器人对手势 E 进行识别后，将会鞠躬，如下动作序列图 5.10 所示：

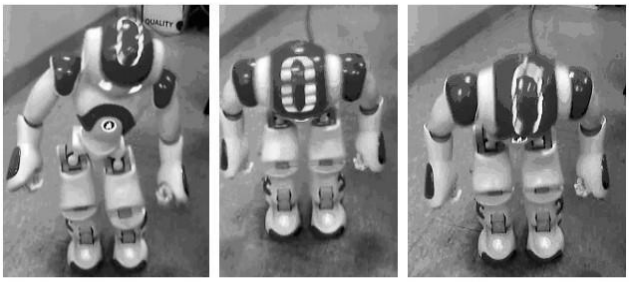


图 5.11 鞠躬

NAO 机器人对手势 F 进行识别后，将会踢腿，如下动作序列图 5.11 所示：

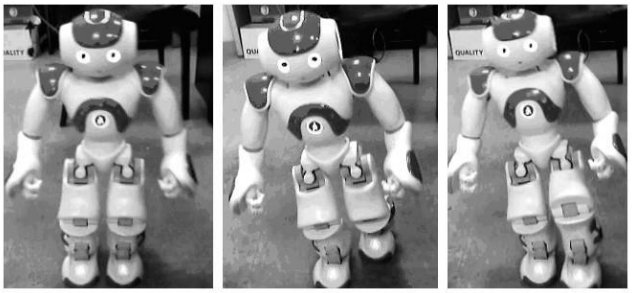


图 5.12 踢腿

以上六个手势指令即人发出的动作命令，NAO 机器人在识别了手势后，将会理解手势具体代表的含义并做出相应的反馈动作，完成人机交互。本文在人机交互中分别对每种手势进行了 200 次完整的交互实验。实验结果如下表 5.3。

表 5.3 人机交互实验识别率

动作指令	识别次数	识别率%
手势 A	189	94.5
手势 B	192	96
手势 C	196	98
手势 D	194	97
手势 E	182	91
手势 F	190	95

由表 5.3 中数据可见，本算法在移植到 NAO 机器人后，算法的准确性与 PC 上的结果变化不大，说明本算法在类人机器人平台上是有效的。

5.5 本章小结

本章主要介绍了本文所用到软件、硬件平台，对类人机器人实验平台进行了详细的介绍。然后介绍了 NAO 机器人的应用软件开发方式和交叉编译过程，将第四章提出的手势识别算法移植到 NAO 机器人上，进行调试和优化，最终实现了 NAO 机器人上的手势识别。

结论

随着科技的进步，人们的生活进入了智能时代。智能设备的更新换代越来越快，新设备提供的交互体验不断冲击着人们的认知。无论是人与智能机器人的交互，还是虚拟现实技术中的人与“虚拟场景”交互，其本质依然是人与计算机交互。传统的交互方式显得比较呆板，且效率太低，在这种情况下，势必需要探索出新的交互方式来解决这一难题。目前新交互方式层出不穷，其中手势交互颇有新意，又具有很强的可操作性，因而逐渐成为了热门。

伴随着手势交互诞生的手势识别技术是一个热门的研究领域，之所以热门正是因为还有许多难题没有得到完美解决。目前随着深度学习的兴起，很多新的方法进入我们的视野，我们正是应该借助这些新方法来解决这些难题。本文也正是从这个角度出发，借助目前比较热门的卷积神经网络算法，寻求找到一种能够在类人机器人上具有很好性能的手势识别方法。本文聚焦于手势识别算法的两个经典问题：手势分割和手势识别，探索它们各自的解决方案。最终设计出一个基于单目摄像头的手势识别系统，该系统与前人的成果相比，重点聚焦于算法的实用性，尽量让算法不仅仅局限于实验室条件下（提出诸多的限制条件），而是可以真正用于实际应用。在这个思想指导下，在完成该方案的设计后，将这套方案对类人机器人平台进行移植，最终在类人机器人平台上实现了手势识别。

针对手势识别，本文把整个过程划分为三个阶段：手势分割（检测、定位、分割）、手势跟踪和手势识别。在检测阶段采用图像处理方法将运动区域分隔开，计算出其位置。再通过肤色模型将干扰区域排除，得到精确的手势区域。在跟踪阶段利用 CamShift 算法不断扫描手的位置，实现跟踪环节。到识别阶段，由于前两阶段已经对手势区域进行了处理，只需要直接将图像输入卷积神经网络完成分类即可。由于本文的工作侧重与对手势分割和手势识别的研究，对手势的跟踪并没有做到最优，因此在实际工作中可能会有许多不足。

本文围绕类人机器人手势识别算法的设计与实现主要完成了以下工作：

第一，在手势分割方面，本文分析了各种手势分割算法的缺点，发现主流的利用肤色模型分割手势难以避免类肤色区域的干扰，而利用运动信息来对手势进行分割得到的结果不包含任何的肤色信息。因此提出了一种将肤色信息和运动信息结合起来的分割方法，使得该手势分割算法在复杂背景下也能够很好的排除其他运动物体和类肤色区域的干扰。

第二，在手势识别方面，对传统的手势识别算法进行了回顾，分析了它们的优点和缺点，发现大多数手势识别算法难以避免对手势特征进行精心设计，这往

往需要耗费大量的工作。而 CNN 所具有的诸多特性恰好可以避免人工设计手势特征，因此将 CNN 卷积神经网络的优点利用起来，避免人工设计和提取特征。并对现有卷积网络进行优化，实现了一种可以用于手势识别的 CNN 算法。

第三，完成对手势分割和手势识别的研究工作后，将工作的成果在 NAO 机器人的硬件平台上进行移植。在移植过程中，发现将训练好后的识别模型直接移植到 NAO 上，NAO 对手势识别正确率下降，且反应较慢。考虑到其原因可能是嵌入式设备的性能限制，于是针对性能进行优化，完成卷积网络模型进行压缩等工作，最后取得不错的效果。

本文是基于单目摄像头的硬件条件下进行的手势识别研究，这主要是考虑到目前绝大多数活跃着的硬件设备仍然是基于单目摄像头，并且目前对于单目摄像头下的嵌入式设备还没有很好的手势识别方案。但是随着科技的不断发展和硬件设备的不断更新换代，越来与多的先进视觉传感器开始进入我们的视野。很多问题仅通过软件是难以解决的，我们可以借助这些先进硬件设备的性能，将很多通过算法优化难以解决的问题最终完美地解决。本文提出的类人机器人手势识别算法虽然取得了一定的研究进展，但是在实验中也发现了一些不足，以后将着重对这些方面进行改进：

（1）手势识别系统在复杂背景下具有不错的适应性，但是这是在单人目标的情况下实现的。在实验中发现，当人打手势时，如果在摄像头范围内存在其他人进行运动行为，本文融合肤色信息和运动信息的手势分割算法将失效，会将其他人的肤色区域分割出来，在这种情况下分割效果不佳。因此考虑到这类场景还需要对算法进行改进。

（2）对类人机器人平台进行硬件改造升级，增加深度视觉传感器。目前基于深度视觉传感器的手势识别已经逐渐成为了领域热门，深度视觉传感器捕捉的深度信息能够帮助我们更好更快的从背景中将手势提取出来，这是二维信息难以企及的。如果真的要使人与类人机器人的手势交互更为有效，让类人机器人搭载这种新的视觉传感器是必然趋势。

（3）对深度视觉进行研究，找到一种可以在嵌入式设备上具有良好表现的基于深度信息的手势分割方法。目前将深度信息用于手势分割的研究很多，但是这些研究无一不表明，深度信息的引入将增加计算量，然而嵌入式设备的不足就是计算能力相对薄弱。因此，在借助深度视觉传感器的硬件性能提升对手势分割效果的同时也要考虑如何在提升性能的情况下减小付出的代价，这也许可以通过对算法的精心设计来实现。

参考文献

[1]易靖国,程江华,库锡树.视觉手势识别综述.计算机科学,2016,43(z1):103-108.

[2]周建英,吴小培,张超,等.基于滑动窗的混合高斯模型运动目标检测方法.电子与信息学报,2013,(7):1650-1656.

[3]严权峰,王岳斌,白天,等.基于压缩感知的实时手势检测和跟踪算法.计算机工程与应用,2016,52(20):182-187,230.

[4]冯志全,杨波,郑艳伟,等.基于特征点分布分析的手势特征检测方法.计算机集成制造系统,2011,17(11):2333-2342.

[5]刘军,田国会,李荣宽,等.智能空间下基于手势识别的人机交互.北京联合大学学报,2010,24(2):14-17.

[6]Bergh M V D, Gool L V. Combining RGB and ToF cameras for real-time 3D hand gesture interaction. Applications of Computer Vision, Kona, HI, USA,2011: 66-72.

[7]Priyal S P, Bora P K. A robust static hand gesture recognition system using geometry based normalizations and Krawtchouk moments. Pattern Recognition, 2013, 46(8): 2202-2219.

[8]Gao J P, Wang Y J, Yang H, et al. Elliptical model based on KL transform for skin color detection. Journal of Electronics & Information Technology, 2007, 29(7): 1739-1743.

[9]Greggio N, Bernardino A, Laschi C, et al. Fast estimation of Gaussian mixture models for image segmentation. Machine Vision and Applications, 2012, 23(4): 773-789.

[10]Lin H I, Hsu M H, Chen W K. Human hand gesture recognition using a convolution neural network. IEEE International Conference on Automation Science and Engineering, Taipei, Taiwan ,2014: 1038-1043.

[11] Lahamy H, Litchi D. Real-time hand gesture recognition using range cameras. 2010 Canadian Geomatics Conference and Symposium of Commission I, ISPRS Convergence in Geomatics - Shaping Canada's Competitive Landscape, June 15, 2010 - June 18, 2010, Calgary, Alberta, Canada,2010.

[12]Ren Z, Yuan J, Meng J, et al. Robust Part-Based Hand Gesture Recognition

Using Kinect Sensor. IEEE Transactions on Multimedia, 2013, 15(5): 1110-1120.

[13] 李丹娇, 彭进业, 冯晓毅, 等. 结合 CSS 与傅里叶描述子的手势特征提取. 计算机工程, 2012, 38(6):178-180.

[14]Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[15] 严焰, 刘蓉, 黄璐, 等. 基于 HMM 的手势识别研究. 华中师范大学学报(自科版), 2012, 46(5):555-559.

[16]Ma L, Zhang J, Wang J. Modified CRF algorithm for dynamic hand gesture recognition. Control Conference, Nanjing, China ,2014: 4763-4767.

[17] 赵新龙, 方贵盛, 沈莉芳. 基于 BP 神经网络的草图编辑手势识别. 浙江水利水电学院学报, 2006, 18(4):31-33.

[18]Elmezain M, Al-Hamadi A, Michaelis B. Real-time capable system for hand gesture recognition using hidden markov models in stereo color image sequences. British Journal of Dermatology, 2008, 144(2): 424.

[19] Głomb P, Romaszewski M, Sochan A, et al. Unsupervised Parameter Selection for Gesture Recognition with Vector Quantization and Hidden Markov Models. Human-Computer Interaction - INTERACT 2011 - Ifip Tc 13 International Conference, Lisbon, Portugal, September 5-9, 2011, Proceedings, 2011: 170-177.

[20] Murthy G R S, Jadon R S. Hand gesture recognition using neural networks. Advance Computing Conference, Patiala, Punjab, India,2010: 134-138.

[21] Li M, He Y. Nonlinear system identification using adaptive Chebyshev neural networks. IEEE International Conference on Intelligent Computing and Intelligent Systems, Xiamen, China ,2010: 243-247.

[22] Tusor B, Varkonyi-Koczy A R. Circular fuzzy neural network based hand gesture and posture modeling. Instrumentation and Measurement Technology Conference, Austin, TX, USA ,2010: 815-820.

[23] 徐成, 马翌伦, 刘彦. 一种基于嵌入式系统实时交互的手势识别方法. 计算机应用研究, 2011, 28(7):2782-2785.

[24]Yi L. Hand gesture recognition using Kinect. 2012 IEEE International Conference on Computer Science and Automation Engineering, Zhangjiajie, China, 2012: 196-199.

[25]曹雏清, 李瑞峰, 赵立军. 基于深度图像技术的手势识别方法. 计算机工程, 2012, 38(8):16-18, 21.

[26]李瑞峰,曹雏清,王丽.基于深度图像和表观特征的手势识别.华中科技大学学报（自然科学版）,2011,39(z2):88-91.

[27]丁毅,曹江涛,李平,等.复杂背景下的手势识别算法研究.自动化技术与应用,2016,35(8):113-116.

[28] Garcia C, Tziritas G. Face detection using quantized skin color regions merging and wavelet packet analysis. IEEE Transactions on Multimedia, 1999, 1(3): 264-277.

[29] 雷明,张军英,董济扬.一种可变光照条件下的肤色检测算法.计算机工程与应用,2002,38(24):123-125.

[30]尚可可,刘迎,路毅行,等.HSV 色彩空间中的肤色特征及其新的识别参量.光电子·激光,2007,18(11):1391-1393.

[31]卢章平,孔德飞,李小蕾,等.背景差分与三帧差分结合的运动目标检测算法.计算机测量与控制,2013,21(12):3315-3318.

[32] Fukunaga K, Hostetler L. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory, 1975, 21(1): 32-40.

[33] Yizong C. Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790-799.

[34] Comaniciu D, Meer P. Robust analysis of feature spaces: color image segmentation. Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Puerto Rico, 1997: 750-755.

[35] Bradski G R. Real time face and object tracking as a component of a perceptual user interface. Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on, Princeton, NJ, USA, USA ,1998: 214-219.

[36] Hinton G E, Salakhutdinov R R. Reducing the Dimensionality of Data with Neural Networks. Science, 2006, 313(5786): 504-507.

[37] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets. Neural Computation, 2006, 18(7): 1527.

[38]蔡娟,蔡坚勇,廖晓东,等.基于卷积神经网络的手势识别初探.计算机系统应用,2015,(4):113-117.

[39] He S, Lau R W H, Liu W, et al. SuperCNN: A Superpixelwise Convolutional Neural Network for Salient Object Detection. International Journal of Computer Vision, 2015, 115(3): 330-344.

[40] Wang X, Guo R, Kambhamettu C. Deeply-Learned Feature for Age

Estimation. 2015 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA 2015: 534-541.

[41] Lu J, Liong V E, Wang G, et al. Joint Feature Learning for Face Recognition. IEEE Transactions on Information Forensics and Security, 2015, 10(7): 1371-1383.

[42] Sun Y, Wang X, Tang X. Deep Convolutional Network Cascade for Facial Point Detection. 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA ,2013: 3476-3483.

[43] Dong Z, Pei M, He Y, et al. Vehicle Type Classification Using Unsupervised Convolutional Neural Network. 2014 22nd International Conference on Pattern Recognition, Stockholm, Sweden ,2014: 172-177.

[44] 刘琮, 许维胜, 吴启迪. 时空域深度卷积神经网络及其在行为识别上的应用. 计算机科学, 2015, 42(7):245-249.

[45] 王田苗, 陶永. 我国工业机器人技术现状与产业化发展战略. 机械工程学报, 2014, 50(9):1-13.

[46] 倪自强, 王田苗, 刘达. 医疗机器人技术发展综述. 机械工程学报, 2015, (13):45-52.

[47] 梁荣健, 张涛, 王学谦. 家用服务机器人综述. 智慧健康, 2016, 2(2):1-9.

[48] 张婷. NAO 机器人在自闭症干预中的应用. 系统仿真技术, 2013, 9(4):327-331, 338.

[49] Tofighi G, Monadjemi S A, Ghasem-Aghaee N. Rapid hand posture recognition using Adaptive Histogram Template of Skin and hand edge contour. 2010 6th Iranian Conference on Machine Vision and Image Processing, Isfahan, Iran ,2010: 1-5.

[50] 陈锻生, 刘政凯. 肤色检测技术综述. 计算机学报, 2006, 29(2):194-207.

[51] Chai D, Ngan K N. Locating facial region of a head-and-shoulders color image. IEEE International Conference on Automatic Face & Gesture Recognition, Nara, Japan, Japan ,1998: 124.

[52] Fang J, Qiu G. A colour histogram based approach to human face detection. International Conference on Visual Information Engineering, Guildford, UK, 2003: 133-136.

[53] Duda R O, Hart P E, Stork D G. Pattern Classification (2nd Edition). Wiley, 2000: 55-88.

[54] Grimson W E L, Stauffer C, Romano R, et al. Using adaptive tracking to classify and monitor activities in a site. Proceedings. 1998 IEEE Computer

Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231), Santa Barbara, CA, USA, 1998: 22-29.

[55] Stauffer C, Grimson W E L. Adaptive background mixture models for real-time tracking. Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), Fort Collins, CO, USA, 1999: 1-252 Vol. 2.

[56] 袁敏, 姚恒, 刘垒. 结合三帧差分和肤色椭圆模型的动态手势分割. 光电工程, 2016, 43(6):51-56.

[57]王龙, 刘辉, 王彬, 等. 结合肤色模型和卷积神经网络的手势识别方法. 计算机工程与应用, 2017, 53(6):209-214.

[58]刘辉, 张云生, 张印辉, 等. 基于差分链码曲率的转炉火焰边界弯曲度计算. 计算机工程与应用, 2013, (7):171-175.

[59]张宏志, 张金换, 岳卉, 等. 基于 CamShift 的目标跟踪算法. 计算机工程与设计, 2006, 27(11):2012-2014.

[60]Chu H, Ye S, Guo Q, et al. Object Tracking Algorithm Based on Camshift Algorithm Combinating with Difference in Frame. IEEE International Conference on Automation and Logistics, Jinan, China, 2007: 51-55.

[61]邬大鹏, 程卫平, 于盛林. 基于帧间差分和运动估计的 Camshift 目标跟踪算法. 光电工程, 2010, 37(1):55-60.

[62]Denton E, Zaremba W, Bruna J, et al. Exploiting linear structure within convolutional networks for efficient evaluation. Eprint Arxiv, 2014:1269-1277.

[63]Gong Y, Liu L, Yang M, et al. Compressing Deep Convolutional Networks using Vector Quantization. Computer Science, 2014.

[64]Chen W, Wilson J T, Tyree S, et al. Compressing Neural Networks with the Hashing Trick. Computer Science, 2015:2285-2294.

[65]Cheng Y, Yu F X, Feris R S, et al. Fast Neural Networks with Circulant Projection.IEEE International Conference on Computer Vision. IEEE, Chile, 2015.

[66] Novikov A, Podoprikhin D, Osokin A, et al. Tensorizing Neural Networks. 2015.

[67]Kim Y D, Park E, Yoo S, et al. Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications. Computer Science, 2015, 71(2):576-584.

致　谢

转眼之间三年就过去了，我的硕士生涯就要结束了。回想起自己的整个学生时光，这个三年是最让人难以忘怀的。在湖南大学的三年里我不断成长，无论是在知识的积累，还是在为人处事的磨砺，我真的获得了很多。在这三年研究生生涯中，我体验到了和本科不一样的生活，我深深的喜欢上了实验室充实而温馨的氛围。在这里不会有人强迫我学习，但是看到大家都忙碌着，自己也不自觉地投身其中。回想到在过去的三年的学习生活中，我得到了老师和同学无数的帮助，在这里我想向帮助关心过我的老师、同学和朋友们表示真诚的谢意。

首先，我要感谢我的导师李仁发教授。李老师是一位极具人格魅力的学者，他丰富的专业知识，广阔的见解都令我折服。在这三年的研究生学习中，李老师无数次对我悉心指导，从选题到开题，从中期答辩到论文修改，都给了我许多建议。并且他还给我提供了去企业实习的机会，使我增强了自己的实践能力，也是这次实习机会让我在就业时有很多优势。值此论文完成之际，我由衷地对李老师表达深深的谢意。能够成为李老师的学生，我深感荣幸。

然后感谢杨科华、刘彦和李蕊老师。谢谢杨科华老师和刘彦老师在开题和中期检查对我进行指导，让我避免许多弯路。在我到企业实习时，李蕊老师给予我许多帮助和指导，使我在企业实习期间磨砺了实践能力。

感谢项目组的博士师兄，吴武飞和黄晶师兄都在我学习遇到困难时给了我极大的帮助。感谢我的好朋友朱立民，他在我的整个研究生生涯中给予我许多帮助。感谢李坤明师兄，他给我就业提供了很多建议。感谢陈明和石韫琛两位学弟，他们在和我一起进行 NAO 机器人相关项目时完成了许多工作，给了我许多灵感，并且帮助我完成实验。

特别感谢我的父母，他们给予了我无私的关爱和支持，让我没有经济的压力能够顺利地完成学业。

最后，谨向审阅本论文及答辩组的老师们致以诚挚的谢意。由于本人水平有限，文中难免有不足之处，敬请各位老师批评、指正。

卢兴沄

2017 年 5 月 4 日于长沙