电 子 科 技 大 学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

专业学位硕士学位论文

MASTER THESIS FOR PROFESSIONAL DEGREE

论文题目 基于深度学习的人脸检测和识别方法研究

专业学位类别 工程硕士

学 号 **20152210323**

作 者 姓 名 刘婧月

指 导 教 师 高建彬 副教授

分类号 密级

UDC 注 1

学 位 论 文

基于深度学习的人脸检测和识别方法研究

（题名和副题名）

刘婧月

（作者姓名）

指导教师 高建彬 副 教 授

电子科技大学 成 都

（姓名、职称、单位名称）

申请学位级别 硕士 专业学位类别 工程硕士

工程领域名称 电子与通信工程

提交论文日期 **2018.4.2** 论文答辩日期 **2018.5.22**

学位授予单位和日期 电子科技大学 **2018 年 6 月**

答辩委员会主席

评阅人

注 1：注明《国际十进分类法 UDC》的类号。

Research on Face Detection and Recognition Method Based on Deep Learning

A Master Thesis Submitted to
University of Electronic Science and Technology of China

Discipline: **Master of Engineering**

Author: **Liu Jingyue**

Supervisor: **Vice Prof. Gao Jianbin**

School : **School of Resources and Environment**

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它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谢意。

作者签名：[signature] 日期：2018 年 5 月 11 日

论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将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、汇编学位论文。

（保密的学位论文在解密后应遵守此规定）

作者签名：[signature] 导师签名：[signature]

日期：2018 年 5 月 11 日

摘 要

人脸检测和识别技术一直以来都是计算机视觉领域的一个重要研究方向。近几年来，随着计算机视觉领域相关技术的快速发展，特别是深度学习技术的广泛应用，人脸检测和识别获得了越来越多关注。人脸检测作为人脸识别之前的重要步骤，其检测速度和准确度能够影响到整个识别流程的性能。在约束条件下，现存的很多人脸识别算法已经能够在准确度上达到人眼识别的精度。然而在非可控条件下，由于采集设备和采集者自身的因素，人脸图像中往往存在着各种复杂干扰，用于约束条件下的人脸识别技术已经无法满足现实应用的需求。为此，设计一种对复杂干扰鲁棒的识别技术就变得相当重要。

本文针对人脸检测和识别中存在的问题进行研究，提出一个高性能的人脸检测和识别算法，主要内容如下：

1．对卷积神经网络的结构和理论基础进行总结。介绍了网络中各层的计算方法以及反向传播算法。之后对 ReLU 单元，Dropout，批度归一化等深度学习中常用的关键技术进行阐述，作为后面章节的理论基础。

2．研究基于 YOLO 目标检测模型的人脸检测算法。利用卷积神经网络对人脸目标的置信度和位置信息进行预测，并对模型中的网络结构进行改进，用深度可分离卷积代替传统的卷积单元，实现在加深网络深度的同时网络参数规模减小，在提高人脸检测速度的同时不损失检测准确度。

3．研究基于深度卷积神经网络的特征提取方法。提出针对复杂条件下的人脸图像特征提取的网络结构，用人脸验证任务作为目标对网络参数进行优化。之后用实验证明，使用该特征进行相似性度量时，比传统的人工设计特征具有更好的鲁棒性并能更好地对人脸进行表达。

4．研究基于高斯混合模型的特征匹配算法。利用高斯混合模型对人脸图像子区域进行隐式的建模，在更加细粒度上对人脸图像进行验证。高斯混合模型的训练是无监督的，并且不需要数据的标注信息，因此可以节约训练成本。之后对于相匹配的人脸子块上的特征使用联合贝叶斯算法进行相似性度量，能够在很大程度上减轻姿态，表情等高度非线性的变化对识别结果的影响，提升识别的准确度。

关键词： 人脸识别，人脸检测，深度学习，卷积神经网络，高斯混合模型

ABSTRACT

Face detection and recognition technology have been an important research direction in the field of computer vision. In recent years, with the rapid development of related technologies of computer vision, especially the extensive application of deep learning, face detection and recognition have attracted more and more attention. Face detection as a necessary step before face recognition, its detection speed and accuracy affect the performance of the entire process. Under constraint conditions, many existing face recognition algorithms have achieved the accuracy of human eye recognition. However, under non-controllable conditions, there are various complex interferences in the collected face images, the face recognition technology under constraint conditions can no longer meet the needs of the application in reality. For this reason, it is very important to design an identification technique that is robust to complex disturbances.

In this paper, a high-performance face detection and recognition algorithm is proposed. The main contents of this paper are as follows:

1. The structure and theoretical basis of the convolutional neural network is summarized. This article describes the calculation methods and back propagation algorithms in the network. Afterwards, the key technologies commonly used in deep learning such as ReLU unit, Dropout, and Batch Normalization are introduced as the theoretical basis of the following chapters.

2. Research on face detection algorithm based on YOLO detection model. The convolutional neural network is used to regress the confidence score and the position information of the face image, and the convolutional neural network structure in the model is improved. Depthwise separable convolution is used in place of the traditional convolution unit to reduce the size of parameters while deepening the depth of the network. In addition, the face detection speed is accelerated without the loss of detection accuracy.

3. Research on the feature extraction method based on convolutional neural network. A network structure for face image feature extraction under complex conditions is proposed. The face verification task is used as a target to optimize the network parameters. Afterwards, it is experimentally proved that when using this feature to measure similarity, it has better robustness than traditional artificial design features

and can better express face images.

4. Research on the feature matching algorithm based on Gaussian mixture model. The Gaussian mixture model is used to implicitly model the subregions of the face image, and the face image is verified at a finer granularity. The training of the Gaussian mixture model is unsupervised and does not require data annotation, which saves training costs. Afterwards, using the joint Bayesian algorithm to measure the similarity of features on the matching face sub-blocks, which can greatly reduce the impact of the highly nonlinear variations such as postures and expressions and improve the recognition accuracy.

Keywords: face recognition，face detection，deep learning，convolutional neural network，Gaussian Mixture Model (GMM)

目 录

第一章 绪论

1.1 研究工作的背景与意义

随着技术的不断进步，人工智能引起了越来越广泛的关注，并取得了令人瞩目的成果。人工智能也已经渗透进日常生活的各个方面，为我们提供了方便快捷的服务，如智能机器人，机器翻译、自动驾驶、虚拟现实技术等等。人脸检测与识别作为人工智能领域的一个热门方向，同时也是计算机视觉领域中的研究内容之一。近几十年来，一直引起着研究者们的广泛关注，并取得了十分丰硕的研究成果。特别是最近十多年来，随着理论基础的不断丰富以及计算机硬件性能的提升，人脸检测和识别技术得到了迅猛的发展。在公共安全和生活服务中的各中应用场景都发挥着举足轻重的作用，如公共区域的视频监控，设备登陆，刷脸支付等等。

目前，人脸识别算法在经过对齐的统一标准人脸上的识别精度已经超过了人眼的识别精度[1]。然而在非可控条件下，人脸识别和检测算法无法达到令人满意的性能。主要是因为采集图像的周围环境以及人脸自身因素等影响，使得原始的人脸图像中往往包含着各种复杂的噪声和干扰。人脸检测和识别通常存在以下几方面问题需要解决：

(1).低像素问题：由于用于图像采集的设备分辨率的不同，尤其是在视频监控应用中，采集到的图像往往分辨率较低，包含人脸的图像会丢失较多的细节信息，并会有不同程度的噪声混入到图像中，这为人脸的检测和识别工作带来了很大的挑战；

(2).遮挡问题：自然场景下，很多人会佩戴眼镜，帽子，围脖等，这会给采集到的人脸图像带来遮挡，有些遮挡物会遮挡住人脸关键位置的信息，人脸检测和识别比较难以取得非常准确的效果；

(3).姿态、表情变化：大多数人脸都不是正对着拍摄设备的，存在不同程度的姿态和表情变化，检测到的人脸会有不同程度的旋转和偏移，而这些因素会给人脸检测和识别的准确度带来很大的影响。

人脸检测是人脸识别的前置步骤，其检测结果的准确性直接影响了后续识别算法的结果。同时由于人脸图像的预处理步骤无法将上述的几种复杂干扰彻底消除，这些复杂干扰将会依次影响到人脸识别流程中的特征提取、相似性度量和分类判别阶段，从而对识别结果的准确度带来不可估量的影响。提出一种更加合理有效的应对复杂干扰的人脸检测和识别算法，是现如今极具意义也亟待解决的一

个关键性问题和发展瓶颈。

1.2 国内外研究现状

人脸检测与识别技术在社会生活的各个领域都扮演着十分重要的角色，且有着非常可观的市场前景。随着国内外科研学者不断的探索和研究，人脸检测与人脸识别技术被不断地应用和发展，并取得了许多优秀成果。

1.2.1 人脸检测的研究现状

人脸检测是指根据原始的输入图像，确定图像中人脸位置、大小和数量的过程。作为计算机视觉领域中的一个基本问题，人脸检测人脸识别和验证工作的第一步，由于检测结果的准确度直接影响了后续算法结果的准确性，人脸检测受到越来越多的重视。随着人脸检测的应用越来越广泛，用户对检测算法速度的需求越来越大，保证检测精度的同时实现实时检测是当下人脸检测算法的一个发展方向。

在 2001 年，Viola 和 Jones[2]提出了一套快速的人脸检测算法，算法的大致流程是：提取待检测图像的类 Haar 特征，之后通过 AdaBoost 方法利用提取到的特征训练分类器，最后将分类器级联得到人脸检测器。该方法较好的解决了彩色图像空间中的人脸检测问题，能够较准确且高效的完成人脸检测，直至今天仍有非常广泛的应用。之后出现了很多此方法基础上的改进。如 Wu[3]第一次将 RAB(Real AdaBoost)应用于物体检测，提出一个较成熟实用的多姿态人脸检测框架，且其中对级联结构的改进有着很不错的效果。

但是，对于自然场景下的人脸图像，Viola-Jones 的检测框架的性能显得力不从心，且很依赖经验。针对多视角下的人脸检测问题，出现了许多研究工作[4]，这些方法采用分治策略，即针对头部的不同视角和姿态，分别训练不同的检测器。然而这些方法通常比较麻烦，并导致系统的性能和准确度降低。之后的一些改进方法不再使用 Boosted-cascade，如文献[7]中，在大规模的数据集上，采用部分可形变混合模型来捕获多视角和多表情下的人脸变化。该模型能够同时完成人脸检测，表情评估和面部关键点定位等工作，但模型本身十分复杂。滑动窗口搜索方法在时间上开销较大，为了提升检测速度，文献[8]利用了图像检索技术，提出一种基于模范样本的人脸检测器。这些方法在一些比较困难的数据集上的效果比 Viola-Jones 检测器要好得多。然而由于复杂度太高（文献[7]中的方法处理一张图像大约需要 40 秒的时间），这些方法的实用性并不强。

近几年来，随着深度学习的快速发展，很多传统的检测方法被颠覆，取而代

之的是基于卷积神经网络的一系列方法。基于深度学习的人脸检测算法大致可以分为两类：基于目标检测的方法和级联法。

通用的目标检测算法近几年来发展迅速，其中的开山鼻祖是 Girshick 等人提出的 R-CNN[9]模型，该模型首次利用卷积神经网络提取的局部区域特征来进行目标检测，把检测的问题转化为分类问题，并提出边框回归用来对预测的目标窗口位置进行修正。这套算法相比传统算法有着巨大的优势，一方面能够充分利用卷积神经网络提取到鲁棒的超完备特征，另一方面得益于提出的边框回归算法，能够在粗略选中目标的基础上更加精确的描述目标位置，在 VOC2010 上获得 53.7%的 mAP，相比当时的传统方法提高了至少 10%。然而，R-CNN 模型在计算特征时存在重复计算的问题，模型需要提取几千个候选区域，每个候选区域都需要通过卷积神经网络提取一次特征，而这些区域存在着大量的重叠，这使得该模型的效率很低，计算开销大。花费在候选框的选择和提取特征的时间是 13s/张-GPU 和 53s/张-CPU。随后的改进版本 Fast R-CNN[10]针对特征的重复计算问题进行了改进，整幅待检测图像只需要通过一次卷积神经网络，就能提取到全部候选窗口的特征，极大的提升了检测速度，但是候选框的提取速度仍是整个检测任务速度的瓶颈，Fast R-CNN 在 VOC2007 上的 mAP 提高到了 68%。随后提出的 Faster R-CNN[11]框架把候选框的选择和特征特区用一个卷积神经网络实现，并对两个任务进行联合训练，极大的提升了检测速度和准确度，用 VGG 网络[12]作为特征提取网络时在 VOC2007 上的 mAP 可以达到到 73%。SSD[13]使用单一的神经网络来封装所有计算，在多分辨率的特征图上输出预测结果，实现了超过 Faster R-CNN 方法的性能，在 VOC2007 上取得 75.1%的 mAP。为了提升检测速度，2016 年提出的 YOLO[14]模型再次把目标检测任务回到回归的方法上来，把待检测图像作为输入，用卷积神经网络对目标的位置和类别置信度进行回归，在 VOC2007 上 mAP 能够达到 63.4%，而且检测的速度能够达到对视频图像的实时处理。基于深度学习的物体检测方法同样能够迁移到人脸检测的任务中来，做一些简单的改进就很容易取得比传统检测方法更好更快的性能。

另一类方法是级联法，级联法在检测领域里仍有广泛的应用。级联法的优势是能够很好的处理正、负样本的分布不平衡问题。级联结构的前几级，弱分类器能排除大部分背景等负样本，这可以为后几级的强分类器提供较少数量的输入，从而减少计算开销。随着深度学习的广泛应用，基于卷积神经网络的多级检测结构开始涌现出来。第一级网络用来产生候选区域，之后的几级网络用来完成检测工作。如 Cascaded CNN[15]，是对经典的 Viola-Jones 方法的深度卷积网络实现。检测流程中其中包含多个分类器，每个分类器都是一个小型的卷积神经网络，卷积

神经网络的复杂度逐级递增。Cascaded CNN 仍采用滑动窗口提取检测区域，为了减少计算开销，前面的神经网络结构较简单，用来拒绝大量的非人脸区域，后面的网络结构较复杂，用来进一步区分难以分类的区域，从而得到更加准确的结果。MTCNN[17]同样采用了三阶级联，该算法有三个阶段组成：第一阶段，利用浅层卷积神经网络快速的生成候选区域；第二阶段，通过较复杂的卷积神经网络精炼候选区域，丢弃大量的重叠区域；第三阶段，使用更加复杂的卷积神经网络，实现候选区域的判定，同时预测出五个面部关键点的位置信息。然而这些方法大多都没有进行联合训练，而是利用贪婪算法进行优化，导致这些方法忽略了卷积神经网络可以被联合优化的特性。此外，由于对多个卷积神经网络进行训练，这些方法的训练开销往往较大。

1.2.2 人脸识别的研究现状

图 1-1 姿态表情变化下的人脸图像

人脸识别是基于面部特征来进行身份识别的技术，即给定一张人脸图像，判断其身份的过程。人脸图像中通常包含光线、姿态和表情变化等的类内差异和身份信息变化的类间差异，这些差异是高度非线性的，无法用传统的线性模型进行分类，因此如何有效的区分这两种差异是人脸识别中最大的挑战。在之前的研究中，Turk[18]和 Belhumeur[19]已经证实，可以通过把正面人脸投影到一个对表情和光照等变化具有不变性的低维子空间的方法，降低表情和光照对识别的影响。很多算法在对齐的人脸图像上都能取得很好的识别效果，相比之下，姿势变化就成为无约束条件下人脸识别的一个巨大的障碍，也是影响稳健的人脸识别的各项视觉干扰因素中最具挑战性的因素[20]。这是由于，姿态变化能够造成产生巨大的差异性，即使是同一个人在不同的姿势下往往也会看起来有很大的不同。在如图 1-1 所示的四张属于同一个人的人脸图像中，姿态和表情的变化增加了识别的难度，即使是人用肉眼来进行辨认，也十分容易出错，认为是不同的人。因此，对姿势变化的处理就显得格外重要。同时，解决姿势变化问题也可以在很大程度上帮助

减轻其他视觉变化带来的不利影响。

现存处理人脸姿态，表情变化的方法可以大致分为两类：人脸对齐和提取姿势不变性特征。

第一类方法可分为基于 2D 图像的方法和基于 3D 模型的方法，在这些方法中，通过在全局范围内对齐人脸以减少姿势变化的影响。一类基于 2D 图像的方法是基于图像像素或子块的空间映射学习，来模拟三维空间中人脸姿势的几何变换。例如 Arashloo 和 Ashraf[21]等人通过用图模型来推断正面姿态的人脸。尽管在这些方法中，任意姿态的人脸图像像素经过重新排列都可以得到近似正面的人脸图像，但是人脸的形状和一致性无法很好地保留。另一类 2D 方法是学习不同姿势外观之间的变换关系，如局部线性回归（Local Linear Regression）[23]，但在这些方法中用线性模型来近似高度非线性的姿势变换，故很难保留人脸图像的个体信息。同时，随着深度学习方法的广泛应用，Zhu[24]等人把深度学习模型应用到重建正面人脸问题中来，提出一种深度限制波尔兹曼机（RBM）模型。对姿态和光照变化影响下的人脸图像，使用深度限制波尔兹曼机提取特征，该特征能够保留人脸的身份信息，之后用全连接层对该特征进行非线性回归，重建出正面的人脸图像。

由于人脸姿势和表情的变化是一种三维非刚性变换，基于 3D 模型的方法越来越多的被应用于人脸对齐。这类方法旨在用 3D 模型估计深度信息，并通过 3D 几何变换恢复正面姿态：首先建立 2D 人脸图像与 3D 面部模型的对应关系，然后旋转 3D 模型来渲染正面人脸视图。一种具有代表性的方法是 3D 形变模型（3DMM）[25]，使用主成分分析法(PCA)来构建人脸图像的 3D 纹理和形状模型，通过最小化 3D 形状模型和原始 2D 人脸图像之间差异来估计模型参数。虽然已经提出十几年，3D 形变模型仍能获得有竞争力的效果。然而，这种方法处理图像的时间成本较高，大约每分钟仅能处理一幅图像。其中一种替代方法是，通过使用 2D 图像和 3D 模型上关键点之间的对应关系来估计模型参数。

Zhu[26]等人提出一种高保真的姿势和表情归一化方法。通过 3D 变换，得到保留了身份信息的归一化图像。该方法能够极大的降低人脸图像中的信息损失，并能够使重建的人脸更加逼近真实人脸。Hassner[27]等人采用了不同于以往的为不同 2D 图像分别建立 3D 模型的方法，对所有的 2D 人脸图像均使用一个统一的、不变的 3D 模型，忽略面部形状上的个体差异。该方法得到的归一化结果，仍可以很好的保留个体之间的差异信息，且能得到比较正面的人脸。

然而，此类方法也存在很多缺陷，一方面捕获 3D 数据增加了额外的成本和资源负担，另一方面此类方法在很大程度上依赖于面部特征点的精确定位，而且即使是同一个人，不同姿势下的图像经过对齐后并不能保证得到的正面图像完全一

样。

第二类方法是姿势鲁棒的特征表达。传统的人脸特征提取方法，大多是针对某些特定的干扰因素而设计的，此类特征很有针对性，对某些特定的干扰也很鲁棒。例如， Guillaumin[28]等人采用 SIFT（Scale-Invariant Feature Trans-form）特征提取人脸图像九个关键点位置的特征来描述人脸。Zhang[29]等人提出了 LGBPHS（Local Gabor binary Pattern Histogram Sequence）特征描述子，该特征融合了 Gabor 和 LBP 两种特征的优点。为了更完备的描述人脸的特征，越来越多的算法采用局部描述符的高维度连接。Zhang[30]等人在前面的研究中证实，高维联合特征能够提升人脸识别的准确度。Chen[31]等人在面部关键点周围的区域密集的提取 SIFT 特征，并把这些特征联结起来，通过用高维度的特征表达人脸实现姿态不变性。Wright[20]等人通过构建人脸图像的多尺度图像集，并把这些图像集中的图像分割成密集重叠的小块，实现姿态鲁棒的特征提取。该方法在大的姿势变化下仍能获得不错的识别能力，并能得到很好的泛化性能。

过去几十年中，手工设计的特征描述子在计算机视觉领域的各种应用里广泛使用。然而，人工设计的人脸描述算子主要依赖设计者的先验知识，需要很多工程经验和不断的尝试。此外，人工设计的人脸描述符都是浅层的模型，它将高维图像空间分成若干局部区域，提供的是局部表达，因而不能更好的表达人脸图像语义上下文的全局特征。

随着神经网络的模型设计和训练方法的进步、GPU 等硬件性能的发展以及大规模训练数据的出现，深度学习在视觉目标识别中的应用越来越广泛。深度学习方法可以从大量的训练数据中自动学习特征，该特征能够很好的表达全局和上下文的信息，相比传统的特征其表达能力更强。此外，特征提取和分类器的训练能够进行联合优化，效率更高的同时能够使两种任务协作完成。近几年来，卷积神经网络发展迅速，各种用于特征提取和分类的网络结构和运算单元的提出层出不穷。2012 年 ImageNet 图像分类比赛中，获得冠军的小组采用了一种 5 层卷积层的深度网络来完成分类任务[32]，提取到的特征对各种干扰具有鲁棒性，排名第 2 的小组采用了传统的手工特征，该方法的准确率超出第二名 10%以上。谷歌[33]提出一种 22 个卷积层的深度卷积神经网络，对网络的结构进行改进，加深网络的深度的同时扩大网络每一层的规模，但参数数量相比传统卷积层的连接方式大大减少。为了实现对网络前几层的误差进行控制，将监督信号加入到网络中的多个中间层，进一步把错误率降到了 6.656%。Sun[34]等人提出一种用于图像特征提取的卷积神经网络，在多尺度多视角的人脸图像子块上提取 160 维的低维特征，之后把这些低维特征级联在一起用于分类。通过在网络中加入正则化约束使得网络稀疏，从

而对于较小的训练数据子集也不会出现过拟合现象，具有很强的泛化能力。该特征能够很好的表达人脸图像的身份信息，并能够与任何分类器结合。通过人脸识别任务学习得到的人脸特征包含较多的类间变化，而未能很好的利用同一个人因为光照、表情以及姿态等产生的类内变化。所以他们随后提出了 DeepID2[35]，同时把人脸识别误差和验证误差加入到损失函数中，同时对网络结构以及训练数据进行了扩增，从而得到能够使类内变化最小、类间变化最大的特征，其识别结果在 LFW 数据集上取得了 99.47%的正确率。后续提出的 DeepID2+[36]在前面的基础上对网络结构进行了改进，加大网络宽度并把浅层特征进行全连接加入到最后得到的特征中。同时验证了 CNN 提取的特征是稀疏、鲁棒并且有选择性的。文献[37]在前面文章的稀疏特性基础上，进一步改进网络结构，并提出一种使网络中权值达到稀疏的训练策略。把训练好的网络权重作为初始值，之后根据一定的规则对网络中的连接权重进行剪枝，然后重新训练网络。该方法使网络中连接数大大减少，在获得稀疏的网络同时，得到较好的性能。

Facebook[38]把 3D 人脸对齐方法和深度学习结合在一起。在原始图像上分别进行 2D 对齐和 3D 对齐，得到近似正面的人脸图像，之后利用卷积神经网络进行特征提取。该网络最终可得到非常紧凑且保留更多差异性的特征表达。该方法较之前最好的识别系统提升了 27%的准确度，在 LWF 数据集上可以得到 97.35%的识别率，几乎达到了人类的识别精度。

为了得到更好的识别准确度，把网络结构做大做深是识别领域的一个发展趋势。然而由于误差反向传播过程中，越靠近浅层的误差梯度值越容易弥散，这使得更深的网络结构优化起来会更加困难，对硬件的并行计算以及存储能力的要求也越高。与此同时，更大的网络中包含的参数也更多，这需要海量带标签的训练数据来避免过拟合。

由于姿势变化下的人脸有很多区域不可见，在进行正面人脸重建的过程中需要对这部分进行填充，这会引入较多的不确定信息和干扰。另一方面，重建结果并不能做到和原始人脸图像完全一致，所以不能很好的保留一些身份和细节信息，这会对识别结果产生一定的影响。相比之下，提取对复杂干扰鲁棒的特征，能够在充分利用原图像的像素级信息的同时又不引入新的干扰信息，从而对于解决人脸识别中的复杂干扰问题有一定的优势。

1.3 本文的主要工作

本论文分别针对检测阶段和识别阶段的算法进行改进，以人脸识别中的特征提取和特征匹配算法以及识别之前需要进行的人脸检测作为重点研究内容，本文

改进的算法能够很好的应用在人脸识别系统中。主要工作内容如下：

1、在现有的基于深度学习的检测流程中对卷积神经网络的结构进行改进，使之在网络结构加深的同时，参数数量和网络规模变小。这样的网络结构能够实现提升检测速度的同时，不损失检测结果的准确度，其速度能够达到实时检测。

2、提出了一种把卷积神经网络与高斯混合模型进行结合的方法流程，把深度网络提取的特征用于高斯混合模型的训练，能够更好的去除掉无约束条件下人脸图像中的各种复杂干扰，使识别系统对这些干扰鲁棒。

3、用人脸验证作为目标对提取特征的卷积神经网络进行训练，为了与高斯混合模型进行结合，对网络的结构进行改变，在不同的池化层上输出不同尺度的特征，提取到的特征对复杂干扰更加鲁棒。

1.4 本论文的结构安排

本文针对人脸检测和识别任务进行了分析和研究。首先简要的介绍了人脸检测和识别问题的发展现状，以及卷积神经网络在人脸检测和识别中的应用。之后简单介绍了深度学习的算法基础以及常用技术，后三章分别对人脸识别系统中三个重要环节的算法框架进行分析和介绍，并通过实验验证算法的性能。本文的结构安排如下：

第一章为绪论，首先是人脸检测和识别问题的研究背景与意义，之后重点介绍了人脸检测和人脸识别技术的研究现状与发展方向。随着深度学习的发展，人脸检测和识别逐渐由传统的计算机视觉方法，转变到基于深度学习的方法。本章的结尾列出了本文的主要工作内容以及组织结构。

第二章作为后面三章的理论基础，主要介绍了卷积神经网络的相关理论基础以及常用的技术。首先是对传统的神经网络的理论知识和优化过程进行介绍，由此引出卷积神经网络的相关理论基础，包括网络构成以及各层的数学定义，最后介绍了几种卷积神经网络中常用的技术。

第三章介绍了本文人脸检测的算法流程。首先对传统的 Viola-Jones 人脸检测器的原理和方法进行介绍，之后介绍一种基于卷积神经网络的目标检测模型——YOLO 目标检测器，该模型能够实现快速实时的检测。我们把该检测结构迁移到人脸检测任务上来，出于对检测器的实时性和准确性的权衡，对其中的网络结构进行改进。之后在 Celeb A 数据集上对改进的网络结果进行评估。

第四章介绍了利用卷积神经网络进行人脸特征提取的过程。首先介绍了几种传统的人脸图像特征描述符，之后重点介绍本文中用于特征提取的卷积神经网络结构，以及其训练过程。为了得到作为第五章中高斯混合模型输入的特征，对网

络结构进行设计。最后对本文提出的网络在 LFW 数据集上进行测试。

第五章阐述了人脸识别流程中的特征匹配阶段。首先介绍了几种常用的特征相似性度量方法，之后介绍了本文的特征匹配算法流程。使用第四章卷积神经网络提取的特征来构建图像的高斯混合模型，并详细介绍了高斯混合模型的训练和测试，最后在 LFW 数据集上对本文提出的识别算法进行验证，并与其他方法进行比较。

第六章是总结与展望，对本文的相关工作和贡献点进行总结，并对未来人脸检测和识别的研究和发展方向进行了展望。

第二章 卷积神经网络的基本理论

随着深度学习的迅猛发展，深度学习的方法开始广泛的被应用在计算机视觉领域的各项研究工作中。卷积神经网络是其用于视觉领域的代表性方法，早在 1998 年 Yann Lecun[39]提出了于手写体数字的识别的 LeNet5，该网络包含了卷积神经网络中的基本模块，作为之后深度网络模型的基础，该方法一直被广泛的应用在手写体识别问题中。最近几年，卷积神经网络在多个领域内均有突破性进展，在目标检测、运动分析、语音识别、物体识别、自然语言处理等方面呈现出了比以往方法更优异的结果。本章将从工作原理，网络结构，前向传播和反向传播方法等方面对卷积神经网络进行详细的介绍。

2.1 神经网络的理论基础

神经网络由神经元的相互连接构成，网络中的每个神经元代表一个输出函数，每两个神经元之间的连接表示通过该连接信号的权重。神经网络算法能够提供一种复杂且非线性的模型假设，通过有监督的学习方式来学习网络中的参数，用此参数来拟合训练的数据。网络的输出值由网络的连接方式、权重以及激活函数所共同确定。

2.1.1 前馈神经网络的结构

神经元是神经网络中的基本单元，神经元的结构示意图如图 2-1。每个神经元与其他神经元相连，当它处于激活状态时，就会产生一个输出值，传递给下

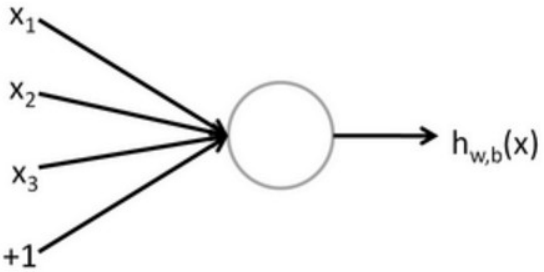


图 2-1 神经元示意图

一个神经元。每个神经元中存储了一个阈值，如果输入的线性组合超过这个阈值，则该神经元就会被激活。图 2-1 中，假设左侧输入信号为(x_1, x_2, x_3)，经过加权线性组合后传递给神经元，神经元接收到的输入值与阈值进行比较之后，通过 “激活函数”进行非线性运算，运算结果作为该神经元的输出值。具体如公式(2-1)所

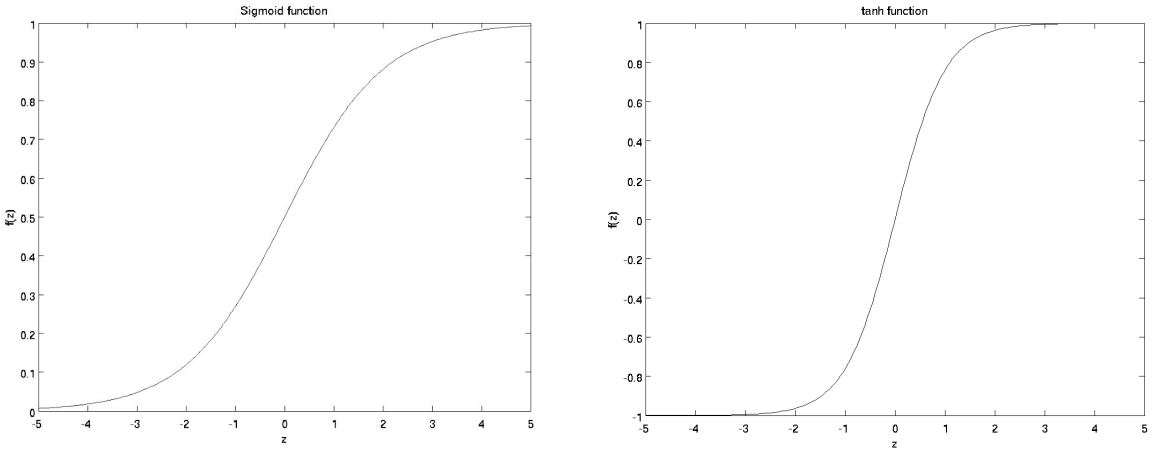
示：

$$h_{w,b}(x) = f(W^T x) = f(\sum_{i=1}^{n} W_i x_i + b) \tag{2-1}$$

式(2-1)中， x 表示该神经元的输入向量，W 表示权重向量，b 为偏置，函数 $f: R \to R$ 为激活函数，表示该神经元对输入的响应程度，其作用是对线性运算 $W^T x$ 进行一个非线性的操作。常用的激活函数为 Sigmoid 和 Tanh 函数，如图 2-2 所示。Sigmoid 的表达式为公式(2-2)，它可以把在较大范围内的输入值压缩到(0,1)的输出值范围内；Tanh 函数表达式为公式（2-3），它把输入值压缩到(-1,1)的输出范围内。

$$f = \frac{1}{1+e^{-x}} \tag{2-2}$$

$$f = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2-3}$$



(a) (b)

图 2-2 两种激活函数的函数图

(a)Sigmoid 函数；(b)Tanh 函数

神经网络为层级结构，每一层包含多个神经元节点，这些节点之间相互连接，如图 2-3 所示。神经网络通常必须包含一个输入层和一个输出层，中间允许有一个或者多个隐藏层，前一层神经元的输出作为后一层的输入。相邻两层之间连接的方向是单一的，同层和跨层的神经元之间不进行连接。在神经网络的训练阶段，就是根据训练数据的分布来学习网络中的连接权重和每个神经元中的阈值。

图 2-3 为一个简单的 3 层前馈神经网络模型。其中，网络最左边的一层 LayerL_1 为输入层，最右边的一层 LayerL_3 为输出层。LayerL_2 为隐藏层，其作用是把接收到的上一层的输出值，通过该层的计算转变为下一层的输入值。

实际应用中，网络中的层数以及每层中包含的神经元个数可以根据具体的应用场景来进行确定。但当神经元个数较多时，网络结构会变复杂，参数随之增多，其对训练数据的表达功能也会增强，训练的速度也相对较缓慢。

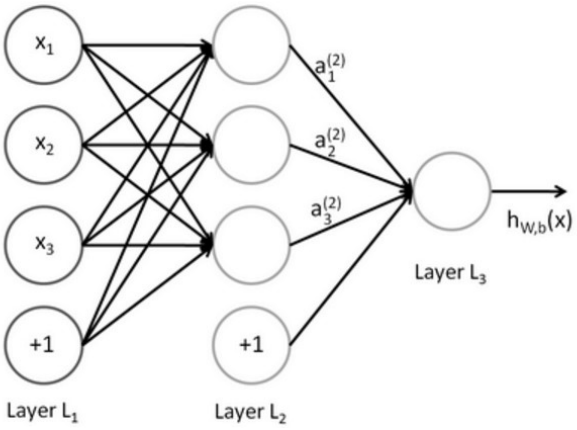


图 2-3 神经网络模型

2.1.2 神经网络的反向传播

1986 年，鲁梅尔哈特(Rumelhart)、欣顿(Hinton)和威廉姆斯(Williams)[40]等人提出了用于神经网络训练的误差反向传播算法(error BackPropagation, 简称 BP)，并发表在《自然》杂志上。BP 算法直到今天仍广泛应用于现实的神经网络训练任务中，该算法不仅可以用于训练多层前馈神经网络，还可以对其他类型的神经网络进行优化求解，如递归神经网络。BP 算法的主要思想是将网络输出层计算得到的误差，反向的传回到网络的前面各层，通过梯度下降法循环迭代前面各层的网络参数，直到得到最终使得误差最小的模型参数。

给定含有 m 个训练样本的训练集 D={$(x^{(1)}, y^{(1)}),(x^{(2)}, y^{(2)}),...,(x^{(m)}, y^{(m)})$}，其中 $x \in R^d, y \in R^1$。对于单个样本 $(x^{(i)}, y^{(i)})$，其代价函数如式(2-4)所示：

$$J(W,b;x,y)=\frac{1}{2}\| h_{w,b}(x)-y \|^2 \tag{2-4}$$

式(2-4)是一个平方误差函数。对于一个含有 m 个样本的训练集，整体的代价函数如式(2-5)：

$$\begin{aligned} J(W,b;x,y) &= \left[\frac{1}{m}\sum_{i=1}^{m} J(W,b;x^{(i)},y^{(i)})\right]+\frac{\lambda}{2}\sum_{l=1}^{n_{l-1}}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji}^{(l)})^2 \\ &= \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\| h_{w,b}(x^{(i)})-y^{(i)} \|^2\right)\right]+\frac{\lambda}{2}\sum_{l=1}^{n_{l-1}}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji}^{(l)})^2 \end{aligned} \tag{2-5}$$

上面的表达式由两部分组成，第一项是均方误差项，第二项是正则化项，用来惩罚网络中数值较大的权重，防止网络过拟合。λ 是正则项系数，用来调节损失

函数中前后两项的相对重要性。

为了对神经网络进行求解，需要对网络中的参数 W 和 b 进行初始化，一般初始化为一个接近零的随机值，如使用高斯分布来生成随机值。BP 算法基于梯度下降策略，以目标的负梯度方向对参数进行调整，使得整体的代价函数$J(W,b)$最小。每一次迭代都按照公式(2-6)和(2-7)对参数 W 和 b 进行更新：

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W,b) \tag{2-6}$$

$$b_{i}^{(l)} = b_{i}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W,b) \tag{2-7}$$

其中，第l层隐藏层的参数为$W_{ij}^{(l)}$和$b_{i}^{(l)}$，i表示l层的第i个参数，j表示l-1层中的第j个神经元；$\alpha \in (0,1)$是学习率，控制着算法每一步迭代的更新步长，若太大则容易发生震荡，太小则降低收敛速度；根据梯度下降算法，等式右边的第二项是损失函数分别关于权值$W_{ij}^{(l)}$和偏置$b_{i}^{(l)}$的偏导数，由公式(2-8)和(2-9)计算得到：

$$\frac{\partial}{\partial W_{ji}^{(l)}} J(W,b) = \left[\frac{1}{m} \sum_{i=1}^{m} \frac{\partial}{\partial W_{ji}^{(l)}} J(W,b;x^{(i)},y^{(i)}) \right] + \lambda W_{ji}^{(l)} \tag{2-8}$$

$$\frac{\partial}{\partial b_{i}^{(l)}} J(W,b) = \frac{1}{m} \sum_{i=1}^{m} \frac{\partial}{\partial b_{i}^{(l)}} J(W,b;x^{(i)},y^{(i)}) \tag{2-9}$$

由于损失函数中的正则化项只对权值进行限制，而不作用于偏置项，所以公式比公式多了后面的一项偏导数。

BP 算法执行如下操作：首先将训练样本提供给输入层的神经元，之后将运算结果逐层向前传播，直到输出层产生最终的输出结果；之后计算输出层的误差，将误差反向传播回前面几层；最后根据各个隐层神经元的误差，调整网络中的权重和阈值。该迭代过程循环进行，直到达到迭代的停止条件，如训练误差达到一个很小的值。

具体流程如下：

(1).前向传播，根据当前网络中的参数计算输入样本在各层节点的输出值：

$$a_{i}^{(l)} = h_{w,b}(x) = f(z_{i}^{(l)}) \tag{2-10}$$

其中，$a_{i}^{(l)}$表示第l层的第i个节点的输出值；

(2).计算输出层神经元的梯度项，如公式(2-11)所示：

$$\begin{aligned} \delta_{i}^{(n_l)} &= \frac{\partial}{\partial z_{i}^{nl}} J(W,b;x,y) = \frac{\partial}{\partial z_{i}^{nl}} \frac{1}{2} \| h_{w,b}(x) - y \|^2 \\ &= \frac{\partial}{\partial z_{i}^{nl}} \frac{1}{2} \sum_{j=1}^{s_{nj}} (a_{j}^{n_l} - y_j)^2 = \frac{\partial}{\partial z_{i}^{nl}} \frac{1}{2} \sum_{j=1}^{s_{nj}} (f(z_{i}^{(nl)}) - y_j)^2 \\ &= -(f(z_{i}^{(nl)}) - y_j) \cdot f'(z_{i}^{(nl)}) = -(a_{j}^{(n_l)} - y_j) \cdot f'(z_{i}^{(nl)}) \end{aligned} \tag{2-11}$$

(3).计算完输出层的残差后，将误差逐层向前传播，利用链式法则计算各个隐层神经元的梯度项，如式(2-12)所示：

$$\delta_i^{(l)} = (\sum_{j=1}^{s_{nj}} W_{ji}^{(l)} \delta_j^{(l+1)}) f'(z_i^{(l)}) \tag{2-12}$$

式中，$l = n_l - 1, n_l - 2, ..., 2$，将式中 nl 替换为 $l+1$，则各层的残差计算公式如(2-13)所示：

$$\delta_i^{(l)} = (\sum_{j=1}^{s_{nj}} W_{ji}^{(l)} \delta_j^{(l+1)}) f'(z_i^{(l)}) \tag{2-13}$$

权值 W 和偏置 b 的偏导数计算公式为式(2-14)和(2-15)：

$$\frac{\partial}{\partial W_{ji}^{(l)}} J(W, b; x, y) = a_j^{(l)} \delta_i^{(l+1)} \tag{2-14}$$

$$\frac{\partial}{\partial b_i^{(l)}} J(W, b; x, y) = \delta_i^{(l+1)} \tag{2-15}$$

中间层各个节点的残差计算公式可统一表示为式(2-16)的形式：

$$\delta^{(l)} = (W^{(l)})^T \delta^{(l+1)} f'(z^{(l)}) \tag{2-16}$$

(4).计算权值 W 和偏置 b 的偏导数，即更新的差值，如(2-17)和(2-18)所示：

$$\nabla_{w^{(l)}} J(W, b; x, y) = \delta^{(l+1)} (a^{(l)})^T \tag{2-17}$$

$$\nabla_{b^{(l)}} J(W, b; x, y) = \delta^{(l+1)} \tag{2-18}$$

2.2 卷积神经网络的理论基础

2015 年，Yann Lecun 在《自然》上发表了介绍深度学习的综述性文章[41]，对卷积神经网络进行了详细的介绍。卷积神经网络与传统的神经网络结构类似，都是由多个运算单元堆叠在一起的层级结构。不同的是，卷积神经网络中的每个运算单元由卷积层加上一个下采样层组成，其内部结构如图 2-4 所示。网络可以学习到图像特征的分层表达，最底层可以学习到原始图像中局部的边缘和纹理信息；中层可以学习到描述不同类型目标的特征；最高层学习到的是描述整个图像的全局特征。卷积神经网络的卷积层的输出包含多个特征图(featureMap)，特征图上的每个像素表示一个神经元。同一个特征图上的神经元共享同一个卷积核的权值，权值共享好处是减少网络需要学习的参数数量，降低模型复杂度的同时避免过拟合。下采样层也叫池化（pooling）层，可以将语义相似的特征融合为一个。该层在降低特征维度的同时，能够保证新的特征层表示不敏感于前一层元素在位置和表现上的变化，从而实现对平移和形变的鲁棒性。

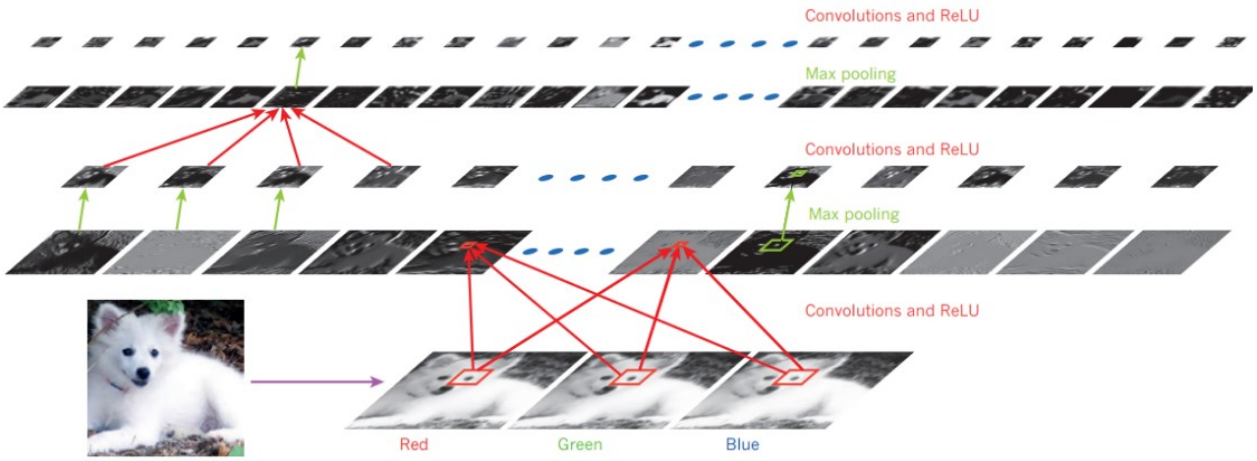


图 2-4 卷积神经网络内部结构[41]

2.2.1 卷积神经网络的模型定义

一个卷积神经网络通常由输入层，卷积层，下采样层，全连接层和输出层组成。网络的输入是一个二维图像，经过每种连接层的映射成为更加抽象的特征。

2.2.1.1 卷积层(Convolutional Layer)

卷积层的输入为上一层的输出，对输入图像中的每个局部感受野中的像素值进行卷积计算，构成输入图像的特征图(Feature map)，特征图上的神经元保持原有图像中像素点之间的相对位置。

卷积层可以用公式(2-19)表示：

$$y^{j(r)} = f\left(b^{j(r)} + \sum_{i} k^{ij(r)} * x^{i(r)}\right) \tag{2-19}$$

其中，x^i 和 y^j 分别表示第 i 个输入特征图和第 j 个输出特征图，k^{ij} 是该层的卷积核，激活函数 $f(\cdot)$ 可以为 sigmoid 函数，修正线性单元（RELU）等。每个卷积核大小为 N×N，N 通常取值为奇数，卷积核中的参数就是卷积层要学习的内容，包括权值 W 和偏置项 b。通常一个卷积层的卷积核有多个，这些卷积核具有相同的大小并按照通道排列在一起。

卷积核重复作用于整个输入图像，保证卷积核中的参数是共享的。相比全连接方式，这样做的好处是能够极大地减少需要学习的参数数量，降低网络复杂度并避免过拟合；同时，每个卷积核负责检测某一种类型的特征，如边、角、曲线等，权值共享可以在输入图像中识别某种特征，而不考虑该特征在输入特征图中的位置。

2.2.1.2 池化层(Pooling Layer)

池化层的目的是为了减少特征图的维度(尺寸)，并保持特征图的深度(通道数目)大小不变。常见的池化规模为 2×2，步长为 2。常用的池化操作有平均池化和最大池化。平均池化的输出值是 2×2 模板中四个点上的均值，最大池化的输出值是四个点中的最大值。

下采样层的计算可以表示为式(2-20)：

$$y_{j,k}^{i} = f(x_{j\cdot s+m,k\cdot s+n}^{i}) \tag{2-20}$$

其中 $f(\cdot)$ 表示为最大池化或平均池化等。池化层是一种非线性的下采样方法。池化层可以使网络模型具有平移、缩放的不变性，使提取到的特征有更强的鲁棒性，同时降低特征维度，有效减小网络中的计算量。

2.2.1.3 全连接层(Full-Connected Layer)

全连接层位于卷积神经网络的输出层之前，一般作为网络结构的后几层。全连接层的连接方式与传统的神经网络神经元的连接方式相同，相邻两层的神经元之间两两相连。

全连接层一般用公式(2-21)表示：

$$y_j = f\left(\sum_i x_i \cdot \omega_{i,j} + b_j\right) \tag{2-21}$$

其中，x_i 和 y_j 分别代表该层的输入和输出，$\omega_{i,j}$ 和 b_j 表示权重和偏移量，激活函数 $f(\cdot)$ 可以为 sigmoid 函数，修正线性单元（RELU）等。

全连接层的权值不共享，因此该层的参数数量往往占了整个卷积神经网络中总参数数量的大部分，所以该层中很容易出现过拟合。解决过拟合的方法通常是在损失函数中加入正则项，此外，文献[32]中提出了一种避免过拟合的 Dropout 策略，该方法将在 2.2.4 节做详细介绍。

2.2.2 ReLU 激活函数

Sigmoid 和 Tanh 等传统的激活函数存在一个致命缺点：当激活函数的输入值非常大或者非常小的时候，激活函数值处于饱和区，此时神经元的梯度接近于 0。这会导致在反向传播阶段出现梯度消失的情况，参数的更新值变得非常小，训练过程中网络的收敛速度很慢，训练的时间成本增多。此外，使用 sigmoid 和 tanh 作为激活函数时，由于激活函数的计算较复杂，网络前向传播和反向传播阶段的计算量较大。因此，它们很少在卷积神经网络中使用。

针对上述问题提出了 ReLU 激活函数[42]，ReLU 函数能够为网络引入稀疏性，

其前向运算和反向运算不需要计算指数运算，极大的减少了计算量。同时降低参数之间互相依存的关系，能够在一定程度上减轻网络的过拟合现象。ReLU 的表达式如式(2-22)所示：

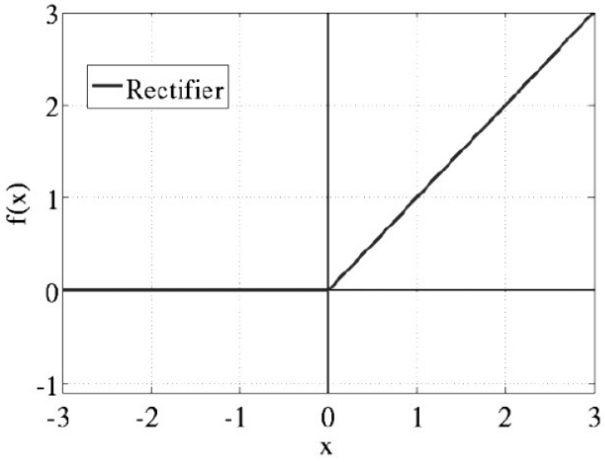


图 2-5 ReLU 函数模型

$$f(x)=\max(0,x) \tag{2-22}$$

反向传播中导数的计算公式为式(2-23)：

$$f'(x)=\begin{cases}1 & (x>=0)\\ 0 & (x<0)\end{cases} \tag{2-23}$$

其函数模型如图 2-5 所示，由于 ReLU 是不饱和函数，不会造成梯度弥散现象，因此可以使网络的收敛速度加快，减少网络的训练周期。文献[32]中通过 ReLU 和 Tanh 激活函数的对比实验，证实了 ReLU 函数能够加速网络的收敛，缩短训练网络的时间。如图 2-6 所示，在 CIFAR 数据集[43]上对网络进行训练，对使用 ReLU 和 Tanh 函数的网络训练速度进行比较。网络的训练误差率降至 25%时，使用 ReLU（实线）比使用 Tanh（虚线）的学习速度提升了大概 6 倍。

但是 ReLU 在训练时比较“脆弱”，实际的网络训练中，如果学习率设置的比较大，很可能网络中很多神经元的梯度永远是 0。为了解决这个问题提出了 Leaky ReLU[44]，表达式如式(2-24)所示：

$$f(x)=\begin{cases}x & (x>=0)\\ \alpha x & (x<0)\end{cases} \tag{2-24}$$

这里α是一个很小的常数，这样做的好处是，能够保留一些负轴的值，使得小于 0 的输入信息不会全部丢失。

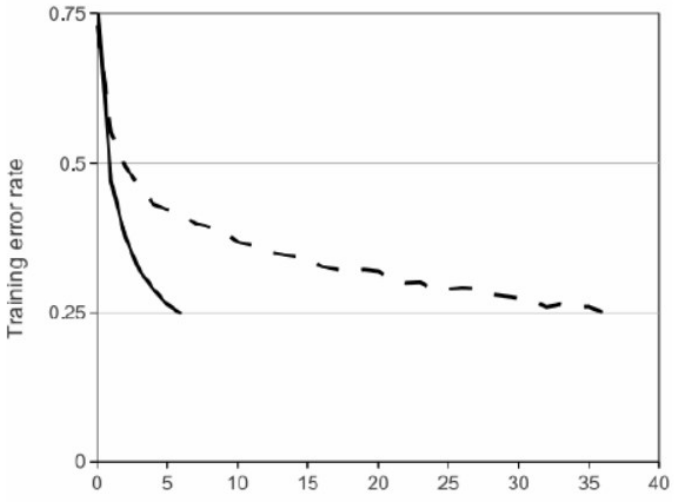


图 2-6 ReLU（实线）与 Tanh（虚线）激活函数训练速度对比[32]

2.2.3 批度归一化（Batch Normalization）

除了激活函数会影响到卷积神经网络的收敛速度之外，网络输入数据的分布也会对网络的收敛速度造成影响。这是因为输入的整体分布如果逐渐向非线性函数取值区间的上下限两端靠近,会导致反向传播时底层神经网络的梯度消失。在卷积神经网络训练之前，对输入数据进行一些预处理操作，能够在一定程度上把输入数据的分布限制在均值为 0 方差为 1 的标准正态分布，从而加速网络的收敛速度。常见的图像预处理操作有图像去均值，归一化以及白化预处理等。图像的相邻像素间往往存在较强的相关性，使得训练数据存在冗余。白化过程可以使得模型的输入数据具有以下性质：(i)特征之间相关性较低；(ii)所有特征具有相同的方差。然而白化预处理的计算量比较大，且白化不是处处可微的，在深度学习中，白化处理很少被用到。

文献[45]提出一种数据的预处理方法——批度归一化(Batch Normalization，以下简称 BN)，用来对深度网络中的每层（主要是卷积层）的输出数据进行处理，即在激活函数前面加入一层 BN 层，用来对激活函数的输入值作规范化处理，使得该层输出的特征在各个维度上的均值为 0、方差为 1。

对于一个批次中的 m 个训练样本，其具体的 BN 操作是对隐层内每个神经元进行如式(2-25)的变换，设输入为一个 d 维的向量$x=(x^{(1)},x^{(2)},...,x^{(d)})$：

$$\hat{x}^{(k)}=\frac{\hat{x}-E[x^{(k)}]}{\sqrt{Var[x^{(k)}]}} \tag{2-25}$$

其中，k 表示的是第 k 个维度，$E[x^{(k)}]$和$\sqrt{Var[x^{(k)}]}$分别表示第 k 维度上的均值和方差。但是这样会导致原始输入数据的分布被破坏，使得网络对数据的表达能力下降，为了避免这一点，BN 层的每个神经元增加两个可学习的参数$\gamma^{(k)}$和$\beta^{(k)}$，用来对变换后的输出进行反变换，这样网络就能够学习到原始数据的特征分布式(2-26)：

$$y^{(k)}=\gamma^{(k)}\hat{x}^{(k)}+\beta^{(k)} \tag{2-26}$$

BN 层具体的前向传播过程计算公式如式(2-27)-(2-30)所示：

$$\mu_{\mathrm{B}}\leftarrow\frac{1}{m}\sum_{i=1}^{m}x_i \tag{2-27}$$

$$\sigma_{\mathrm{B}}^2\leftarrow\frac{1}{m}\sum_{i=1}^{m}(x_i-\mu_{\mathrm{B}})^2 \tag{2-28}$$

$$\hat{x}_i\leftarrow\frac{x_i-\mu_{\mathrm{B}}}{\sqrt{\sigma_{\mathrm{B}}^2+\varepsilon}} \tag{2-29}$$

$$y_i\leftarrow\gamma x_i+\beta \tag{2-30}$$

其中，m 为一个批次中样本的个数，μ_{B}和σ_{B}^2分别为该批次样本所有维度上的均值和标准差，$\hat{x}_i$和y_i为分别为归一化后的输入向量和 BN 层恢复出的特征。

训练阶段结束之后，网络中的参数值都是固定的，上述式子中的参数γ和β已经通过学习得到。BN 层的均值为所有批次训练样本上的均值，标准差为每个批次样本标准差的无偏估计值，计算公式如(2-31)和 (2-32)所示：

$$E[x]\leftarrow E_{\mathrm{B}}[\mu_{\mathrm{B}}] \tag{2-31}$$

$$Var[x]\leftarrow\frac{m}{m-1}E_{\mathrm{B}}[\sigma_{\mathrm{B}}^2] \tag{2-32}$$

在测试阶段，对于一个输入的测试样本，前向传播时 BN 层的输出结果可以用公式(2-33)计算得出：

$$y=\frac{\gamma}{\sqrt{Var[x]+\varepsilon}}\cdot x+(\beta-\frac{\gamma E[x]}{\sqrt{Var[x]+\varepsilon}}) \tag{2-33}$$

2.2.4 Dropout

过拟合是很多机器学习问题面临的一个重要问题，一旦发生过拟合，模型的泛化性能会大幅度降低。一般解决过拟合的策略是增加训练样本的规模和参数正则化。由于卷积神经网络中包含着数量十分庞大的的参数，导致训练过程中模型的过拟合问题时常发生。为了解决该问题，Hilton 等人[32]提出了 Dropout 技术，能够在训练样本较少的情况下有效的防止模型的过拟合现象，操作简单性且能得到很可观的效果。

工作流程如下：在网络训练的前向传播阶段，随机的以概率 p 临时删除网络中的某些神经元，删除的神经元的输出都被设置为 0。一般仅对全连接层的神经元进行 dropout 处理，其他各层的神经元保持不变。如图 2-7 所示：

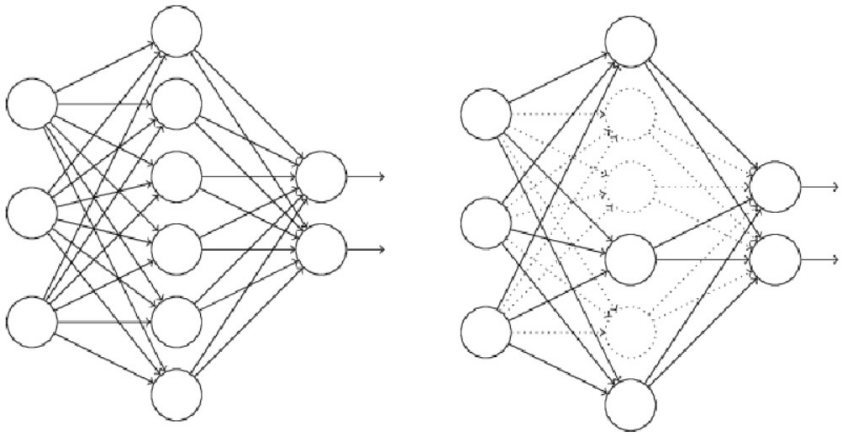


图 2-7 全连接（左）与 dropout（右）模型示意图

对深度网络进行 dropout 后，将会得到类似上图中右侧的网络结构。图中的虚线连接就是临时删除的节点间的连接，这些节点只是暂时被删除，当下一批训练样本输入时，会再次以概率 p 随机删除一批神经元。网络训练的反向传播阶段，会根据误差对当前网络中的权值和偏置进行更新。通过重复的迭代训练，网络最终学习到一系列参数。由于这些参数是在部分神经元被临时删除的情况下学习到的，当真正运行整个神经网络的时候意味着 p 倍的隐层神经元将被激活。为了抵消该问题的影响，为每个全连接层神经元的输出结果乘以一个 1/p。

Dropout 策略能够减少神经元之间的相互依赖关系，相当于同时训练若干个有差异性的网络，之后对这些不同的网络进行模型平均，得到具有较强泛化性能的网络。

2.3 本章小结

本章节中，首先简要介绍了传统神经网络的相关理论基础，详细阐述了网络结构构成以及误差反向传播算法的具体实现流程。之后对卷积神经网络的概念以及网络结构进行了较为详细的介绍，并总结了网络中各个非线性计算单元的计算方法，包括卷积运算、池化操作以及激活函数等。最后介绍了卷积神经网络训练过程中的数据预处理和避免过拟合的策略。以上介绍的这些作为下面几个章节的理论基础。

第三章 基于深度学习的人脸检测

本章中针对人脸检测任务中存在的复杂干扰以及检测速度等问题，提出了一种基于卷积神经网络的检测方法。本章节的安排如下：首先介绍传统的 Viola-Jones 检测器，之后介绍本文提出的基于卷积神经网络的人脸检测方法流程，最后在 Celeb A 数据集上对我们的方法和其他方法进行比较和评估。

3.1 传统的 Haar+adaboost 算法

2001 年，Viola 和 Jones[46]提出了一个用于人脸检测的算法流程，该算法简单易行且可以达到不错的检测效果，至今仍有较广泛的应用。算法流程大致分为三个部分：Harr 特征和积分图，基于 AdaBoost 算法的特征选择，以及建立级联分类器。经过一系列研究人员的研究，Viola-Jones 检测器不断得到改进和完善。

3.1.1 Harr 特征和积分图

一张正面对齐的人脸图像中，五官的纹理特征和相对位置存在一些基本的共性，比如眼睛区域比人脸其他区域的亮度低很多，鼻子区域是脸部亮度较高的区域。由于人脸具有上述特性，因此可以用一个反映图像灰度变化的特征来对人脸图像区域进行描述。Haar-like 特征考虑的是图像不同位置中相邻的矩形区域之间的像素灰度变化，对于块特征的表达（鼻子，眼睛，嘴巴等）具有比较好的效果。人脸图像的 Haar-like 特征的计算方法十分简单，使用一些设计好的运算模板，将模板上的白色区域像素之和与黑色区域的像素之和做差，即可得到图像的 Haar-like 特征。

总得来说，基本对应以下几种情形，如图 3-1 所示，对于图中的(a),(b),(e)三种模板，特征的计算公式为：$F = Sum_{white} - Sum_{black}$， (c)(d)两种模板中，黑色区域面积白是色区域的面积的一半，所以特征的计算公式为：$F = Sum_{white} - 2 \times Sum_{black}$。

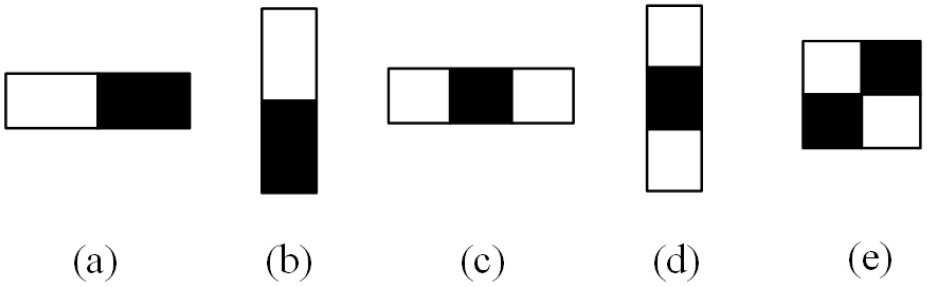


图 3-1 几种经典的矩阵特征

除此之外，又演变出了几种改进版的特征，极大的丰富和扩展了 Haar 特征。如图 3-2 所示，分别为扩展后的几种经典矩阵特征，分别为边缘特征、线性特征、中心环绕特征以及对角线特征。

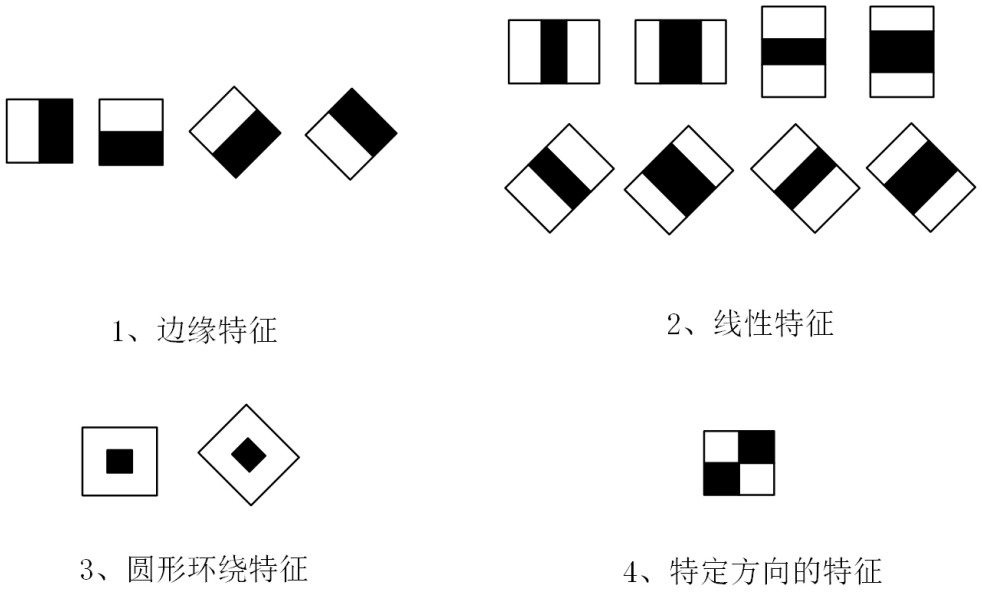


图 3-2 扩展后的四种经典矩阵特征

计算 Haar-like 特征时，由于需要遍历到矩形模板中的所有像素点，并对这些区域内的像素值进行求和，其中的运算量非常庞大且存在重复计算。由此 Viola-Jones 检测器提出了积分图的概念，计算特征时，每个像素只需要被遍历一次，这样计算特征的效率非常高。

图像上任意一点处积分图像值的计算方法如式(3-1)所示，位于该点左上角的所有像素之和即为积分图上该点处的像素值：

$$I(x,y)=\sum_{x'\le x}\sum_{y'\le y} f(x',y') \tag{3-1}$$

其中 $I(x,y)$ 表示像素的累加和，$f(x',y')$ 为图像上的像素点。当需要计算某个区域的特征时，首先计算图像的积分图并保存到内存中，需要使用时直接引用这些数值。之后通过积分图像 $I(x,y)$ 来计算图像上任意一个矩形区域的像素和，这些区域的像素值累加可以通过积分图上的四个像素点做简单的加减法计算得到。最后，使用矩形区域内的像素和来计算该区域的 Haar-like 特征。特征图能够避免重复运算，极大的提高特征的计算效率。

3.1.2 Adaboost 算法与分类器级联

除了特征提取阶段，检测器的速度也取决于分类过程运行的速度。分类过程的速度往往取决于分类器的复杂程度，即由特征向量转变成类别信息过程中的复杂度。简单的分类器由于模型简单，运算代价很小，但具有比较弱的分类能力，无因此法获得很好的分类准确度；而复杂的分类器的分类能力强，能够得到较好的分类结果，但是计算代价也会有所增多。为了兼顾计算代价和分类准确度两方面，Viola-Jones 检测器通过 AdaBoost 方法来学习分类器，达到实现相同的分类准确率的同时减小计算开销。

1997 年提出的 AdaBoost[47]算法，是英文”Adaptive Boosting“（自适应增强）的缩写。该算法的基本思想是：一个强分类器可以由多个弱分类器的线性组合构成，其中的每个弱分类器是使用不同的训练集进行训练得到的。一个弱分类器的定义如公式(3-2)所示：

$$h(x,f,p,\theta)=\begin{cases}1 & pf(x)<p\theta\\ 0 & \text{其他}\end{cases} \tag{3-2}$$

一个弱分类器 $h(x,f,p,\theta)$ 由以下四部分组成：子窗口图像 x，该窗口的 Haar-Like 特征 $f(x)$，不等号方向指示 p，以及阈值 θ。对弱分类器的训练就是寻找关于特征 f 的最佳阈值的过程。

具体操作过程如下：

设输入的 n 个人脸训练样本为：$\{(x_1,y_1),(x_2,y_2),\mathrm{L}\ ,(x_n,y_n)\}$，其中 x_i 表示输入的人脸样本图像，$y_i\in\{0,1\}$ 分别表示负样本和正样本，其中负样本数为 m，正样本数为 l 。

⑴ 初始化每个样本的权重 $w_i,i\in \mathrm{D}(i)$；

⑵ 对每个 $t=1,\mathrm{L}\ ,T$（T 为弱分类器的个数）：

① 把权重归一化为一个概率分布，如式(3-3)所示：

$$w_{t,i}=\frac{w_{t,i}}{\sum_{j=1}^{n}w_{t,j}} \tag{3-3}$$

② 从训练数据集中训练一个弱分类器 h_j，计算弱分类器在对应特征上的加权错误率，如式(3-4)所示：

$$\varepsilon_j=\sum_{i=1}^{n}w_t(x_i)\left|h_j(x_i)\neq y_i\right| \tag{3-4}$$

③ 选取错误率最小的弱分类器 h_t，作为该轮迭代的最佳弱分类器 ε_t；

④ 根据这个最佳弱分类器，更新样本的分布，如式(3-5)所示：

$$w_{t+1,i}=w_{t,i}\beta_t^{1-\varepsilon_i} \tag{3-5}$$

其中，若 x_i 被正确地分类，则 $\varepsilon_i=0$，被错误地分类，则 $\varepsilon_i=1$，$\beta_t=\frac{\varepsilon_t}{1-\varepsilon_t}$；

⑶ 最后的强分类器为(3-6)所示：

$$h(x)=\begin{cases}1 & \sum_{t=1}^{T}\alpha_t h_t(x)\geq\frac{1}{2}\sum_{t=1}^{T}\alpha_t\\ 0 & \text{其他}\end{cases} \tag{3-6}$$

其中 $\alpha_t=\log\frac{1}{\beta_t}$。

单个强分类器的准确度往往无法满足检测任务的要求，因此需要训练多个强分类器，并把它们级联在一起。这样一方面可以减少强分类器的输入规模，快速的过滤掉过多的背景，使得总体的时间开销降低，另一方面能够使强分类器进行联合，提升检测的准确率。检测过程如图 3-3 所示。

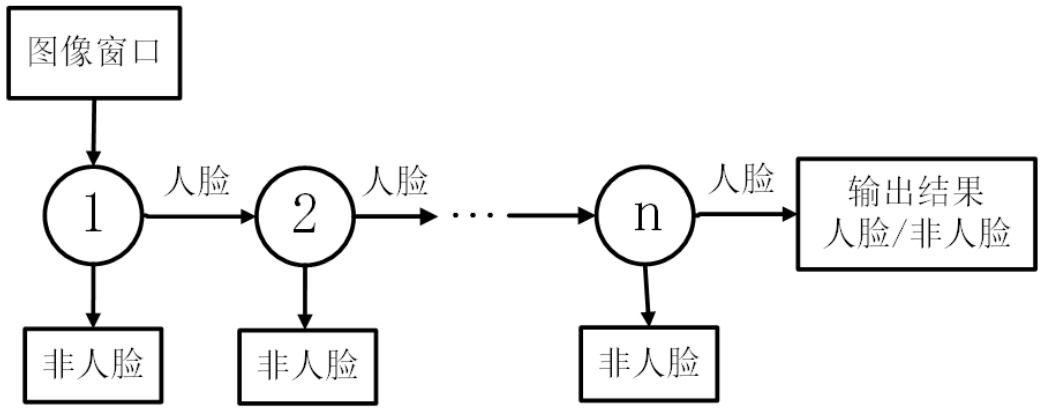


图 3-3 人脸检测过程示意图

级联分类器的思路如下：对强分类器根据复杂度大小进行排序，前面的强分类器复杂度较低，后面的强分类器复杂度较高。从前往后，分类器的复杂度逐级

升高。对于一个输入的图像窗口，首先由最前面的分类器(即最简单的分类器)对其进行分类，如果该窗口被判断为非人脸图像，则当作背景直接排除掉，不会送入到后面的分类器中继续进行分类；否则下一级分类器将会继续对该窗口进行分类，直到该检测窗口被排除或者被所有的分类器均判定为人脸。每经过一级分类器，输入到下一级分类器中需要判别的窗口数量就会减少。这样做的好处是，能够通过较少的计算排除掉大部分的非人脸窗口，降低检测所需要的时间开销，提升人脸检测的速度。

3.2 基于深度学习的人脸检测算法

3.2.1 相关研究工作

人脸检测作为计算机视觉领域中的一个基本问题，是人脸识别任务前面的必要步骤。因此，拥有很广阔的研究前景以及研究的必要性。现存的许多人脸检测系统需要面临的一个共同问题是，在检测的准确率和速度之间的权衡，要想获得较高的准确度，就需要以牺牲检测速度为代价。另外，由于人脸检测的输入图像中存在各种复杂干扰，给人脸检测任务带来了极大的不便，经常会成为影响检测准确度的关键问题。因此，为了满足实际应用中对检测器效率和性能的要求，设计一种快速又准确的人脸检测器变得越来越重要。

前一节我们介绍了 Viola-Jones 检测器。作为一种具有里程碑意义的方法，Viola-Jones 检测器取得了很大的成功，至今依然得到广泛的应用。该算法的三个关键要素是：快速的特征计算方法，有效的分类器学习方法以及高效的分类策略。随后涌现出了很多基于 Viola-Jones 人脸检测器的改进工作，主要围绕检测流程中的三个关键要素进行，对其中一点或者几点进行改进。随着深度学习技术的不断发展和广泛应用，基于深度学习的人脸检测方法应运而生[15]，大致可以分为两类：一类是基于目标检测的方法，即使用把目标检测的模型迁移到人脸图像检测的任务上来；另一类是级联法，如 MTCNN[17]，CascadeCNN[15]等。

大部分的检测器都存在一个在速度和检测准确度之间的权衡问题，为了获得更加准确的检测结果，势必要牺牲检测速度。随着人脸检测的应用场景越来越广泛，检测算法的速度越来越受关注，实时性渐渐成为衡量检测器性能的一个重要标准。为此，本文选择了一种快速且检测准确率较高的目标检测模型，在此基础上对其网络结构和参数设置进行改进，来实现实时的人脸检测。

这一节，我们首先详细的对本文使用的人脸检测算法进行阐述，之后介绍改进的卷积神经网络模型结构的细节以及检测流程的改进细节。

3.2.2 YOLO 目标检测模型

1、算法概况：

如图 3-4 所示，该图是 YOLO 目标检测算法[14]的流程图。YOLO 的主要思想是：将目标检测的流程统一到单个卷积神经网络中，利用卷积神经网络回归出目标的位置坐标，实现端到端实时目标检测任务，具体做法如下：

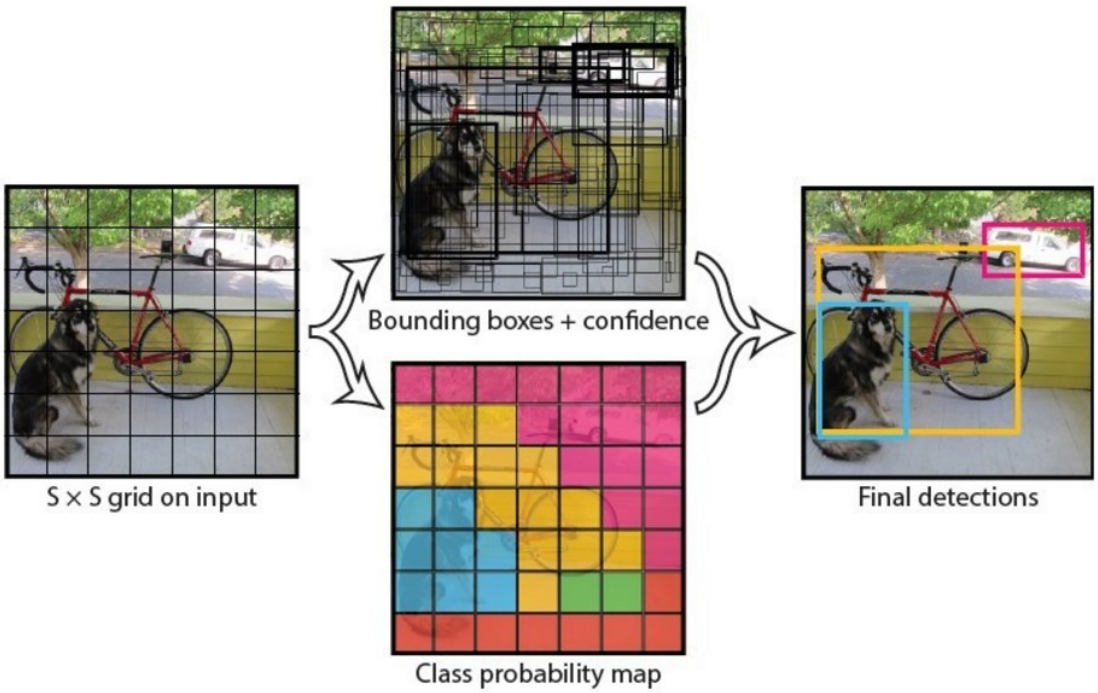


图 3-4 YOLO 目标检测算法流程[14]

把输入的人脸图像划分成不重叠的 M×M 个网格，将整幅图像作为卷积神经网络的输入，如果一个目标的中心落入格子，该格子就负责检测该目标。每个网格分别预测 B 个检测窗口，每个窗口包含人脸的位置坐标 (x, y, w, h) 以及该预测窗口的置信度 C。(x, y) 表示预测窗口中心点坐标，相对于对应的网格归一化到(0,1)；(w, h) 表示预测窗口的宽和高，相对整幅图像的宽和高做了归一化，范围也是(0,1)。

置信度值表示该预测窗口是否包含一个目标以及这个坐标预测的准确度。置信度的计算方式如式(3-7)：

$$C = P(Object) * IOU_{pred}^{truth} \tag{3-7}$$

其中，$P(Object)$ 为改预测窗口含有目标的概率，IOU_{pred}^{truth} 为预测窗口和真实的目标窗口之间的交并比(IOU)。如果一个网格里没有目标，置信度值为 0,有目标则该值为 1。

此外，每个网格还需要预测 N 个类别的信息，即 N 个类别概率。所以每个网

格的输出为一个 5×B+N 的向量，整个卷积神经网络的最后一层全连接层输出一个 M×M×(5×B+N)的向量，作为最终的预测结果。

2、损失函数

为了让坐标(x,y,w,h)，置信度和类别信息这个三个方面达到很好的平衡，设计如下的损失函数，式(3-8)：

$$\begin{aligned} Loss = & \lambda_{coord}\sum_{i=0}^{M^2}\sum_{j=0}^{B}\mathrm{I}_{ij}^{obj}(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2 \\ & +\lambda_{coord}\sum_{i=0}^{M^2}\sum_{j=0}^{B}\mathrm{I}_{ij}^{obj}(\sqrt{w_i}-\sqrt{\hat{w}_i})^2+(\sqrt{h_i}-\sqrt{\hat{h}_i})^2 \\ & +\sum_{i=0}^{M^2}\sum_{j=0}^{B}\mathrm{I}_{ij}^{obj}(C_i-\hat{C}_i)^2 \\ & +\lambda_{noobj}\sum_{i=0}^{M^2}\sum_{j=0}^{B}\mathrm{I}_{ij}^{noobj}(C_i-\hat{C}_i)^2 \\ & +\sum_{i=0}^{M^2}\mathrm{I}_{i}^{obj}\sum_{c\in classes}(p_i(c)-\hat{p}_i(c))^2 \end{aligned} \tag{3-8}$$

其中，(x_i,y_i,w_i,h_i)为网络最后一层输出的预测坐标值，$(\hat{x}_i,\hat{y}_i,\hat{w}_i,\hat{h}_i)$为目标实际位置的坐标值，$C_i$和$\hat{C}_i$分别为每个网格预测窗口的置信度值和真实的置信度值，$p_i(c)$和$\hat{p}_i(c)$为每个网格类别信息的预测值与真实值。I_i^{obj}判断是否有目标的中心落在网格内，为 1 则网格负责预测该目标，为 0 则不参与到损失函数的计算。I_{ij}^{noobj}表示对没有目标的网格的损失进行计算。为了平衡位置坐标和类别的维度对损失函数带来的影响，设置一个权值λ_{coord}来调节位置坐标在损失函数中的作用，类别误差项前面的系数设置为 1。

如果一个网格中没有目标，那么就会将这些网格中预测窗口的置信分数设置为到 0，相比之下有目标的网格很少，如果，有目标和没有目标的网格在计算损失函数时的权重一样大，会导致网络的不稳定甚至发散，为此，为不含目标的网格赋予一个小的损失权重λ_{noobj}。 此外，小目标的预测结果相对真实位置的偏移会比大目标的结果位置偏移更加严重，所以为了缓和这个问题的影响，这里选择使用预测结果的平方跟来代替原来的宽和高。

3、测试阶段

每个网格的类别置信分数为每个网格预测的类别信息$P(Class_i\,|\,Object)$和每个网格中检测窗口预测的置信度 C 的乘积，如(3-9)所示的式子：

$$P(Class_i)*IOU_{pred}^{truth}=P(Class_i|Object)*P(Object)*IOU_{pred}^{truth} \quad (3\text{-}9)$$

若得到的值高于设置的阈值，则保留该预测窗口，否则舍弃掉。监测到的人脸窗口往往存在较多的重叠，为了去掉多余的重叠的窗口，对保留下来的预测窗口进行非极大值抑制(NMS)处理，得到最终的检测结果。

3.2.3 VGG16 网络模型

VGG16 网络[12]是一种可用于特征提取的卷积神经网络，该网络的卷积层和全连接层的层数加起来总共有 16 层。将固定大小的 RGB 三通道图像作为网络的输入，之后依次通过网络中的各个非线性计算层。把每两个或三个相邻的卷积层组成的卷积单元模块看为一个整体，命名为 Block。特征图(feature map)通过每个 Block 后会输入到一个最大池化(Max-pooling)层，用于对输入的特征图进行降采样，降低特征的维度并能够保证特征具有平移不变性。图像经过所有的 Block 单元之后，输出若干张特征图，之后这些特征图依次输入到网络最后的三个全连接层中。最后，整个网络的输出层是一个 softmax 多分类器。

每个 Block 单元中卷积核大小均为 3×3，卷积核每次向右或向下滑动的步长为 1。在卷积操作之前，首先对输入的特征图进行边缘扩充（padding），大小为 1。这样能够保证每个卷积层的输出特征图大小与输入特征图大小一致。Block 单元中卷积层之间没有最大池化层，在每个 Block 单元之后连接一个最大池化层，池化的步长和核大小均为 2，所以经过每个 Block 之后特征图的大小变为原来尺寸的一半。VGG16 网络使用 ReLU 作为激活函数，每个卷积层和全连接层的输出结果都会通过激活函数来进行非线性运算。此外，为了提高网络的泛化能力，防止过拟合，在网络的训练阶段使用了 dropout 技术，仅在前两个全连接层之后的网络连接使用。

卷积神经网络每一层输出的特征图上的像素点对应了原始特征图上的一定区域的感受野，如下图 3-5 所示，假设原始的特征图大小为 M×M×K，经过大小为 3×3，5×5 和 7×7 大小的卷积核进行卷积操作后，得到的特征图大小如图所示，设卷积核滑动步长为 1，特征图扩充为 0，每层卷积层卷积核的通道数均为 N。可以看到，原始图像经过两层 3×3 大小的卷积核之后，输出特征图上的每个像素点覆盖的感受野区域大小等效于一个 5×5 大小的卷积核覆盖的区域范围。经过三层 3×3 大小的卷积核之后，每个像素点覆盖的感受野区域大小等效于一个 7×7 大小的卷积核覆盖的区域范围。然而，小卷积核能够较多的关注图像中的细节信息，提取到更加细致的特征，卷积核越大，对图像细节信息的表达越弱。

同时，由于每次卷积操作之后都要通过 ReLU 激活单元进行非线性转换，多

个小核卷积层的堆叠能够实现多次非线性转换，相比于单个大维度卷积核只进行一次转换，得到的特征具有更好的非线性性，能够更好的对图像进行表达。

具体的网络配置如表 3-1 所示。

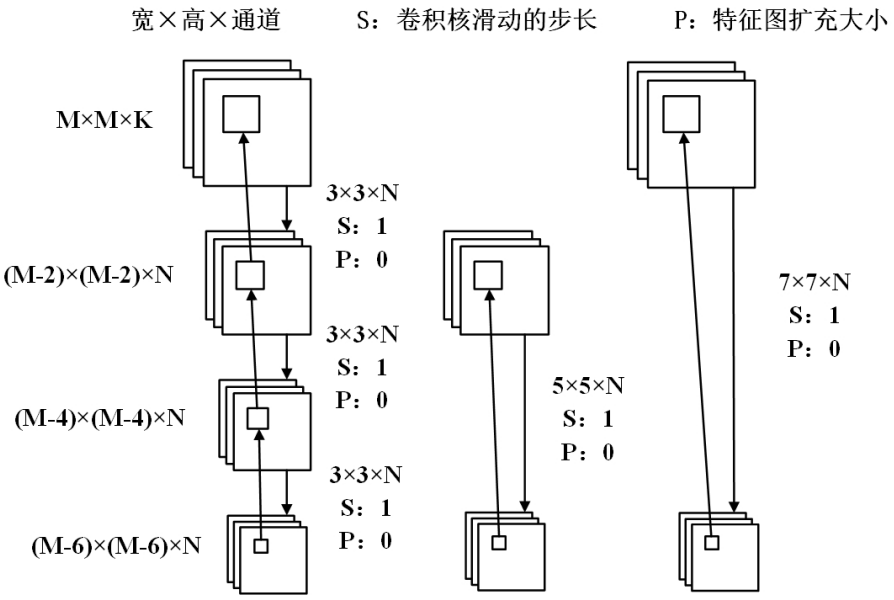


图 3-5 不同大小卷积核与感受野的关系

3.2.4 深度可分解卷积

传统卷积操作中，卷积核可以表示为一个四维的矢量(M,M,D,N)，其中 M 为卷积核的大小；D 为卷积核的深度，与输入特征图的通道数一致；N 为卷积层卷积核的个数，即输出特征图的通道数。卷积过程中，卷积核在输入特征图上以一定的步长滑动，一次卷积操作可以看成是完成了一次跨通道和跨空间像素的聚合，同时寻找空间和通道之间的相关性。在后面的实践中发现[48]，通道之间的相关性和空间的相关性是完全可分离的，可以对它们进行分别映射，深度可分解卷积的提出就是基于这样的假设。深度可分解卷积是一种对标准的卷积进行分解的操作，分解为一个深度卷积和一个点卷积。深度卷积是分别为每个通道单独的执行空间卷积的操作，卷积核的深度为 1，卷积核个数与输入的通道数相同，每个卷积核分别与输入的各个通道进行卷积操作，得到一个通道数和输入图像一致，空间尺度变小的中间输出；点卷积即卷积核大小为 1×1 的跨通道卷积操作，每个卷积核深度为输入图像的通道数，卷积核的个数决定了该卷积层输出特征图的通道数，图 3-6 为深度可分解卷积操作的示意图。

表 3-1 VGG16 网络模型结构[12]

	Conv 3×3	Conv 3×3	Conv 3×3	Pooling	输出大小
输入大小	224×224×64				
Block 1	核大小：3 核数量：64 步长：1 扩充：1	核大小：3 核数量：64 步长：1 扩充：1		核大小：2 步长：1 类型：Max	112×112×64
Block 2	核大小：3 核数量：128 步长：1 扩充：1	核大小：3 核数量：128 步长：1 扩充：1		核大小：2 步长：1 类型：Max	56×56×128
Block 3	核大小：3 核数量：256 步长：1 扩充：1	核大小：3 核数量：256 步长：1 扩充：1	核大小：3 核数量：256 步长：1 扩充：1	核大小：2 步长：1 类型：Max	28×28×256
Block 4	核大小：3 核数量：512 步长：1 扩充：1	核大小：3 核数量：512 步长：1 扩充：1	核大小：3 核数量：512 步长：1 扩充：1	核大小：2 步长：1 类型：Max	14×14×512
Block 5	核大小：3 核数量：512 步长：1 扩充：1	核大小：3 核数量：512 步长：1 扩充：1	核大小：3 核数量：512 步长：1 扩充：1	核大小：2 步长：1 类型：Max	7×7×512
全连接层 1	4096 个神经元				1×1×4096
全连接层 2	4096 个神经元				1×1×4096
全连接层 3	4096 个神经元				1×1×4096

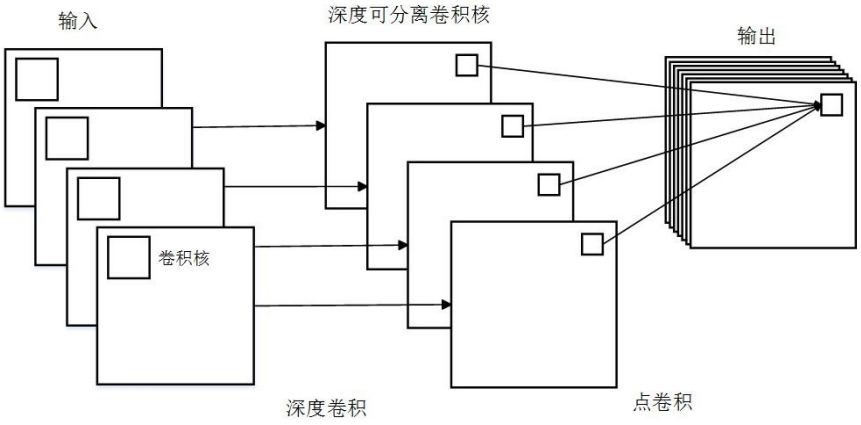


图 3-6 深度分解卷积操作示意图

直观看来，这种分解和标准的卷积操作在效果上是等价的。假设输入图像维度是 11×11×3，标准卷积核大小为 3×3×3×16，步长为 2，扩充为 1 个像素，则输出的特征图的大小为 6×6×16。进行深度分解之后，输入图像大小不变，先通过一个维度是 3×3×1×3 的深度卷积，得到 6×6×3 的中间输出，之后再通过一个维度是 1×1×3×16 的卷积核进行点卷积，得到输出特征图的大小同样为为 6×6×16。

假设输入特征图为 $D_F \times D_F \times M$，其中 D_F 为输入特征图的大小，M 为其通道数，输出的特征图为 $D_G \times D_G \times N$，D_G 为输出特征图的大小，N 为其通道数，卷积核大小为 $D_K \times D_K$。标准的卷积操作需要的计算量为 $D_K \times D_K \times M \times N \times D_F \times D_F$，而深度分解卷积操作所需要的参数数量为 $D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F$，这里忽略了加法的计算量。两者进行比较，可以得到式(3-9)中的结果：

$$\begin{aligned} &\frac{D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F}{D_K \times D_K \times M \times N \times D_F \times D_F} \\ &= \frac{1}{N} + \frac{1}{D_K^2} \end{aligned} \tag{3-9}$$

可见，当输出的通道数 N 很大时，计算量的差异主要取决于卷积核的大小，由于 VGG16 网络中均为 3×3 大小的卷积核，因此深度分解卷积可节约 8~9 倍的计算量。深度可分解卷积结构可以有效减少网络中的参数数量，降低模型规模，很大程度上减少了网络的计算量和硬件要求，提高运算效率。

3.2.5 改进的人脸检测方法

自 2012 年 AlexNet[32]获得 ImageNet 分类比赛的冠军之后，卷积神经网络开始被广泛的使用。随着科研和实际应用中对深度学习任务精度的要求越来越高，设计出更深更复杂的网络逐渐地成为当下卷积神经网络的一个发展趋势。深度网络的设计隐含了其在速度和准确度之间的权衡，网络结构越复杂，其提取到的特征更精细抽象度更高，准确度更高，相应的计算的时间和空间成本也越大。然而在实际应用中，人脸检测对速度的要求较高，且考虑到硬件资源有限，这就需要一种轻量级的网络模型，能够在不降低太多检测精度的同时，实现更快的检测速度。

这一节，我们将介绍本文改进的卷积神经网络结构的模型细节。 VGG16 网络可以对人脸特征进行很好的表达，但我们希望能够把网络结构加深，从而提取到更加抽象的特征表达，同时能够保证检测的速度和精度。因此，本文在网络结构加入可分离卷积操作，极大的减少网络中的参数以及训练速度。

保持 VGG16 网络中 Block1~Block5 的结构不变，即在堆叠的卷积运算后进行池化操作，以保证网络对图像特征的提取能力。不同的是网络中的卷积操作用深度可分解卷积代替，把原始的 3×3 的卷积核分解为两个卷积操作，卷积核大小分别为 3×3 和 1×1，并在每一个卷积层之后加入批度归一化（Batch Normalization）处理，如图 3-7 所示。同时，把输入层的大小修改为 448×448，使得卷积网络提取到更加细节的特征，网络中间产生的特征图大小会相应的依次发生变化，网络的详细结构模型如表 3-2 所示。

此外，考虑到 ReLU 激活函数对学习率的大小较敏感，我们把网络中的每一个卷积层和全连接层后面使用激活函数设置为 leaky-ReLU。为了防止网络的过拟合，网络的训练阶段，在全连接层的后面使用了 dropout 技术。由于整个网络的大部分参数集中在网络的全连接层，最后三层全连接层的参数个数占了整个网络参数总数 90%以上。为了减少网络中的参数数量，本文对网络全连接层的结构进行缩减，仅保留一个全连接层，去掉后面的两个全连接层。

把输入图像分为 11×11 个小网格，每个网格分别预测出两个预测窗口的位置坐标以及一个两类的类别信息，即是/不是人脸。所以，整个网络的输出为一个 11×11×(2+2×4)=1331 的向量。本文采用的卷积神经网络模型的整体结构如图 3-8 所示。

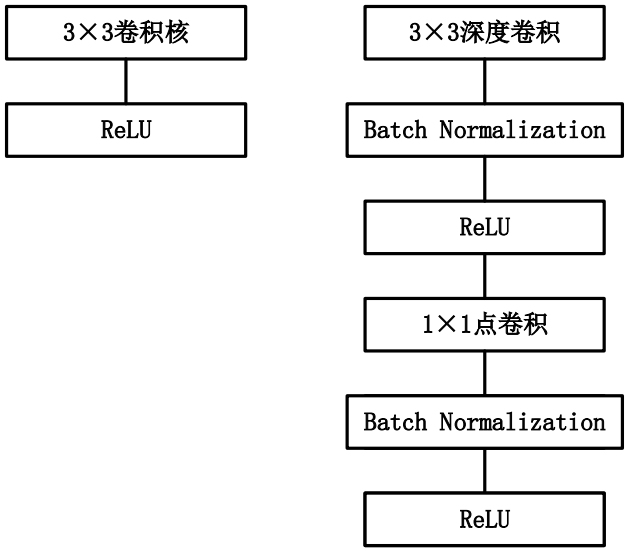


图 3-7 卷积神经网络中的卷积模块

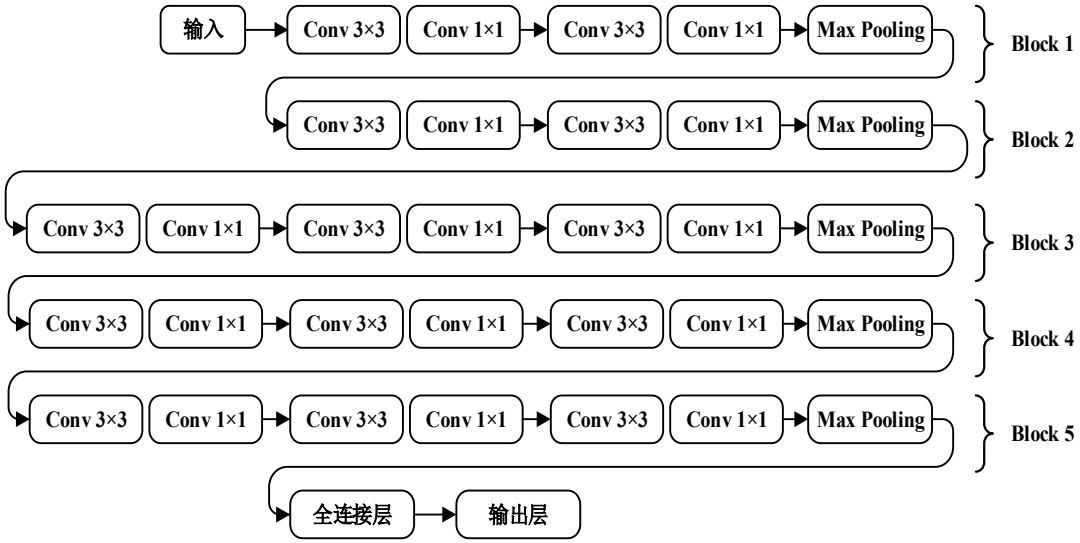


图 3-8 本文提出的卷积神经网络模型结构

3.3 试验设计与结果分析

3.3.1 图片数据集与预处理

本文人脸检测采用的是香港中文大学多媒体实验室提供的 Celeb A[49]数据集。其中包含 202,599 张不同姿态，表情的人脸图像。每张图像包含 5 个关键点（双眼、鼻子和左右嘴角）的坐标信息以及人脸的位置信息（左上角坐标，宽和高），此外还包含性别信息，是否带眼镜等等 40 种属性。

从数据集中随机选取 25000 张图像作为训练集，5000 张图像作为测试集。我们对输入图像做镜像操作，目的是为了实现数据扩增，提高网络的泛化能力。为了便于卷积神经网络的计算，首先将图片的大小调整到 448×448×3。之后计算训练数据集的均值图像，每张训练图像在输入到网络之前首先减去该均值图像，作去均值处理，并把像素值归一化到(0,1)范围内，这样做能够是数据的分布更加均匀，加速网络的收敛速度。

图 3-8 Celeb A 数据集示意图

3.3.2 网络模型的训练与实验结果分析

模型在深度学习开源平台 caffe[50]上进行训练，分别对 YOLO 模型中使用的网络和本文改进后的网络进行训练，学习率初始值设置为 0.001，每 5000 次迭代学习率进行一次衰减，衰减为原来的 0.1 倍，动量因子设置为 0.9。网络的卷积层和全连接层的参数均初始化为均值为 0 方差为 0.01 的高斯分布。由于人脸检测是一个二分类问题，损失函数中的超参数设置如下：λ_{coord} 设置为 2，λ_{noobj} 设置为 0.5。两个网络均在迭代 25000 次后停止训练。测试阶段，把原始人脸图像分成 11 个小网格，并把图像缩放至网络输入的大小，每个网络的输出层均输出一个 1331 维的向量，分别的每个网格存在目标的置信度，是人脸的概率以及人脸的位置信息。利用式(3-9)将网格内有人脸存在的置信度和预测的人脸类别概率作乘积，计算出

每个网格内人脸的置信分数。设置阈值为 0.25，对于类别概率小于这个阈值的窗口进行排除，之后对剩余的检测窗口进行非极大值抑制处理，得到最佳的检测窗口，并在原始图像上用红色矩形框进行显示。

我们首先评估了几种激活函数对网络收敛速度的影响。参与实验的激活函数有 Sigmoid，ReLU 以及 leaky-ReLU。图 3-9 显示了几种激活函数随着迭代次数的增加，在 Celeb A 测试集上的结果变化。我们还比较了使用 leaky-ReLU 激活函数，并加入批度归一化（BN）操作的网络的收敛速度。可以看到，虽然在迭代到 20000 次时，网络的精度基本相同，但使用 Leaky-ReLU 的网络收敛速度要好于 Sigmoid 和 ReLU 函数。加入批度正则化后，网络收敛速度进一步提高。因此为了缩短训练时间，本文的网络结构在选择 Leaky-ReLU 作为激活函数的同时，也加入了批度归一化操作。

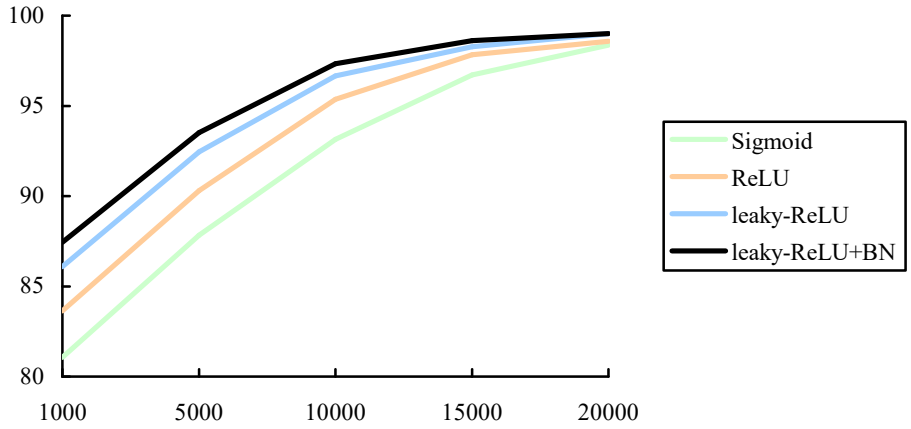


图 3-9 使用不同激活函数的精度变化

我们在 Celeb A 数据集上评估了本文中人脸检测网络的效果，分别评估了 YOLO 中使用的网络和本文使用的卷积神经网络在检测准确度和速度上的检测结果。评价指标为 ROC 曲线和常用的检测指标 AP（平均精度）。

ROC 曲线反映了二分类问题分类器的性能，横坐标为预测为人脸但实际为非人脸的样本在所有非人脸样本中的比例，纵坐标表示预测为人脸且实际也是人脸的样本在所有人脸样本中的比例。图 3-10 展示了在 Celeb A 数据集上进行人脸检测任务时的 ROC 曲线，由 ROC 曲线可以得到一个动态的全局性能的反映。图中黑色实线为本文使用的网络的 ROC 曲线，浅灰色实线和深灰色实线分别为 YOLO 中的网络结构和 VGG16 网络的实验结果，可以看到我们的网络在整体的性能上相比 YOLO 中的网络和 VGG16 网络均有所提升。

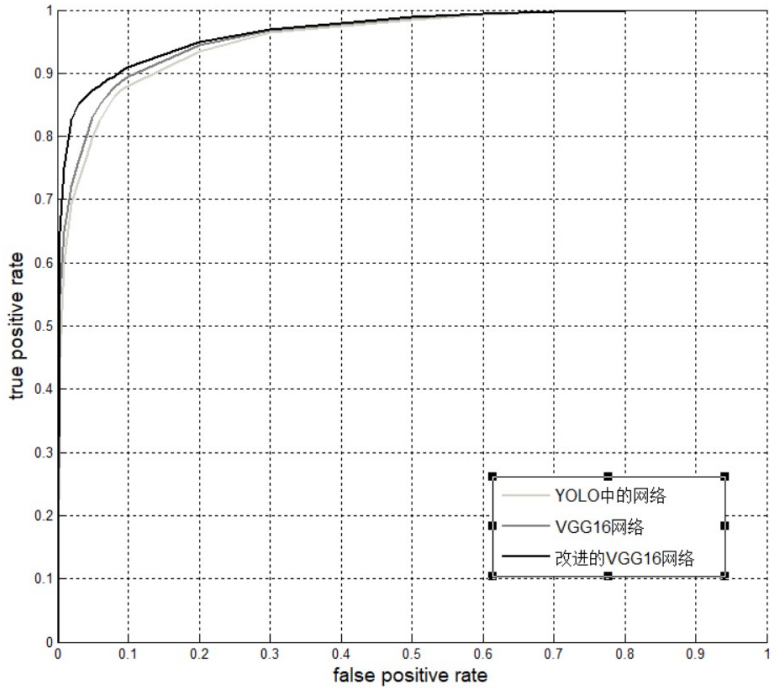


图 3-10 Celeb A 数据集上的 ROC 曲线

表 3-2 几种检测结果的对比

检测算法	检测时间（ms/im）	AP
Viola-Jones[46]	92	82.10%
Joint Cascade[51]	45	95.78%
Cascade CNN[16]	37	98.37%
YOLO[14]	52	98.20%
本文的检测方法	31	98.85%

表 3-2 展示了几种方法检测的对比结果，主要从检测的平均时间和平均精度两个指标进行比较。我们比较了传统的 Viola-Jones 检测器和其他基于深度学习的检测器。前三种检测器都是采用了级联的结构，Joint Cascade[51] 和 Cascade CNN[16] 中采用浅层的卷积神经网络作为分类器。可以看到，传统的 Viola-Jones 检测器在速度和平均精度上均与其他几种方法相差较多。Joint Cascade[51]取得了 9.78%的检测精度，但速度略慢，需要 45ms 来处理每张图片。Cascade CNN[16]相比 Joint Cascade[51]在检测速度和精度上都有所提升，达到了 98.37%的精度。但这两种方法都需要训练多个浅层的卷积神经网络，且这些训练是单独进行的，无法联合优化求解，导致无法充分利用卷积神经网络可以联合优化的特性。后两种方法中，仅使用一个卷积神经网络就可以完成端到端的预测，能够将检测和分类任务放在一

起进行优化。YOLO 算法中的网络的 AP 为 98.2%，但速度比较慢，这是因为其网络结构相比 Cadcade CNN 中的浅层网络要复杂很多。本文改进的网络能够得到 98.85%的 AP，相比 YOLO 中的网络我们的网络在精度方面有所提升，同时极大的减少了网络的复杂度。YOLO 检测模型需要的时间平均为 52 毫秒，而使用改进的网络后检测一张图像需要的时间平均为 31 毫秒。此外相比前面级联结构的检测器，我们的方法在速度和精确度上都有不同程度的提升，能够在不损失检测精度的同时提升检测速度，实现实时的人脸图像检测。

图 3-11 中展示了本文使用的检测方法与其他两种基于深度学习的方法的检测效果图，图中人脸目标具有不同分辨率大小以及不同的姿态表情。前三行分别为使用 Viola-Jones[46]，使用 YOLO[14]和 Cascade CNN[16]的检测结果；最后一行为使用本文方法的检测结果。前三幅图像中的人脸目标尺寸较大，我们分别选择了正面以及不同角度的侧面图象进行效果展示。能够看到前三种方法中，Viola-Jones[46]的检测结果在遇到姿态变化较大时，没有另外两种方法的结果鲁棒，预测窗口会产生较大的偏移。但是对于正面姿态的人脸目标，前三种方法的检测结果均较好，能较准确的定位到人脸图像的位置。我们的检测算法对于正面姿态的人脸的检测结果与前面三种差别不大，但对于姿态等的变化较鲁棒，即使是侧面的人脸依然能被很好的检测出来。后三幅图像中，人脸目标较小，且存在不同程度的干扰，如遮挡，姿态，表情等。前三种方法的检测结果有不同程度的位置偏移，其中 Viola-Jones[46]的检测结果远没有其他两种效果好，检测窗口产生了较大的偏移；YOLO[14]对于目标的大小表现得较为敏感，在小目标上的检测效果没有大目标效果好；Cascade CNN[16]对于姿态表情等变化的影响仍然不够鲁棒；我们的算法能较好的定位人脸的位置，但对于目标较小的图像的预测结果相比标定信息稍有偏移，但整体偏差不大。

3.4 本章小结

本章首先介绍了经典的人脸检测算法，之后介绍了基于深度学习的人脸检测方法，并详细介绍了针对网络结构对卷积神经网络进行的改进，通过实验结果不难看出，本文改进的网络结构在不损失检测结果精度的同时能够降低网络的复杂度，提升检测的速度。本章的人脸检测作为下面两章的预处理步骤，后面两章我们将详细的介绍本文的识别算法流程和涉及到的网络模型

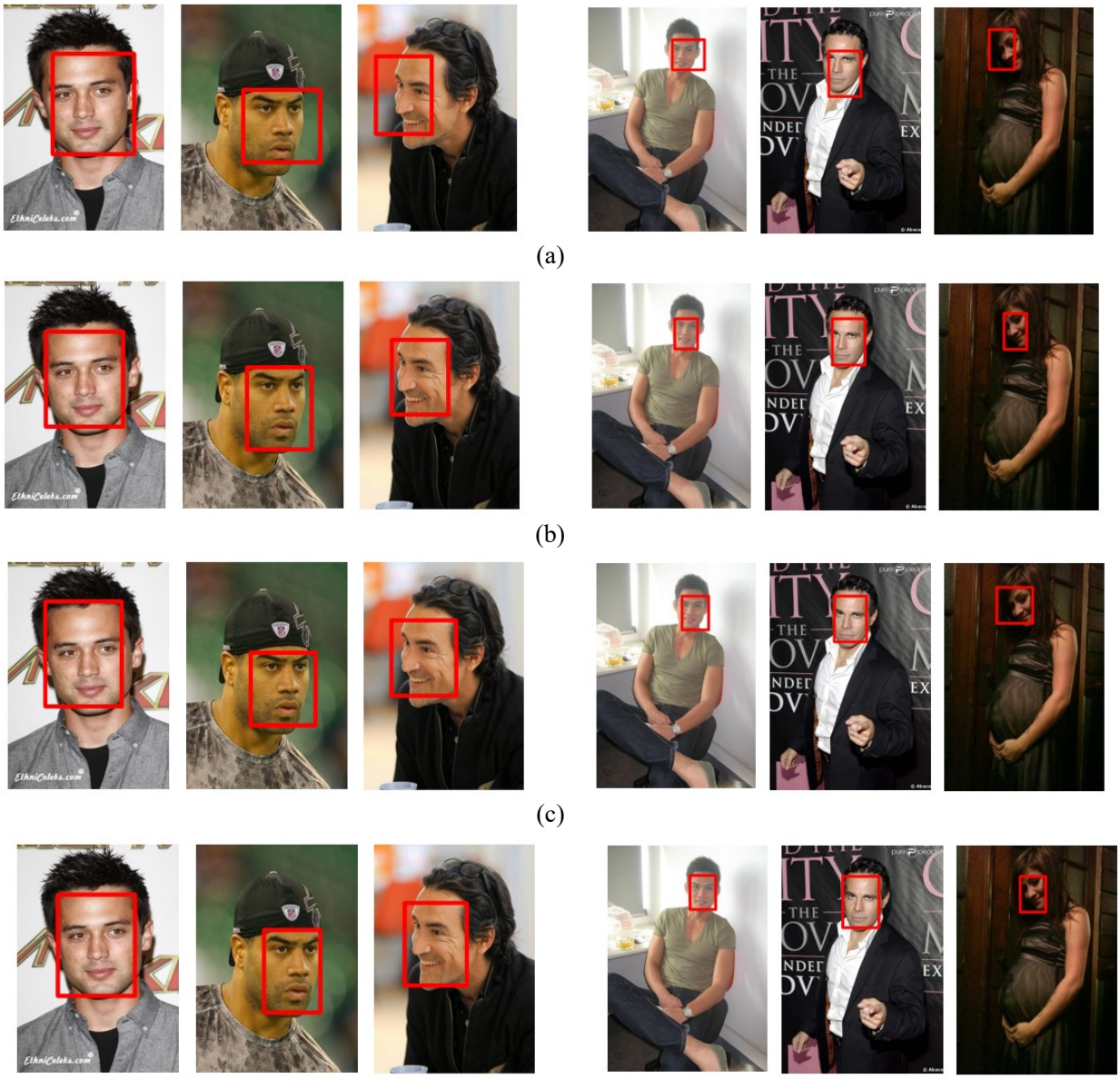


图 3-11 Celeb A 数据集上的检测效果图

(a) Viola-Jones；(b)YOLO；(c) Cascade CNN；(d)本文方法

第四章 基于深度学习的人脸特征提取

人脸特征提取是人脸识别中最关键的步骤，也是其中的一个重要研究方向。人脸特征提取旨在通过特征描述算子来描述人脸图像的形状、纹理、灰度、梯度等信息，以及更加高级的特性，来更好的区分人脸对象的身份信息。

本章的开始我们先介绍几种传统的人脸特征描述子，之后针对复杂环境下的人脸识别问题，提出一种基于深度学习的特征提取方法，该特征能够对人脸进行很好的表达的同时，也对各种复杂干扰具有鲁棒性，并且影响到之后人脸识别任务的准确度。

4.1 传统的人脸特征提取方法

在第三章的 3.1 节中，我们已经介绍了一种人脸图像描述子——Haar 特征，本节中我们介绍另外两种特征——LBP 特征描述子和 SIFT 特征描述子。

4.1.1 LBP 特征

LBP（Local Binary Pattern，局部二值模式）特征[52]是一种用来描述图像局部纹理特征的描述子。LBP 衡量了一个像素点和它周围像素点的关系，具有灰度不变性，可以有效地消除光照对图像的影响，且具有计算简单，效果好等优点。

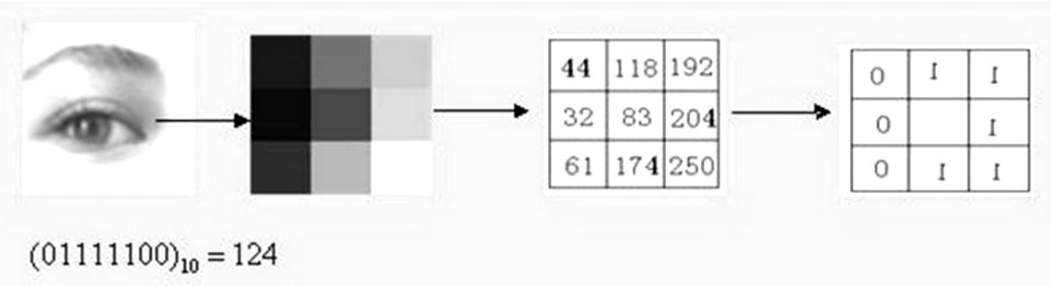


图 4-1 中心像素点的 LBP 值

对于图像中一个 3×3 大小的窗口，该窗口中心点像素的 LBP 值的计算方式如图 4-1 所示：将中心点的像素的灰度值设置为阈值，3*3 邻域内与中心点相邻的 8 个像素值分别与该阈值进行比较，若灰度值大于阈值，则该像素点的位置被标记为 1，否则为 0。围绕中心点的 8 个点经过比较之后，得到一个 8 位二进制数，将其转换为十进制即为该中心点的 LBP 值。LBP 值为 0-255 范围内的整数，该值可以反映局部区域内的纹理信息。

原始的 LBP 算法提出之后，针对不同问题的改进算法不断的涌现出来。为了实现旋转不变性，同时使纹理特征适应于不同的尺度， T. Ojala[53]等人提出了改进的 LBP 算子。将 3×3 的范围扩展到任意邻域，并将邻域的形状由正方形调整为圆形，半径为 R 的圆形邻域内包含任意多个像素点，通过像素的插值计算对角线上点的像素值，从而得到区域内 P 个像素点的 LBP 算子。由于 LBP 特征不具备旋转不变性，对图像进行旋转往往会得到不同的 LBP 值。Pietikinen 等人[54]针对此问题改进了 LBP 算子，将圆形邻域进行不同角度的旋转，之后计算初始的 LBP 值，取其中的最小值作为该邻域的 LBP 值。这样得到的特征能够具有灰度和旋转的不变性。

4.1.2 SIFT 特征

SIFT 算法是由 David G.Lowe[55]在 2004 年提出的经典算法，用来提取图像关键点周围的局部特征。该算法在不同的尺度空间上寻找特征点，这些特征点对光照，仿射变换和噪音等因素具有鲁棒性，之后计算出该特征点的方向信息。这些特征点包括图像的边缘点、角点、暗区的亮点及亮区的暗点等。

SIFT 特征的计算步骤如下：

(1).对图像进行二维高斯模糊，构建多分辨率下的高斯图像金字塔。之后根据高斯金字塔构建高斯差分尺度空间(DOG)，高斯差分尺度图像表示为式(4-1)的形式 ，即将每一组金字塔的相邻两层做差：

$$\begin{aligned} D(x,y,d) &= (G(x,y,kd) - G(x,y,d)) * I(x,y) \\ &= L(x,y,kd) - L(x,y,d) \end{aligned} \tag{4-1}$$

式中， $I(x,y)$ 表示输入的图像， $G(x,y,d)$ 表示高斯核函数， d 表示尺度空间因子，k 表示相邻层高斯图像尺度相差的比例因子。高斯差分尺度空间中的每一个像素点都要与其周围 26 个像素点上的像素值进行比较，这些像素点包括该点所在阶上下邻层图像的 18 个（9×2）像素值，以及该像素点同一层上周围的 8 个像素值。如果该像素点的像素值大于或小于其周围的 26 个相邻点的像素值，则该点为图像空间和尺度空间上的一个局部极值点。

(2). 高斯差分尺度空间上的极值点不是真正的极值点，这是由于该局部极值点是离散空间上的极值点。离散空间是对连续空间的采样，因此检测到的极值点往往不是真正的极值点。由于高斯差分算子会产生较强的边缘响应，这就需要对不稳定的边缘响应点和对比度较低的极值点进行剔除。利用 DOG 函数在尺度空间的曲线拟合来寻找极值点，以增强特征的抗噪声能力和稳定性。

(3). 生成 SIFT 特征向量描述符，包含如下两个步骤：

1、特征点主方向的确定：

为了使特征描述符具有旋转不变性，需要为每个特征点分配一个基准方向。对于高斯差分金字塔中检测出的特征点，计算该点的图像邻域内像素点的梯度和方向的分布特征。梯度幅值 $m(x,y)$ 和方向 $\theta(x,y)$ 的计算方法如式(4-2)和式(4-3)所示：

$$m(x,y)=\sqrt{[L(x+1,y)-L(x-1,y)]^2+[L(x,y+1)-L(x,y-1)]^2} \tag{4-2}$$

$$\theta(x,y)=\arctan\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)} \tag{4-3}$$

完成梯度和方向的计算后，将该点邻域内像素点的梯度幅值和方向利用直方图进行统计。梯度直方图的横轴为梯度的方向，将 0 到 360 度的方向范围分成 36 份，以 10° 圆周为一个单元。纵轴为每个梯度方向幅值的高斯加权组合。把特征点邻域内所有像素点的梯度方向对应到相应的方向单元中，直方图的峰值方向代表该特征点的主方向。

为了提升特征的鲁棒性，仅保留峰值大于或等于主峰值 80%的梯度方向作为该特征点的辅助方向，每个特征点可以同时具有一个主方向和多个辅方向。

2、生成 SIFT 特征向量

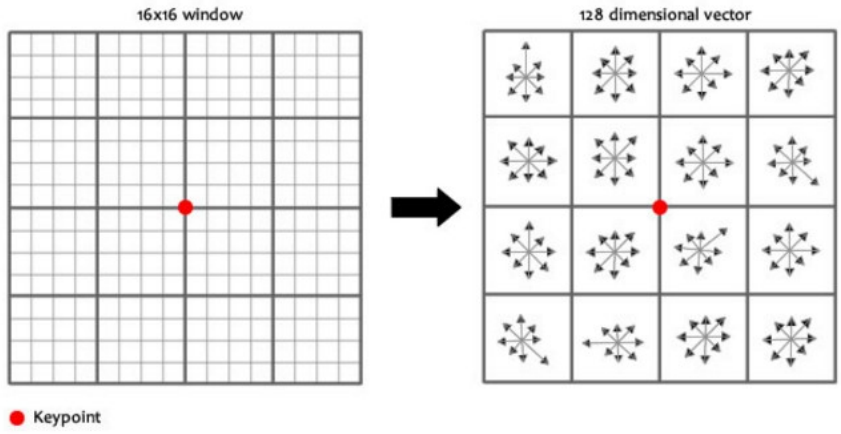


图 4-2 SIFT 特征梯度直方图[55]

通过以上步骤，可以得到一个关键点的三个信息 (x,y,σ,θ)，即位置、尺度和方向。之后，用一个向量描述该关键点，使其对光照，旋转等变化鲁棒。为了确保特征的旋转不变性，将坐标轴旋转为特征点的主方向。之后以该特征点为中心，在 16×16 的窗口内计算梯度直方图，如图 4-2 所示。把窗口内的像素划分为 4×4 的小窗口，把每个小窗口内像素点的梯度方向分配到 8 个方向区间中，并计算其权值生成 8 维的特征向量。整个窗口区域共能产生一个 4×4×8 =128 维的特征向量，即为该特征点的 SIFT 特征。

4.2 基于深度学习的人脸特征提取

在正面的人脸图像中，人工设计的特征描述符能够很好的对人脸图像进行描述。然而在非可控条件下，由于复杂干扰的存在，人工设计的特征描述子已经无法满足识别任务的要求，对不同的复杂干扰设计不同的特征描述符显得费时费力。卷积神经网络能够自动提取图像中的特征，避免了繁复的人工的特征设计工作，同时对平移、光照等变化表现的很鲁棒，能够对图像进行超完备的表达。

4.2.1 对比损失函数

对比损失函数(Contrastive Loss)[56]初主要应用在数据降维中，是用来比较两张图像是否属于一类的损失函数。原本相似的样本经过特征映射后，在新的特征空间仍然保持相似性；而原本不相似的样本，经过特征映射后，在特征空间仍不相似。

对比损失函数常用在孪生神经网络（siamese network）中，作为其卷积神经网络的损失函数。对于两张原始输入图像，经过卷积神经网络进行特征提取，图像从原来的数据空间映射到更容易被区分开的特征空间中。在此过程中，卷积神经网络需要学习原始数据空间到新的特征空间的映射关系，该映射函数能够使同类样本之间的距离最小化，不同类样本之间的距离最大化。对比损失函数就是用来对学习到的映射函数进行衡量，使网络学习到更加鲁棒的映射关系。

设由两个样本点组成的样本对 (x_1, x_2)，标签为 Y。如果这两个样本属于同一类别，则 Y=0；如果这两个样本属于不同的类别，则 Y=1。对比损失函数的表达式如式(4-4)所示：

$$L(w,(Y,X_1,X_2)) = \frac{1}{2N}\sum_{n=1}^{N}\left(Yd^2 + (1-Y)\{\max(0, m-d)\}^2\right) \tag{4-4}$$

其中，$d = \| X_1 - X_2 \|_2$ 表示两个样本特征的欧氏距离，m 是阈值。网络的优化目标是使得上面的损失函数 $L(w,(Y,X_1,X_2))$ 最小。

由上述表达式不难看出，对比损失能够很好的表达两个样本点的匹配程度，也能够很好的训练用于提取特征的卷积神经网络。当样本属于同一类别时（即 Y=1），损失函数只有前面一项 d^2，如果同类别的样本点在特征空间内的欧氏距离较大，则损失函数的值也比较大。当样本点不属于同一类别时（即 Y=0），损失函数只剩下后面一项 $\{\max(0, m-d)\}^2$，只有当属于不同类别的两个样本特征的距离小于阈值 m 时，才对网络整体的学习起作用。

4.2.2 卷积神经网络模型

本文用于特征提取的卷积神经网络的结构如表 4-1 所示，该网络是在 VGG16 网络的基础上的改进。把网络的输入调整为 300×300，后面各层的特征图大小也依次发生变化。卷积层的卷积核大小为 3×3，步长和扩充为 1，因此卷积操作不会改变特征图的大小。池化层的步长为 2，扩充为 0，进行最大池化，所以每经过一次池化操作，Block 输出的特征图尺寸都会变为原来的一半。保持网络第 5 个 Block 前面的层不变，在第五个 Block 后面加入若干个新层，组成 Block6。Block6 中包含两个卷积层，卷积核均为 3×3×128 大小，步长和扩充为 1 个像素。第一个卷积层的输入为 Block4 的输出，该层输出特征图大小为 19×19×128。第二个卷积核与 Block5 的输出作卷积操作，输出为一个 10×10×128 的特征图。为了将两个不同大小的特征图进行融合，得到相同维度的输出向量，在 Block6 中每个输出特征图后面分别加入了一个全连接层，这两个全连接层的输入分别为 Block6 中 19×19×128 和 10×10×128 的输出特征图，输出均为一个 256 维的特征向量。之后将这两个特征向量联结为一个 512 维的高维特征，作为网络最后一层损失层的输入。网络中的每个卷积层和全连接层之后均使用 leaky-ReLU 作为激活函数。网络的对比损失层用来对两张输入图像的特征进行相似性度量，输出是一个损失函数的数值，计算方法如式（4-4）所示。

网络中 Block6 设计有两个目的，一是为了提取到网络中不同层次的特征，卷积神经网络中不同的池化层输出的特征图具有不同的分辨率，将这些不同分辨率的特征进行融合能够更加有效地对图像的内容信息进行表达，提高提取到的特征的鲁棒性。另一个目的是极大的减少特征计算的运算量，作为下一章高斯混合模型的输入特征。如果将输入图像分成若干重叠的子块之后，再分别对每个子块利用卷积神经网络提取特征，则需要对网络进行几百次的前向计算，这个过程十分耗时，且需要非常大的存储空间。同时，由于个格子块之间存在重叠，因此提取特征的过程也存在着大量的重复运算。卷积神经网络的特征图与原图像之间存在着对应关系，特征图上每个像素点均对应于输入图像上一个特定大小的区域。Block4 输出的特征图是在原图像上作了 4 次池化操作，特征图上的每个像素点近似对应于原图像上 16×16 的区域；同理，Block5 输出的特征图是在原图像上作了 5 次池化操作，每个像素点近似对应于原图上的 30×30 的区域。这样可以近似的把原图像分割成多个子块，19×19 和 10×10 的特征图分别相当于把原图像分割成 361 和 100 个的子块。同样大小的子块之间不互相重叠，不同大小的子块之间相互重叠。特征图上每个像素点的深度为 128，可以近似的表示对应的图像区域的内容

表 4-1 卷积神经网络的结构细节

<table>
<tr><th></th><th>Conv 3×3</th><th>Conv 3×3</th><th>Conv 3×3</th><th>Pooling</th><th>输出大小</th></tr>
<tr><td>输入</td><td colspan="5">300×300×64</td></tr>
<tr><td>Block 1</td><td>核大小：3
核数量：64
步长：1
扩充：1</td><td>核大小：3
核数量：64
步长：1
扩充：1</td><td></td><td>核大小：2
步长：1
类型：Max</td><td>150×150×64</td></tr>
<tr><td>Block 2</td><td>核大小：3
核数量：128
步长：1
扩充：1</td><td>核大小：3
核数量：128
步长：1
扩充：1</td><td></td><td>核大小：2
步长：1
类型：Max</td><td>75×75×128</td></tr>
<tr><td>Block 3</td><td>核大小：3
核数量：256
步长：1
扩充：1</td><td>核大小：3
核数量：256
步长：1
扩充：1</td><td>核大小：3
核数量:256
步长：1
扩充：1</td><td>核大小：2
步长：1
类型：Max</td><td>38×38×256</td></tr>
<tr><td>Block 4</td><td>核大小：3
核数量：512
步长：1
扩充：1</td><td>核大小：3
核数量：512
步长：1
扩充：1</td><td>核大小：3
核数量:512
步长：1
扩充：1</td><td>核大小：2
步长：1
类型：Max</td><td>19×19×512</td></tr>
<tr><td>Block 5</td><td>核大小：3
核数量：512
步长：1
扩充：1</td><td>核大小：3
核数量：512
步长：1
扩充：1</td><td>核大小：3
核数量:512
步长：1
扩充：1</td><td>核大小：2
步长：1
类型：Max</td><td>10×10×512</td></tr>
<tr><td rowspan="2">Block 6</td><td colspan="2">输入分别为 Block4 和 Block5 的输出</td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2">19×19×128
10×10×128</td></tr>
<tr><td>核大小：3
核数量：128
步长：1
扩充：1</td><td>核大小：3
核数量：128
步长：1
扩充：1</td></tr>
<tr><td>全连接层 1</td><td colspan="4">256 个神经元，输入为 19×19×128 的特征图</td><td>1×1×256</td></tr>
<tr><td>全连接层 2</td><td colspan="4">256 个神经元，输入为 10×10×128 的特征图</td><td>1×1×256</td></tr>
<tr><td colspan="6">对比损失层</td></tr>
</table>

信息。这样，就可以将整幅图像作为卷积神经网络的输入，只经过一次前向运算就能够计算出用来表达各个图像块的特征，极大的减少了特征提取阶段的的时间和空间开销。

4.3 试验设计与结果分析

4.3.1 图片数据集与预处理

表 4-2 LFW 数据库中图片的统计

	总人数	只有一张图片	多于一张图片
	5749 人	4069 人	1680 人
总计		13233 人	

LFW 数据集[57]是一个用于评测无约束条件下人脸识别问题的数据集。其中包含从互联网上搜集 13233 张全世界知名人士的图片，共有 5749 个人。数据库中的人脸图像均处于自然场景中，包含了实际环境中遇到的各种姿势、表情、光照、遮挡、年龄和性别变化。LFW 数据库中图片的统计如表 4-2。

图 4-3 LFW 数据集示意图

我们随机的选择 10000 对样本，其中，属于同一个人的人脸图像对和不同人的图像对各 5000 对。由于原始的图像周围存在太多的背景，对于验证网络来说这

些背景不属于有用的数据部分，我们对每张训练图像进行裁剪，只保留图像中人脸部分的信息。由于 LFW 数据集中人脸区域位于整幅图像的中心，所以只保留每张图像中心点周围 150×150 大小的人脸区域，之后将该区域调整为 300×300 的大小，并对所有的人脸图像进行去均值和归一化处理。

4.3.2 卷积神经网络的训练

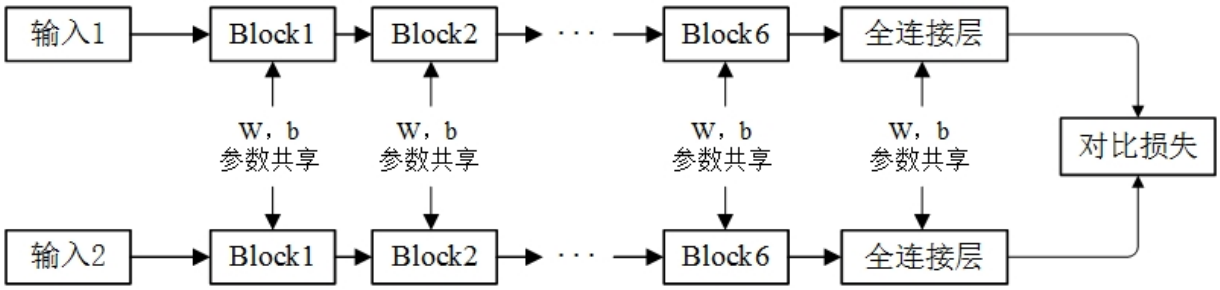


图 4-4 网络训练的结构模型

本文用人脸验证任务作为网络的训练目标，网络的损失函数设置为对比损失函数。首先制作训练数据的标注信息，若两张图像属于一个人，则标签设置为 1，若不属于同一个人，则标签设置为 0。图 4-4 是网络训练的结构模型，两幅图像分别输入到卷积神经网络中进行前向运算，选择网络的最后一层的特征进行对比损失函数计算。

将一对图像作为网络的输入，通过最小化网络的对比损失函数，对网络进行训练。对比损失函数中的的超参数 m 设置为 0.5，两个输入图像的相似度大于 0.5 时，就基本判定两张图片是相似的，不会对网络的训练起作用，小于 0.5 时就判定为不同的人，对网络中的参数进行调整。学习率初始值设置为 0.001，每 5000 次迭代学习率进行一次衰减，衰减为原来的 0.1 倍，动量因子设置为 0.9。把网络中卷积层和全连接层的参数初始化为均值为 0，方差为 0.01 的高斯分布。

损失函数值能够反映出网络的学习状况，网络的 loss 值会随着网络的训练过程逐渐变小，直到收敛到一个很小的值。图 4-5 展示了网络训练过程中 loss 值的走势，可以看到训练刚开始时，loss 值下降的较明显，第 6 个 epoch 到第 14 个 epoch 左右 loss 值下降趋势放缓，第 14 个 epoch 之后网络的 loss 值趋于稳定，18 个 epoch 之后接近收敛。训练过程中没有遇到梯度弥散（loss 值不变化）或梯度消失（loss 值突然很大）的现象。

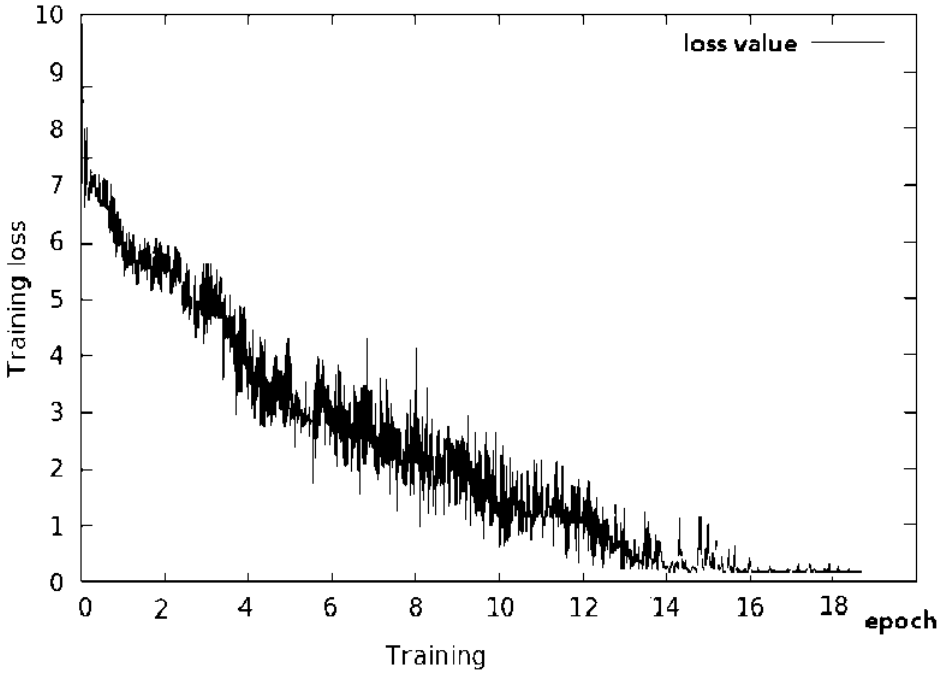


图 4-5 网络训练集的 loss 值走势图

4.3.3 训练结果分析

训练好的卷积神经网络可以作为一个人脸特征的提取器，提取对光照，表情，姿态等鲁棒的人脸内容信息，用于后续的人脸验证阶段。我们在 LFW 数据集中选择 500 对图像作为测试集，首先将测试集中的图片裁剪到 150×150 的大小，之后对图像进行对齐处理。人脸对齐处理中，我们使用 CFSS[66]算法来定位人脸图像的关键点位置，利用检测到的关键点和模板的关键点计算仿射矩阵，之后利用仿射矩阵计算对齐后的人脸图像，如图 4-6 所示。

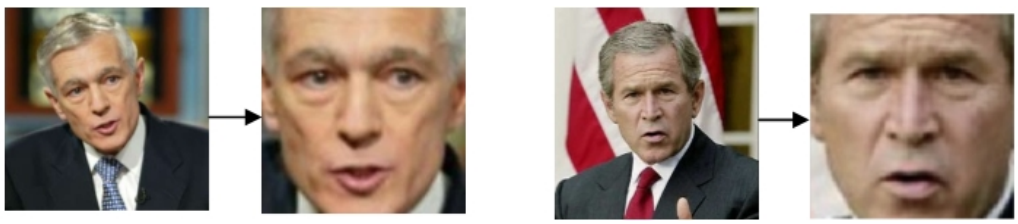


图 4-6 裁剪和对齐处理后的人脸图像

测试图像依次经过卷积神经网络进行前向运算，网络的输出为其对应的特征表示，即联结两个全连接层的输出作为图像的特征向量，共 256 维。图 4-7 可视化了卷积神经网络 Block6 中池化层的输出特征图，左侧为原始的人脸图像，中间和右侧分别为 Block6 中大小为 19×19 和 10×10 的特征图。

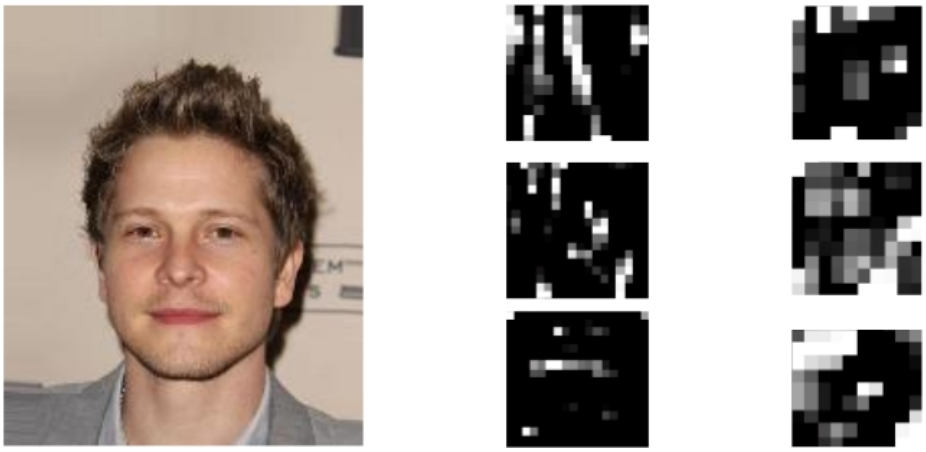


图 4-7 网络中部分特征图可视化

我们对网络提取到的特征在测试集上进行测试，实验中我们对比了传统的特征和卷积神经网络提取的特征对人脸验证结果的影响。我们将 50×150 大小的原始图像分为 10×10 个不重叠的小块，在每一块上分别计算 Dense SIFT[58]，LBP[52]，LGBPHS[29] 和 WPCA-POEM[59]特征。由于得到的特征维度较高，之后我们用 PCA 对其进行降维处理。表 4-2 中分别展示了用于测试的特征维度和测试结果。测试阶段，分别计算测试集中每对图像特征向量之间的欧式距离，若小于 0.5 则认为是同一个人，大于则认为是不同人。实验中选择了三种特征算子与本文使用的深度特征进行对比，可以看到，改进的 VGG16 虽然只提取到了 256 维特征，但取得了十分优异的结果。这是由于深度特征对人脸图像的超完备表达，并具有对复杂干扰的鲁棒性。传统的手工特征仅针对对某些特定的干扰设计，而非可控条件下的图像中干扰因素十分多样且复杂，所以手工设计的特征显得力不从心。

表 4-2 LFW 数据集上测试结果对比

特征算子	平均准确率	特征维度
Dense SIFT[58]+LBP[52]	90.00%	10000
LGBPHS[29]	92.25%	2000
WPCA-POEM[59]	94.30%	900
本文使用的深度特征	98.80%	256

4.4 本章小结

本章详细介绍了用于特征提取的网络结构细节，该网络能够对人脸图像提供一个超完备且鲁邦的特征表达。本章的开始，首先介绍了常用的手工特征提取算子，之后介绍了我们改进的网络结构模型，并介绍了网络训练的损失函数和训练

细节，之后在 LFW 数据集上对提取到的特征进行初步的评估，得到了远胜于传统特征的结果。为了进一步得到对复杂干扰更加鲁棒的特征，以提升人脸识别的准确度，我们在将在下一章中详细介绍本文使用的人脸特征匹配方法，该方法利用了本章中深度卷积神经网络提取的特征，是本章中改进网络的应用。

第五章 人脸特征匹配方法研究

尽管已经提取到了鲁棒的人脸特征表达，但是人脸图像往往包含姿势、表情等变化，这些干扰是高度非线性的且会一直混杂在提取到的特征中。所以，我们希望能够找到合适特征匹配方法，来减少这些干扰对人脸识别任务的影响。这一章，我们首先介绍几种人脸特征匹配方法，之后针对上述问题提出本文使用的特征匹配方法，最后对该方法进行评估。

5.1 人脸特征匹配方法

人脸特征匹配是通过判断两幅图像特征的相似度，来判断两张人脸图像是否来源于同一个个体。作为人脸识别任务中的重要步骤，选择何种匹配算法来判断人脸特征之间的相似性，既能够排除复杂干扰对人脸验证的影响，又能够对这些特征进行最大程度的分类，是影响识别结果好坏的一个关键性问题。

5.1.1 相似性度量方法

常用的特征相似性度量方法是基于距离的度量方法，即比较两个特征向量之间的距离与给定阈值的大小。若大于该阈值，则判断两个特征不属于同一类别，若小于该阈值，则属于同一类别。一种常用的距离度量方法是平方马氏距离，其定义如式(5-1)所示：

$$d_M(x,t)=(x-t)^T M(x-t) \tag{5-1}$$

其中 M 是一个半正定矩阵，$x,t\in \mathrm{R}^d$，为两个待验证的特征向量，维数为 d 维。

在实际的人脸验证过程中，直接使用距离度量方法的性能相比较而言比较一般，所以很少使用。取而代之的是相似性学习方法，通过学习一个相似性度量函数 $S_M(x,t)=x^T Mt$，来对相似性进行度量。

此外，另一种人脸验证中常用的度量方法是余弦相似度(Cosine Similarity)。计算两个人脸特征向量夹角的余弦值，使用该值作为两张图像之间相似度的衡量，余弦相似度定义为式(5-2)的形式：

$$CS_M(x,t)=\frac{x^T Mt}{\sqrt{x^T Mx}\sqrt{t^T Mt}} \tag{5-2}$$

5.1.2 联合贝叶斯

采集到的人脸图像中往往存在姿态、光照、表情以及遮挡等问题，这些干扰

因素会在不同程度上影响人脸的特征表达，导致后面的验证结果产生较大的偏差。传统距离度量方法，如欧式距离，余弦距离等，无法很好的应对人脸特征中复杂干扰的影响，因此很难准确地反映两张人脸图像之间的相似性。尤其在与浅层的手工设计特征（如 Gabor 或 LBP）结合时，结果通常很不理想，甚至会出现相同身份的距离会远大于不同身份的距离的情况。

联合贝叶斯[60]作为一种经典的人脸验证方法，其主要思想是对两个人脸图像的特征表示进行联合建模，并在建模过程中加入人脸的先验知识，之后在进行相似性判定。

把两张人脸图像分别表示为 x_1 和 x_2，经典的贝叶斯人脸识别方法利用式(5-3)所示的似然比来判断是否属于同一个人：

$$r(x_1,x_2)=\log\frac{P(\Delta\mid H_I)}{P(\Delta\mid H_E)} \tag{5-3}$$

其中，H_I 表示两张人脸图像属于同一个人，H_E 表示属于不同的人，$\Delta=x_1-x_2$ 表示两个特征向量之间的差异。若 $r(x_1,x_2)>0$，则 $P(\Delta\mid H_I)>P(\Delta\mid H_E)$，判断 x_1,x_2 属于同一个人；反之 $r(x_1,x_2)<0$，则 $P(\Delta\mid H_I)<P(\Delta\mid H_E)$，判断属于不同人。经典的贝叶斯方法是基于人脸图像对之间的差异进行建模，但是，当式(5-3)中的两个概率同时取值较大或较小时，通过上面的公式很难对图像是否属于同一个人进行正确的判定。

受到经典贝叶斯方法的启发，对人脸图像 x_1 和 x_2 进行联合建模，此时 $r(x_1,x_2)$ 表示为式(5-4)：

$$r(x_1,x_2)=\log\frac{P(x_1,x_2\mid H_I)}{P(x_1,x_2\mid H_E)} \tag{5-4}$$

为了得到更加精确的模型，并引入人脸表示的先验知识，把人脸表示为两个独立随机变量的和，如式(5-5)所示：

$$x=\mu+\varepsilon \tag{5-5}$$

其中 x 是减去均值后的人脸图像，μ 表示人脸的身份信息，ε 表示同身份的人脸图像中自身的差异（如表情、姿势、光照等），如图 5-1 所示。μ 和 ε 分别服从式(5-6)和式(5-7)所示的两个高斯分布：

$$\mu=N(0,S_\mu) \tag{5-6}$$

$$\varepsilon=N(0,S_\varepsilon) \tag{5-7}$$

其中，S_μ和S_ε为两个协方差矩阵。由这个先验条件可知，两个人脸(x_1,x_2)的

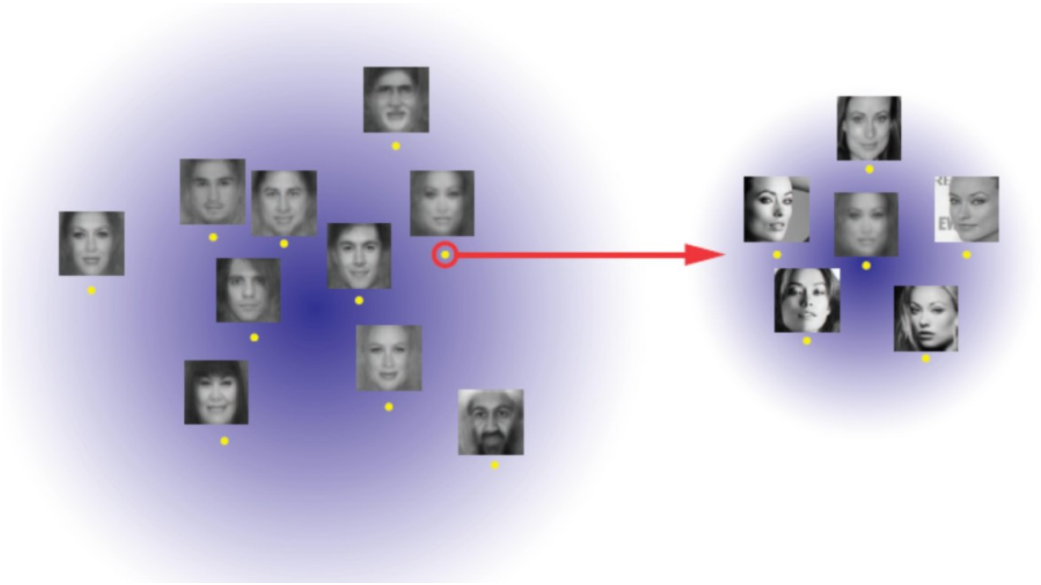


图 5-1 左图为不同身份的人脸图像，右图为人脸内的差异[60]

联合分布也服从均值为 0 的高斯分布。由于随机变量μ和ε相互独立，所以x_1,x_2之间的协方差可表示为式(5-8)：

$$\mathrm{cov}(x_i,x_j)=\mathrm{cov}(\mu_i,\mu_j)+\mathrm{cov}(\varepsilon_i,\varepsilon_j) \tag{5-8}$$

在H_I假设中，身份变量μ_1,μ_2相同，差异变量$\varepsilon_1,\varepsilon_2$相互独立，联合概率$P(x_1,x_2\mid H_I)$的协方差矩阵为式(5-9)的形式：

$$\Sigma_I=\begin{bmatrix} S_\mu+S_\varepsilon & S_\mu \\ S_\mu & S_\mu+S_\varepsilon \end{bmatrix} \tag{5-9}$$

在H_E假设中，身份变量μ_1,μ_2和差异变量$\varepsilon_1,\varepsilon_2$都是相互独立的，联合概率$P(x_1,x_2\mid H_E)$的协方差矩阵为式(5-10)的形式：

$$\Sigma_E=\begin{bmatrix} S_\mu+S_\varepsilon & 0 \\ 0 & S_\mu+S_\varepsilon \end{bmatrix} \tag{5-10}$$

以上的两个条件联合概率中的协方差矩阵S_μ和S_ε可以通过期望最大化(EM)算法[62]来进行学习。EM 算法分为两步，第一步为求期望，通过已知的协方差矩阵S_μ和S_ε求μ和ε；第二步为期望的最大化步骤，根据上一步中的到的μ和ε值更新参数S_μ和S_ε：

$$S_\mu=\mathrm{cov}(\mu) \tag{5-11}$$

$$S_\varepsilon=\mathrm{cov}(\varepsilon) \tag{5-12}$$

最终的似然比如式(5-13)所示：

$$r(x_1,x_2)=\log\frac{P(x_1,x_2\mid H_I)}{P(x_1,x_2\mid H_E)}=x_1^TAx_1+x_2^TAx_2-2x_1^TGx_2 \tag{5-13}$$

其中的 A 和 G 可通过下面的式(5-14)和式(5-15)求得：

$$A=(S_\mu+S_\varepsilon)^{-1}-(F+G) \tag{5-14}$$

$$\begin{pmatrix}F+G & G\\ G & F+G\end{pmatrix}=\begin{pmatrix}S_\mu+S_\varepsilon & S_\mu\\ S_\mu & S_\mu+S_\varepsilon\end{pmatrix}^{-1} \tag{5-15}$$

本文在最后的特征相似性度量阶段，分别使用相似性度量的方法和联合贝叶斯对最终得到的人脸特征进行判定，并通过实验分析两种方法对最终验证结果的影响。

5.2 基于高斯混合模型的人脸识别

非可控条件下人脸图像中往往包含强烈的的复杂干扰，尤其是姿势、表情等非线性干扰，很难在图像预处理步骤去除其影响。为了降低干扰因素的影响，本文在进行特征的相似性度量之前，首先使用高斯混合模型对人脸图像进行建模，以提高识别的准确度。高斯混合模型中的各个高斯分量能够隐式的对人脸图像子块进行表达，从而选择出待验证图像对中的相关区域。如图 5-2 所示，两张待验证的人脸图像被分为若干子块，每个子块包含人脸的一个区域，高斯混合模型能够找出属于同一个高斯分量的两个子块进行匹配，这两个子块往往包含相似的内容信息，如眼睛区域和眼睛区域进行对应，鼻子区域和鼻子区域进行对应。若某个子块受到较大的干扰因素影响，呈现出了不同的姿态和纹理（如正面的鼻子和侧面的鼻子），则这两个子块将会对应于不同的高斯分量，即属于不同的类别，不会选择此区域用来判断图像的相似性，因此能够较好地去除姿态表情等对人脸验证的影响。

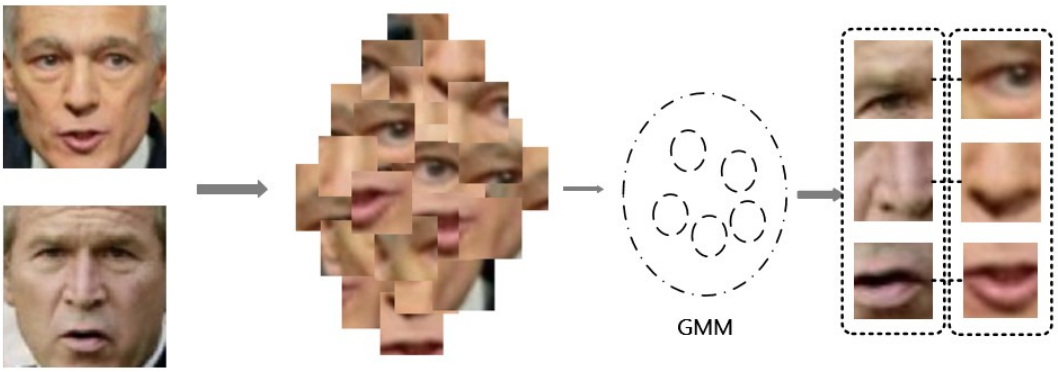


图 5-2 基于高斯混合模型的人脸验证算法流程图

本文提出的基于高斯混合模型的方法分为两个阶段：高斯混合模型模型训练阶段和人脸验证阶段。训练阶段中，首先利用第四章训练好的卷积神经网络对训练数据集中的人脸图像进行特征提取，之后利用该特征对高斯混合模型进行训练，估计模型中参数值，从而得到对整张人脸图像进行建模的高斯混合模型。之后是人脸验证阶段，第一步同样是利用卷积神经网络提取两张待验证人脸图像的特征，用已经建立好的高斯混合模型为每个特征寻找响应度最高的高斯分模型。之后把两张图像中属于同一个高斯分量的特征进行匹配，并将两张图像中所有匹配的特征进行级联形成一个高维特征。最后用判别函数对级联的高维特征进行判断，输出验证结果。基于高斯混合模型（GMM）的人脸验证算法的流程图如图 5-3 所示。

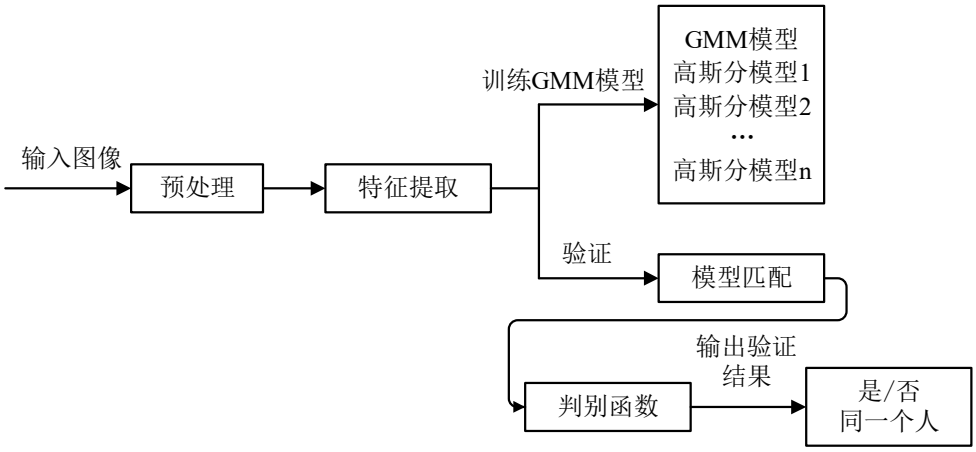


图 5-3 基于高斯混合模型的人脸验证算法流程图

5.2.1 高斯混合模型和 EM 算法

高斯混合模型[61](Gaussian Mixture Model,GMM)，是一种广泛使用的聚类算法，在图像分割、语音识别、视频分析等方面均有应用，并取得很可观的效果。高斯混合模型表示为一个多维的概率密度函数，具有如式(5-16)所示的表达式：

$$p(Y|\theta)=\sum_{k=1}^{K}\alpha_k\phi(Y|\theta_k) \tag{5-16}$$

其中，Y 是一个 D 维的随机变量；α_k 是每个分高斯分量的权重，且满足 $\alpha_k \geq 0,\ \sum_{k=1}^{K}\alpha_k=1$；$\phi(Y|\theta_k)$ 是高斯概率密度函数，其表示为式(5-17)，

$$\phi(Y|\theta_k)=\frac{1}{(2\pi)^{D/2}|\sum_k|^{1/2}}\exp\left(-\frac{1}{2}(Y-\mu_k)^T\sum_k^{-1}(Y-\mu_k)\right) \tag{5-17}$$

称为第 k 个分模型，其中 μ_k 为均值向量，$\sum_k$ 为协方差矩阵。

由上述公式可知，一个高斯混合模型是一个概率分布，由均值向量，协方差矩阵和高斯分量的权重这三个参数进行表示，即 $\theta=\{\alpha,\mu,\sum\}$。这三个参数唯一的

确定一个高斯混合模型。模型的训练就是在某种准则下获得模型参数的过程。

极大似然估计是一种常用的参数估计方法，得到的参数能够在最大程度上描述训练数据的分布。在给定训练数据 Y 后，找到使得高斯混合模型似然函数取最大值时的模型参数θ，将该参数作为高斯混合模型中的参数。一个 D 维的训练数据$Y=(y_1,y_2,...,y_D)$，似然可以表示为式(5-18)：

$$P(Y|\theta)=\prod_{i=1}^{D}P(Y_i|\theta) \tag{5-18}$$

使得$P(Y|\theta)$最大的参数θ即为所求，即式(5-19)：

$$\hat{\theta}=\arg\max_{\theta}P(Y|\theta) \tag{5-19}$$

由于模型参数和似然函数间存在复杂的非线性关系，无法直接对上式进行求解，这里选择期望最大化(EM)算法[62]来估计高斯混合模型中的参数。EM 算法是一种迭代求解的算法，每次迭代中进行两步：E 步和 M 步。E 步计算 Q 函数，M 步计算使得 Q 函数取最极大值时的参数θ。

Q 函数定义为训练数据的对数似然函数的期望，如式(5-20)所示：

$$Q(\theta,\theta')=\sum_{k=1}^{K}P(Y,k|\theta)\log P(Y,k|\theta) \tag{5-20}$$

其中，k 是隐藏状态，是随机且未知的。Q 函数的极大值近似的逼近最大的对数似然度。

求解参数估计值的步骤如下：

E 步：根据当前模型的参数，计算分模型 k 对训练数据y_j的响应，如式(5-21)所示：

$$\hat{\gamma}_{jk}=\frac{\alpha_k\phi(y_j|\theta_k)}{\sum_{k=1}^{K}\alpha_k\phi(y_j|\theta_k)},\quad j=1,2,...,N;\ \text{k=1,2,...,K} \tag{5-21}$$

M 步：Q 函数分别对三个参数求导，得到相应的估计值作为新一轮迭代的模型参数，公式如(5-22)-(5-24)所示：

$$\mu_k=\frac{\sum_{j=1}^{N}\hat{\gamma}_{jk}\,y_j}{\sum_{j=1}^{N}\hat{\gamma}_{jk}},\quad \text{k=1,2,...,K} \tag{5-22}$$

$$\sigma_k^2=\frac{\sum_{j=1}^{N}\hat{\gamma}_{jk}(y_j-\mu_k)^2}{\sum_{j=1}^{N}\hat{\gamma}_{jk}},\quad \text{k=1,2,...,K} \tag{5-23}$$

$$\alpha_k = \frac{\sum_{j=1}^{N} \hat{\gamma}_{jk}}{N}, \quad \text{k=1,2,...,K} \tag{5-24}$$

迭代的重复 E 步和 M 步，直到收敛。

5.2.2 构建人脸图像的高斯混合模型

首先，将每张人脸图像划分为重叠的 461(19×19+10×10)个子块，每个子块覆盖了人脸图像中的部分区域。把待检测的人脸图像输入到训练好的卷积神经网络中，网络结构在上一章提出。卷积神经网络中，Block4 和 Block5 输出的特征图在不同分辨率上对人脸图像进行了表达，特征图上每个像素点近似的与人脸图像上的每个子块相对应，每个像素点的维度为 128 维，即每个图像子块的内容信息可以由该 128 维的向量描述。训练数据集中包含的所有子块的特征组成一个特征集合 $a=\{a_1,a_2,...,a_m\}$，其中 m 为训练数据集中子块的个数。同时，分别为每个图像块提取位置信息 $l_i=[x_i,y_i]$，加入到内容信息之后，得到的 130 维的特征向量集合 $f=\{f_1,f_2,...,f_m\}$，用来描述每个图像块。

利用提取好的特征训练高斯混合模型，模型中的分量隐式的表达人脸图像中的某些子块。由于人脸的五官位置具有相对性，加入位置坐标能够对人脸图像上大致属于相同的分量的子块进行约束，使其不会偏移到太远的区域，如模型中的第 52 个分量大致对应于训练数据集中所有包含鼻子的子块。

由于图像的内容信息为 128 维，位置信息只有 2 维，两者悬殊的差距会导致位置约束变得非常弱。为了平衡内容和位置信息对高斯混合模型的影响，我们把模型中高斯分量的协方差矩阵限制为对角阵。使得图像的内容信息和位置约束处于平等地位的同时，还能够降低计算协方差矩阵时的运算量。用高斯混合模型对特征集合进行建模，如式(5-25)所示：

$$P(\mathrm{f}\,|\,\theta) = \sum_{k=1}^{K} \alpha_k \phi(\mathrm{f}\,|\,\mu_k, \Sigma_{\mathrm{k}}) \tag{5-25}$$

其中 $\theta = (\alpha_k, \mu_k, \Sigma_{\mathrm{k}})$ 分别表示第 k 个高斯成分的权重，均值和协方差矩阵；$\Sigma_{\mathrm{k}} = \sigma_k^2 \mathrm{I}$，I 为单位矩阵。最大化似然函数来对模型中的参数进行估计，似然函数如式(5-26)所示：

$$\begin{aligned} L(\theta) &= \log[\prod_{i=1}^{\mathrm{N}} P(\mathrm{f}_i|\theta)] \\ &= \sum_{i=1}^{N} \log[\sum_{k=1}^{K} \alpha_k \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(\mathrm{f}_i - \mu_k)^2}{2\sigma_k^2}\right)] \end{aligned} \tag{5-26}$$

其中，f_i 表示训练数据集中第 i 个图像块的特征。使似然函数取得最大值的参

数 $\hat{\theta}=\arg\max_{\theta} L(\mathrm{f}\,|\,\theta)$，即为模型的参数。

由于上式无闭式表达，无法直接进行求解。使用期望最大化（EM）算法来对模型中的参数进行估计：E 步计算 f_i 属于每个高斯分量的概率；M 步最大化对数似然函数的期望值来对参数进行更新。具体步骤如下：

E 步：计算训练数据 f_i 由第 k 个分模型产生的后验概率 $P(k\,|\,\mathrm{f}_i)$，如式(5-27)所示：

$$P(k\,|\,\mathrm{f}_i)=\frac{\alpha_k\phi(\mathrm{f}_i\,|\mu_k,\sigma_k^2 I)}{\sum_{k'=1}^{K}\alpha_k\phi(\mathrm{f}_i\,|\mu_{k'},\sigma_{k'}^2 I)} \tag{5-27}$$

之后计算第 k 个模型生成的特征个数 n_k，式(5-28)：

$$\mathrm{n}_k=\sum_{i=1}^{N}P(k\,|\,\mathrm{f}_i) \tag{5-28}$$

M 步：计算高斯混合模型中更新后的权重，均值和协方差矩阵，式(5-29)-式(5-31)：

$$\hat{\alpha}_k=\frac{n_k}{N} \tag{5-29}$$

$$\hat{\mu}_k=\frac{\sum_{j=1}^{N}P(k\,|\,\mathrm{f}_j)\mathrm{f}_i}{n_k} \tag{5-30}$$

$$\hat{\sigma}_k^2=\frac{\sum_{i=1}^{N}P(k\,|\,\mathrm{f}_i)(\mathrm{f}_i-\hat{\mu}_k)^2}{n_k} \tag{5-31}$$

重复的进行 E 步和 M 步，直到模型收敛，即可得到用于人脸验证的高斯混合模型。

5.2.3 人脸验证

测试阶段，输入为两张待验证的人脸图像 A 和 B。首先是图像的预处理操作，进行 2D 人脸对齐。具体的做法是：定位两张图像中眼睛的位置，之后利用双眼的相对位置对图像进行旋转，缩放，最后裁剪到 300×300 的大小。将整张人脸图像作为输入，输入到卷积神经网络中，网络的最后两个池化层输出用来表达每个子块的 128 维特征。提取到子块的内容信息后，加入图像块中心点坐标作为该图像块的位置信息。A，B 两幅图像上提取到的特征集合分别记为：$\mathrm{f}_\mathrm{A}=\{\mathrm{f}_{A1},\mathrm{f}_{A2},...,\mathrm{f}_{Am}\}$ 和 $\mathrm{f}_\mathrm{B}=\{\mathrm{f}_{B1},\mathrm{f}_{B2},...,\mathrm{f}_{Bm}\}$，m 为图像中包含的块数。

训练好的高斯混合模型中的每个高斯分量 $(\alpha_k,\phi(\mu_k,\sigma_k^2\mathrm{I}))$，都隐式的表示了一个特征集合中的特征，即每个高斯分量选择概率最大的图像块，属于该高斯分量

所代表的类别，如式（5-32）所示：

$$\phi_{\mathrm{k}}(F)=\arg\ \max_{i}\alpha_k\phi(\mu_k,\sigma_k^2\mathrm{I}) \tag{5-32}$$

由于在特征中加入了图像块的位置信息，可以使得属于每一类的图像块的大致位置比较集中，不会偏移太远。

对于 A，B 两幅图像中属于同一类别（即同一个高斯分量）的若干个图像块，选择其中概率值最大的两块进行对应，我们认为这两块中包含的内容信息最为相似。这样做的好处是，能够在待验证人脸图像对中可以选择出最相关的区域，复杂干扰在这些相关区域内的影响较低。如果属于某个高斯分量类别的图像块的概率小于 0.7，则丢弃该高斯分量选择的区域，不计入到最后的图像总特征中。这样做的原因是，由于姿势，表情等的复杂因素存在，并不是所有的图像块都能很好的匹配高斯混合模型中的分量。如果一个图像块以最高的概率属于某个高斯分量，但它的概率小于 0.7，则说明该图像中的所有子块对这个高斯分量的响应都比较低，用该分量选择出来的结果进行匹配，误差会比较大，从而影响后面识别的准确度。这样做的另一个好处是，一副图像中大概有五分之三的子块对所有高斯分量的响应均较低，把这些子块排除掉后，余下的特征数量较少，从而降低相似性度量阶段的计算量。

将上述方法得到的匹配区域按照高斯分量的类别顺序进行排列，之后级联在一起形成一个高维的特征，该特征即为用于验证的两张人脸图像的全局特征。最后对两张图像进行特征的相似性度量，5.4 节中我们分别比较了使用欧氏距离，余弦相似度和联合贝叶斯方法的识别结果。

5.3 试验设计与结果分析

5.3.1 数据预处理

本章选择 LFW 数据库作为训练数据集，选择其中的 5000 张人脸图像用于高斯混合模型的训练。首先对每张训练的人脸图像进行裁剪，去除原始图像周围的背景信息，只保留每张图像中心点周围 150×150 大小的人脸区域。之后将人脸图像的大小调整至 300×300，并对所有的图像进行去均值和归一化处理，输入到卷积神经网络中提取特征。

5.3.2 高斯混合模型的训练

高斯混合模型开始训练之前，首先需要确定模型中高斯分量的数量以及模型的初始参数值θ。高斯分量的个数 K 会在很大程度上影响识别的准确度，如果 K 设置的过大，会导致模型变得复杂，模型中参数成倍的增加，收敛的速度变慢，

训练模型的时间和空间开销成倍的增加。同时，如果训练数据不够充足，会导致训练过程中模型的参数难以收敛，影响模型的参数估计，降低算法的识别性能。而 K 太小则会导致模型对于特征的分类不够精细，无法有效的排出复杂干扰的影响，同样也会降低算法的识别性能。通常情况下，K 取 2 的整数次幂，如 64，128，256 等，这里由于我们在每张人脸图像上提取 461 个特征向量，所以本文中选择超参数 K =512。

对于模型的初始参数θ，通常可以通过随机设置法或者聚类的方法来获得。随机设置法是在训练数据中随机选择几个特征向量作为模型初始参数，虽然随机法操作起来简单，然而随机设置会导致高斯混合模型训练时的迭代次数过多，收敛速度变慢。为了给高斯混合模型的赋予一个易收敛的初始值，加快模型的训练速度，本文首先利用 K-means 算法对特征进行聚类。将 K-means 算法得到的每类的类别中心，作为高斯混合模型中每个分量的初始均值，即μ_0。利用属于每个类别的样本点，分别计算高斯分量协方差矩阵Σ_0和系数α_0的初始值。具体计算方法如下式(5-33)和式(5-34)所示：

$$\Sigma_i = \mathrm{cov}(FF^T) \tag{5-33}$$

$$\alpha_i = \frac{n_i}{N} \tag{5-34}$$

其中F为属于每个类别的样本点组成的矩阵，N为样本点的总个数，n_i为属于类别 i 的样本点数。由于高斯混合模型中协方差矩阵为对角阵，故本文选择Σ_i对角线上的元素作为Σ_0，并加入一个小随机数，增加其数值不稳定性。

在使用 EM 算法进行迭代训练的过程中，如果某次迭代模型参数中的协方差矩阵出现接近零的值，则会导致似然函数的误差变得非常大，而这种影响将会一直存在于模型训练的参数更新过程中，从而导致训练不收敛。因此我们为协方差矩阵对角线上的所有元素都加上一个很小的数值，来避免出现此现象，本文实验中将此数值设置为 0.00001。

当似然函数增长很小时训练结束，即当前迭代步骤的似然函数和上次迭代步骤中的似然函数的比值小于一个给定阈值时，停止迭代，此时得到的参数θ即为最优解。本文把该阈值设定为$10\times e^{-11}$。

5.3.3 人脸识别实验结果

在 LFW 数据集上对本文的方法进行评估，选择 LFW 数据集中的 500 对图像进行验证。为了计算特征向量之间的相似程度，分别计算两个特征向量的欧式距离和余弦相似度，我们还使用了联合贝叶斯的方法对特征进行分类，判断是否属于同一个人。表 5-1 中比较了这三种方法的识别结果。

这三种方法使用的特征均为本章方法所提取的特征，从表中可以看出，余弦相似度比欧氏距离结果略好，两种方法结果总体相差不大；联合贝叶斯的识别准确率要明显好于这两种方法。这是由于联合贝叶斯能够进一步对人脸图像中存在的复杂干扰进行分离。因此，在本文后面的实验中，我们采用联合贝叶斯的方法对两个特征向量进行相似性度量。图 5-4 展示了使用上述三种方法进行人脸验证的 ROC 曲线，横坐标表示预测为同一个人脸但实际上是不同人脸的样本在所有负样本中的比例，纵坐标表示预测为同一个人脸且实际也是同一个人脸的样本在所有正样本中的比例。

表 5-1 相似性度量算法比较

相似性度量方法	匹配正确的样本对数	识别正确率
欧氏距离	462	92.40%
余弦相似度	467	93.40%
联合贝叶斯[60]	485	97.00%

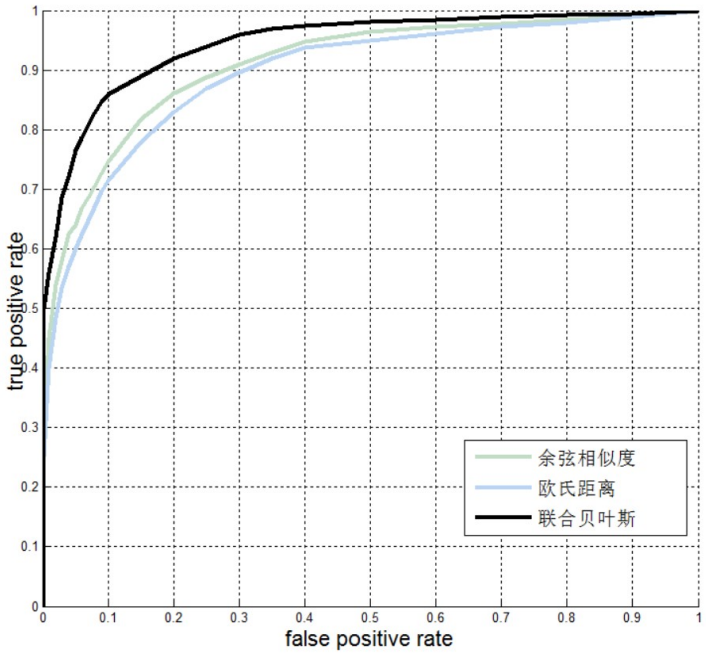


图 5-4 LFW 数据集上的 ROC 曲线

表 5-2 LFW 数据集测试结果

算法	准确度
联合贝叶斯[60]	91.10%
Fisher Vector Faces[63]	93.00%
Hierarchical-PEP[64]	92.40%
CNN 3DMM[65]	92.35%
DeepID[34]	97.25%
本文方法	98.10%

表 5-2 展示了本文的特征提取方法和联合贝叶斯结合的识别结果与其他方法之间的对比结果，能够看到本文的结果在准确性上与其他经典算法有更良好的表现。其中 Hierarchical-PEP[64]方法中用 SIFT 和 LBP 特征训练了一个 3 层的高斯混合模型（即对每个图像子块继续进行划分，在子块上训练一个高斯混合模型），在 LFW 上获得了 91.10%的识别准确度。我们的方法利用卷积神经网络提取的特征仅训练了单层的高斯混合模型，取得了 98.10%的检测精确度。这得益于深度网络特征能够对存在复杂干扰的人脸图像进行很好的表达，提取出更加鲁棒的特征。除此之外，本文还与经典的联合贝叶斯[60]以及 Fisher Vector Faces[63]方法做了比较，这两种方法中均使用了手工设计的浅层特征表达，并采用一些规则分离人脸身份信息与混在其中的噪声，但模型依旧对复杂干扰的影响较敏感，因此结果并不十

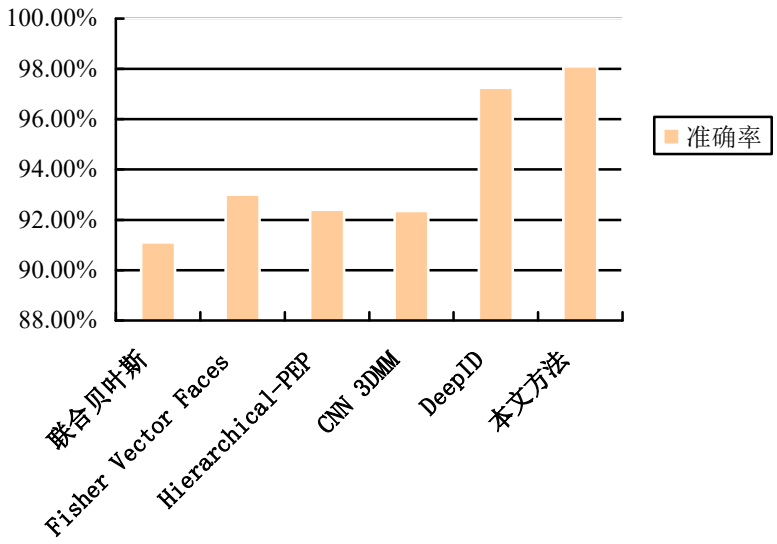


图 5-5 LFW 数据集测试结果对比图

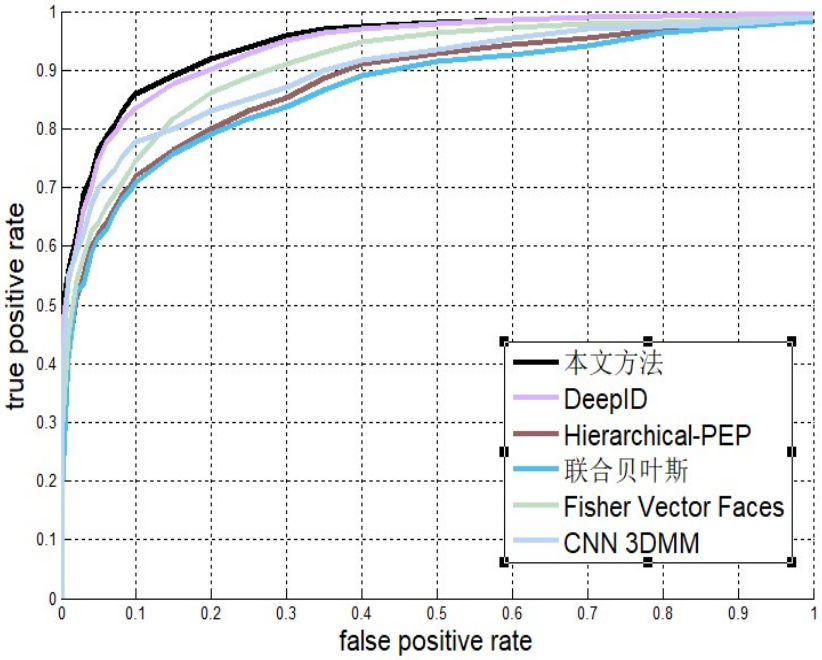


图 5-6 LFW 数据集上的人脸验证 ROC 曲线

分尽如人意。CNN 3DMM[65]通过 3D 的图像变换，对姿势表情等进行处理，将人脸规范化到正面姿态自然表情，以此来降低复杂干扰的影响，然而由于人脸重建过程中会导致大量的有用信息丢失，因此该方法的效果平平。最后，我们还与经典的基于卷积神经网络的识别方法 DeepID[34]进行了比较，DeepID 中使用的训练数据集规模很大，且做了数据扩增，其准确度很大程度上来源于海量的训练数据。相比之下，我们的算法在较小的训练数据集上取得了更好的结果。图 5-5 和图 5-6 分别展示了这几种方法在 LFW 数据集上的结果对比图以及人脸验证的 ROC 曲线。

5.4 本章小结

本章首先介绍了几种人脸特征相似性度量的方法，之后详细介绍了高斯混合模型的训练和预测过程，利用高斯混合模型对提取到的图像特征进行选择，选择出对复杂干扰更加鲁棒的特征用于后面的识别任务。将深度卷积神经网络提取的特征与高斯混合模型进行结合，能够取得比用传统特征训练的高斯混合模型更好的识别结果。通过 LFW 数据集上的实验结果可以看出，本文的提出的识别方法能够实现很好的识别结果。

第六章 全文总结与展望

6.1 全文总结

本文旨在通过将深度学习的相关技术应用到人脸检测和识别的任务中，解决复杂场景下各种因素对人脸检测和识别结果的影响，着重对人脸识别系统中人脸检测、图像特征提取以及特征匹配这三个主要的环节进行分析研究，主要贡献和创新点如下：

1. 在人脸检测阶段，本文对基于 YOLO 模型的目标检测算法进行改进，使之应用到人脸检测任务上来。主要针对人脸检测问题中速度和准确度的权衡问题进行研究，对用于预测的卷积神经网络结构进行改进，使用深度可分离卷积单元代替 VGG16 网络中的传统卷积单元。虽然网络结构变深层数增多，但参数数量仅为原来的 1/9，极大的减少了网络的规模。同时，这种对网络规模的缩减并没有使检测结果变差，实现了提升检测速度的同时不损失检测的准确度。

2. 提出了把深度卷积神经网络特征和高斯混合模型结合起来的识别方法。解决非可控条件人脸识别中的复杂干扰的问题，并在 LFW 数据集上取得有竞争力的结果。

3. 设计一种卷积神经网络模型，利用特征图上的每个像素点与输入人脸图像上的子块进行对应，隐式的把人脸图像分为多个不同尺度的子块，用于后续的特征匹配阶段。之后用人脸的验证信息对特征提取的网络进行训练，在 LFW 数据集上取得了相比传统特征更优异的结果。

4.用高斯混合模型中的分量来对人脸图像中的各个子块进行隐式的归类，寻找属于同一个类别的人脸子块进行匹配，能够在很大程度上减少姿态，表情等高度非线性变化对人脸识别结果的影响。同时在人脸子块的内容信息之后加其位置坐标，能够对各个子块的位置起到一定的限制作用，使得属于同一类别的子块不会偏移太远。考虑到位置信息和内容信息的维度相差较大，混合高斯成分中采用球状高斯分量，在解决此问题的同时也极大的减少了计算量。

6.2 后续工作展望

随着人工智能的迅速发展，各种技术不断推陈出新，也为我们的日常生活带来更多的便利。深度学习作为其中的一个重要研究方向，在实际应用中代替了很多传统的机器学习方法，并取得非常客观的结果。本文针对计算机视觉领域的两大热门的研究方向展开研究，主要研究人脸检测和识别问题，但由于水平和时间

有限，仍有很多问题需要进一步解决优化，主要包括以下几点：

1、基于深度学习的人脸检测问题都会面临速度和准确度的权衡，如何设计网络结构使得在实现实时检测的过程中进一步提升检测的准确度，是未来的一个研究方向。

2、卷积神经网络在实际训练过程中，由于其参数数目多，网络结构深，常常需要大量的训练数据，所以如何获取大量的标定数据是一个问题。此外卷积神经网络也需要很好的硬件支持，为了能在一些移动设备中应用，就需要对网络结构进行优化，使得网络规模降低，在实现缩减网络计算量同时很好的完成特征提取的工作。

3、自然条件下的人脸图像包含各种形式的复杂干扰，且这些干扰越来越多变，如何完善识别算法，使之能够最大程度上的减轻这些干扰对识别结果的影响，同时提升识别的速度，是未来需要进一步研究的问题。

致谢

时光飞逝，三年的研究生生活即将接近尾声。在这三年中，我得到了很多老师，家人和朋友的关心和帮助，并留下了很多非常美好的回忆，在这里我要向所有在我攻读硕士学位期间给予我鼓励，支持的人表达我最诚挚的感谢。

首先，我要感谢我的导师高建彬副教授，是他在我迷茫时为我指点方向，在我困惑时为我解决疑惑。高老师对工作认真的态度，对学术严谨的作风，以及您平易近人，宽以待人的人格魅力，对我影响深渊，并值得学习。感谢您从论文的选题，撰写到定稿给予我悉心的指导和帮助。在此，我想由衷的对我的导师说一声，谢谢您。

其次，我要感谢我的家人在我读研期间给予我的支持和关爱，是你们让我有源源不断的动力，并安心的完成我的学业。感谢我的舍友，这三年里是你们像姐妹一样照顾我包容我，寝室对于我来说就像是一个小家庭般的温暖。还要感谢我教研室的同学们，在我科研上遇到困难时给我及时的帮助，在我心情失落时给我安慰。此外，感谢我研究生期间给予过我帮助的所有人，是你们的出现让我这三年的研究生生活变得丰富起来。

最后， 感谢审阅论文的各位老师，你们辛苦了！

参考文献

[1] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, et al. FRVT 2006 and ICE 2006 large-scale results[R]. Gaithersburg：National Institute of Standards and Technology (NISTIR), 2006.

[2] P. A.Viola, M.J.Jones. Rapid object detection using a boosted cascade of simple features[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Kauai, 2001, 511-518.

[3] B.Wu, H.Ai, C.Huang, et al. Fast rotation invariant multi-view face detection based on real Adaboost[C]. IEEE International Conference on Automatic Face and Gesture Recognition, Seoul, 2004, 79-84.

[4] C.Huang, H.Ai, Y.Li, et al. High-performance rotation invariant multiview face detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2007, 29(4):671-686.

[5] M.J.Jones, P.A.Viola. Fast multi-view face detection[J]. Mitsubishi Electric Research Lab TR-20003-96, 2003, 3(14): 2.

[6] S.Z.Li, L.Zhu, Z.Q.Zhang, et al. Statistical Learning of Multi-view Face Detection.[C]// European Conference on Computer Vision(ECCV), Copenhagen, 2002,67-81.

[7] X.Zhu, D.Ramanan. Face detection, pose estimation, and landmark localization in the wild[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, 2012, 2879-2886.

[8] X.Shen, Z.Lin, J.Brandt, et al. Detecting and Aligning Faces by Image Retrieval[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, 2013, 3460-3467.

[9] R.B.Girshick, J.Donahue, T.Darrell, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, 2014, 580-587.

[10] R.B.Girshick. Fast R-CNN[J]. International conference on computer vision, 2015: 1440-1448.

[11] S.Ren, K.He, R.B.Girshick, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[12] K.Simonyan, A.Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. International conference on learning representations, 2015.

[13] W.Liu, D.Anguelov, D.Erhan, et al. Ssd: Single shot multibox detector[C]// European Conference on Computer Vision(ECCV), Cham, 2016, 21-37.

[14] J.Redmon, S.K.Divvala, R.B.Girshick, et al. You Only Look Once: Unified, Real-Time Object Detection[J]. computer vision and pattern recognition, 2016: 779-788.

[15] I.Kalinovskii, V.Spitsyn. Compact Convolutional Neural Network Cascade for Face Detection[J]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Piscataway, 2015, 375-387.

[16] H.Qin, J.Yan, X.Li, et al. Joint training of cascaded cnn for face detection[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 2016, 3456-3465.

[17] K.Zhang, Z.Zhang, Z.Li, et al. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks[J]. IEEE Signal Processing Letters, 2016, 23(10): 1499-1503.

[18] M.Turk, A.Pentland. Face recognition using eigenfaces[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Maui, 1991, 586-591.

[19] P.N.Belhumeur, J.P.Hespanha, D.Kriegman, et al. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(7): 711-720.

[20] J.Wright, G.Hua. Implicit elastic matching with random projections for pose-variant face recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, 2009, 1502-1509.

[21] S.R.Arashloo, J.Kittler. Energy Normalization for Pose-Invariant Face Recognition Based on MRF Model Image Matching[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(6): 1274-1280.

[22] A.B.Ashraf, S.Lucey, T.Chen, et al. Learning patch correspondences for improved viewpoint invariant face recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, 2008,1-8 .

[23] X.Chai, S.Shan, X.Chen, et al. Locally Linear Regression for Pose-Invariant Face Recognition[J]. IEEE Transactions on Image Processing, 2007, 16(7): 1716-1725.

[24] Z.Zhu, P.Luo, X.Wang, et al. Deep Learning Identity-Preserving Face Space[C]. IEEE International Conference on Computer Vision (ICCV), Sydney, 2013, 113-20.

[25] V.Blanz, T.Vetter. Face recognition based on fitting a 3D morphable model[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(9): 1063-1074.

[26] X.Zhu, Z.Lei, J.Yan, et al. High-fidelity Pose and Expression Normalization for face recognition in the wild[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, 2015, 787-796.

[27] T.Hassner, S.Harel, E.Paz, et al. Efficient face frontalization in unconstrained images[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Patna, 2015, 4295-4304.

[28] M. Guillaumin, J. Verbeek, C. Schmid. Is that you? Metric learning approaches for face identifications[C]. IEEE International Conference on Computer Vision (ICCV), Kyoto, 2009, 498–505.

[29] W. Zhang, S. Shan, W. Gao, et al. Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition[C]. IEEE International Conference on Computer Vision (ICCV), Beijing, 2005, 786–791.

[30] L.Zhang, M.Yang, X.Feng, et al. Sparse representation or collaborative representation: Which helps face recognition? [C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, Barcelona, 471-478.

[31] D.Chen, X.Cao, F.Wen, et al. Blessing of Dimensionality: High-Dimensional Feature and Its Efficient Compression for Face Verification[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, 2013, 3025-3032.

[32] A.Krizhevsky, I.Sutskever, G.E.Hinton, et al. ImageNet classification with deep convolutional neural networks[C]. Conference on Neural Information Processing Systems(NIPS), Lake Tahoe, 2012, 1097-1105.

[33] C.Szegedy, W.Liu, Y.Jia, et al. Going deeper with convolutions[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, 2015, 1-9.

[34] Y.Sun, X.Wang, X.Tang, et al. Deep Learning Face Representation from Predicting 10,000 Classes[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, 2014, 1891-1898.

[35] Y.Sun, Y.Chen, X.Wang, et al. Deep learning face representation by joint identification-verification[C]. Conference on Neural Information Processing Systems(NIPS), Montreal, 2014,.

[36] Y.Sun, X.Wang, X.Tang, et al. Deeply learned face representations are sparse, selective, and robust[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston ,2015, 2892-2900.

[37] Y.Sun, X.Wang, X.Tang, et al. Sparsifying Neural Network Connections for Face Recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 2016, 4856-4864.

[38] Y.Taigman, M.Yang, M.Ranzato, et al. DeepFace: Closing the Gap to Human-Level Performance in Face Verification[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, 2014,1701-1708.

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

[40] D.E.Rumelhart, G.E.Hinton, R.J.Williams. Learning representations by back-propagating errors[J]. nature, 1986, 323(6088): 533.

[41] Y.Lecun, Y.Bengio, G.Hinton. Deep learning[J]. Nature, 2015, 521(7553):436.

[42] V.Nair, G.E.Hinton. Rectified linear units improve restricted boltzmann machines[C]// International Conference on Machine Learning (ICML), Haifa, 2010, 807-814.

[43] A.Krizhevsky, G.Hinton. Learning multiple layers of features from tiny images[J]. 2009.

[44] A.L.Maas, A.Y.Hannun, A.Y.Ng. Rectifier nonlinearities improve neural network acoustic models[C]// International Conference on Machine Learning, Atlanta ,2013, 30(1): 3.

[45] S.Ioffe, C.Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift[J]. international conference on machine learning, 2015: 448-456.

[46] P.A.Viola, M.J.Jones. Robust real-time face detection[J]. international conference on computer vision, 2001, 57(2): 137-154.

[47] Y.Freund, R.E.Schapire. A decision-theoretic generalization of on-line learning and an application to boosting[J]. Journal of computer and system sciences, 1997, 55(1): 119-139.

[48] F.Chollet. Xception: Deep Learning with Depthwise Separable Convolutions[J]. computer vision and pattern recognition, 2016: 1251-1258.

[49] Y.Jia, E.Shelhamer, J.Donahue, et al. Caffe: Convolutional architecture for fast feature embedding[C]//Proceedings of the 22nd ACM international conference on Multimedia, Orlando, 2014, 675-678.

[50] Z.Liu, P.Luo, X.Wang, et al. Deep learning face attributes in the wild[C]// IEEE International Conference on Computer Vision (ICCV), Los Alamitos, 2015, 3730-3738.

[51] D.Chen, S.Ren, Y.Wei, et al. Joint cascade face detection and alignment[C]// European Conference on Computer Vision(ECCV), Cham, 2014, 109-122.

[52] T.Ojala, M.Pietikäinen, D.Harwood. A comparative study of texture measures with classification based on featured distributions[J]. Pattern recognition, 1996, 29(1): 51-59.

[53] T.Ojala, M.Pietikäinen, T.Maenpaa, et al. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7): 971-987.

[54] M.Pietikäinen, T.Ojala, Z.Xu, et al. Rotation-invariant texture classification using feature distributions[J]. Pattern Recognition, 2000, 33(1): 43-52.

[55] D.G.Lowe. Distinctive image features from scale-invariant keypoints[J]. International journal of computer vision, 2004, 60(2): 91-110.

[56] R. Hadsell, S. Chopra, Y. Le Cun. Dimensionality reduction by learning an invariant mapping[C] IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New York, 2006, 1735-1742.

[57] G.B.Huang. Labeled faces in the wild: A database for studying face recognition in unconstrained environments[R]. Amherst, University of Massachusetts, 2007.

[58] J.Yang, K.Yu, Y.Gong, et al. Linear spatial pyramid matching using sparse coding for image classification[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, 2009, 1794-1801.

[59] N. S. Vu, A. Caplier. Enhanced patterns of oriented edge magnitudes for face recognition and image matching[J]. IEEE Transactions on Image Processing (TIP), 2012, 21(3):1352–1365.

[60] D.Chen, X.Cao, L.Wang, et al. Bayesian face revisited: A joint formulation[C]// European Conference on Computer Vision(ECCV), Florence, 2012: 566-579.

[61] C.Stauffer, W.E.L.Grimson. Adaptive background mixture models for real-time tracking[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, 1999, 246-252.

[62] A.P.Dempster. Maximum likelihood estimation from incomplete data via the EM algorithm [J]. Journal of the Royal Statistical Society, 1977, 39(1):1-38.

[63] H.Li, G.Hua. Hierarchical-PEP model for real-world face recognition[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Piscataway, 2015: 4055-4064.

[64] K.Simonyan, O.M.Parkhi, A.Vedaldi, et al. Fisher Vector Faces in the Wild[C]// British Machine Vision Conference (BMVC), Bristol, 2013, 1-11.

[65] A.T.Tran, T.Hassner, I.Masi, et al. Regressing robust and discriminative 3D morphable models with a very deep neural network[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos , 2017, 1493-1502.

[66] V Kazemi, J Sullivan . One millisecond face alignment with an ensemble of regression trees[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, 2014, 1867-1874.

攻读硕士学位期间取得的成果

1. 参与的主要科研项目

[1] 四川省科学技术厅，复杂环境下低分辨率视频人脸识别算法研究，项目编号：2015JY0043

2. 发表的专利

[1] 高建彬,刘婧月.基于深度学习和关键点特征提取的人脸识别方法[P].中国,发明专利,201610682083.6,2016 年 8 月 18 日