

学校代码 10530
分 类 号 TP391

学 号 201510171802
密 级 公 开

湘潭大学

硕士 学位 论 文

基于卷积神经网络的人脸识别研究

学 位 申 请 人 王 飞

指 导 教 师 张莹 副教授

学 院 名 称 信息工程学院

学 科 专 业 控制科学与工程

研 究 方 向 模式识别与智能信息处理

二〇一八年六月

Research on Face Recognition based on Convolution Neural Network

Candidate Wang Fei

Supervisor Associate Prof. Zhang Ying

College College of Information Engineering

Program Control Science and Engineering

Specialization Pattern recognition and intelligent information processing

Degree Master of Engineering

University Xiangtan University

Date Jun , 2018

湘潭大学

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出。本人完全意识到本声明的法律后果由本人承担。

作者签名：

王飞

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。本人授权湘潭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。

涉密论文按学校规定处理。

作者签名：

王飞

日期： 年 月 日

导师签名：

张革

日期： 年 月 日

摘要

卷积神经网络的局部连接、权值共享、池化操作及强大的自动提取特征的特性能够有效降低网络结构的复杂度，并对图像的缩放、平移、扭曲不变性具有良好鲁棒性的特征，因而将卷积神经网络应用于人脸识别能够有效改善识别的性能，本文围绕卷积神经网络架构进行了研究，针对不同的问题提出了相应的解决算法。

针对传统卷积神经网络经过多次池化降维，部分信息在传播过程中损失的问题。本文提出捷径连接的模型：综合考虑浅层次特征和深度特征，并将二者进行融合，减少了特征损失；为了获得更加有区分性的特征，算法在原有 softmax 损失函数的基础上添加一个惩罚项，使已学习到的同类特征到该类特征之间的距离最小化。降低了细节特征的损失程度，增加了深度特征的区分性。

针对传统卷积神经网络不能充分利用图像的多尺度信息的问题，提出了一种多尺度残差网络模型，通过多尺度跨通道卷积的融合来提高图像的特征表达能力；为了降低深度网络架构的学习参数并使其易于优化求解，提出直接将上一层特征跨越多尺度模块层直接映射到下一层，并将大的卷积核分解成小的非对称卷积来降低网络参数量。

为了便于验证模型的性能，将本文设计的非对称小卷积模型进行了图形界面实现，实现了实时人脸识别系统。实验结果表明，本文算法对图像的尺度变化、遮挡、复杂性背景等因素具有良好的鲁棒性。

关键词：卷积神经网络；人脸识别；捷径连接；多尺度残差网络；跨通道卷积

Abstract

The characteristics of local connection, weight sharing, pooling operation and the powerful automatic extraction feature effectively reduce the complexity of the network structure, and obtain good robustness for image scaling ,translation and distortion invariance. So the convolutional neural networks can effectively improve the recognition performance if it is applied to face recognition. This paper studies the architecture of convolutional neural network and corresponding solutions are proposed for different problems.

In order to reduce the loss of information in the process of transmission derived from multiple pooling operation in the traditional convolutional neural network, The shortcut connection model is proposed that the shallow and deep feature are integrated so that the feature loss can be reduced. At the same time, in order to obtain more distinct feature, our algorithm adds a penalty to the original softmax loss function so that the distance can be minimized between the learned class feature . As a result, the loss of the detail feature are reduced and the discrimination of depth feature are increased.

To solve the problem thatthe multi-scale information of the image cannot be made full of use in the traditional convolutional neural network, the multi-scale residual network model is proposed. the feature characterization is improved through convolution fusion of multiscale and cross channel. At the same time, in order to reduce the learned parameters of the deep network architecture and make it easy to be optimized. The feature of the previous layer is concatenate with the feature of the next layer across the multi-scale module layer directly and then the large convolution kernel is decomposed into a small asymmetric convolution kernel which reduces network parameter calculation.

In order to validate the performance of the model, the graphical interface is implemented with the asymmetric little convolution model. The experimental results show that the algorithm has good robustness to the scale change, occlusion, complexity background and so on.

Key words: Convolutional neural network; Face recognition; Shortcut connection; Multiscale residual network; Cross channel convolution

目 录

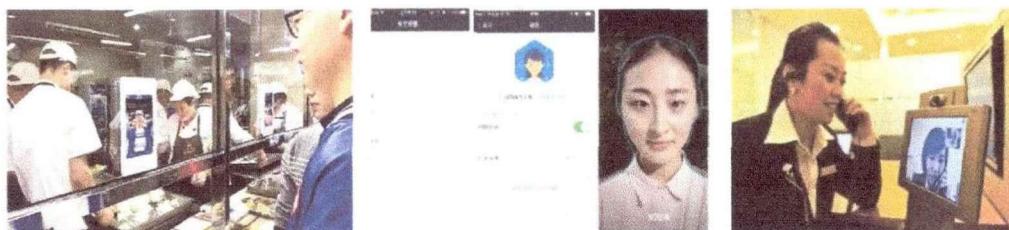
摘要	I
Abstract	II
第 1 章 绪论	1
1.1 研究背景与意义.....	1
1.2 国内外研究现状.....	2
1.2.1 人脸识别及 CNN 的研究现状.....	2
1.2.2 深度学习框架研究现状.....	5
1.3 本文的主要工作及组织结构.....	7
第 2 章 卷积神经网络及其设计基础	9
2.1 卷积神经网络概述.....	9
2.1.1 卷积层.....	10
2.1.2 池化层.....	12
2.1.3 损失函数.....	12
2.2 卷积神经网络设计基础.....	15
2.2.1 数据的存储与交换.....	16
2.2.2 层的计算和连接.....	16
2.2.3 网络的定义和操作.....	17
2.3 本章小结.....	17
第 3 章 基于捷径的卷积神经网络的人脸识别	18
3.1 基于捷径卷积神经网络算法.....	18
3.1.1 捷径连接网络.....	20
3.1.2 附加惩罚函数.....	20
3.2 图像预处理.....	22
3.3 实验结果与分析.....	23
3.3.1 CASIA-web 数据集实验结果.....	23
3.3.2 Facescrub 数据集实验结果.....	26
3.3.3 实验结果分析.....	28
3.4 本章小结.....	29
第 4 章 基于多尺度残差网络模型的人脸识别	30
4.1 基于多尺度残差网络模型.....	30
4.1.1 多尺度跨通道卷积.....	30
4.1.2 卷积核分解.....	32
4.1.3 多尺度小卷积核.....	35

4.1.4 多尺度残差网络.....	38
4.2 实验与数据分析.....	40
4.2.1 数据预处理.....	40
4.2.2 模型训练与性能评估.....	41
4.2.3 实验结果分析.....	42
4.3 人脸识别系统测试.....	42
4.3.1 机器识别一般过程.....	42
4.3.2 识别系统测试.....	43
4.4 本章小结.....	44
第 5 章 总结与展望.....	45
5.1 研究工作总结.....	45
5.2 展望.....	45
致 谢.....	47
硕士期间学术论文及研究成果.....	48
参考文献.....	49

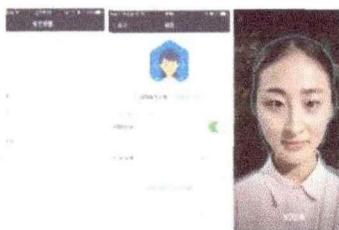
第1章 绪论

1.1 研究背景与意义

生物特征识别技术将计算机技术、光学技术、生物传感器技术、声学等技术进行了有机结合，充分利用人体本身固有的生理特征，并将其作为验证个人身份的一项技术。与其他生理特征相比，面部特征易于获取，不需要与识别设备之间有任何接触，逐渐被应用于日常生活中。比如图 1-1 中的刷脸支付、刷脸门禁、支付宝的刷脸登陆、ATM 刷脸取款、刷脸远程开户、火车站的刷脸进站等。



(a)刷脸支付



(b)支付宝刷脸登陆



(c)刷脸远程开户



(d)刷脸门禁



(e)ATM 刷脸取款



(f) 刷脸进站

图 1-1 各种场景下的人脸识别

但是面部图像的获取对光照条件、获取面部图像的角度、面部图像与获取图像设备之间的距离的远近、成像对象的姿势、面部图像是否受到遮挡等因素比较敏感。而且不同年龄段的面部图像也有很大的差别，以及不同时间段获取的不同表情下的面部图像也会导致面部图像特征的差异。这些因素都给机器正确识别人脸带来了极大的挑战。而卷积神经网络^[1](Convolutional Neural Networks, CNN)对上述缺点有很好的鲁棒性，并且可以自动提取图像特征，减少了科研工作者手工提取特征的繁琐步骤。因而，通过研究使用 CNN 能够有效改善人脸识别的性能。

1.2 国内外研究现状

1.2.1 人脸识别及 CNN 的研究现状

早期人们主要是采用手工设计的脸部几何特征进行面部识别。随着像 FERET^[2]这样的大型人脸库的建立，进一步推动了人脸识别的发展。这个时期科研人员提出的部分人脸识别算法已经达到实用的程度，尤其是由麻省理工学院提出的特征脸法^[3]以及将 PCA 与 LDA 相结合而提出的 Fisherface 人脸识别算法^[4,5]，更是加大了人脸识别算法在现实生活中的应用范围。在 20 世纪 90 年代末，Rikert 等人提出了 AdaBoost 算法^[6]，推动了人脸检测方面研究性进展，以及利用 BP 网络架构^[7,8]进行人脸识别，均为后续的人脸检测与人脸识别的相关研究奠定了基础。20 世纪后期，相关科研人员将支持向量机^[9]、Boosting^[10]等算法成功应用于人脸识别，进一步提高了人脸识别的性能。与此同时，一部分科研人员针对多姿态对识别性能的影响的问题开展了研究并提出了 3D 变形模型(3D Morphable Model)^[11,12]，提高了人脸识别的鲁棒性。在 21 世纪初，Liu^[13]等人提出的 Gabor 算法以及 Ahonen^[14]等人提出的 LBP 算法被广泛的应用于人脸识别，在 2013 年微软亚洲研究院的研究者首先尝试了用 10 万规模的大规模训练数据对模型进行训练，提出的基于高维 LBP 特征和联合贝叶斯方法在 LFW^[15]上获得了 95.17% 的精度。这一结果表明：增加训练数据集的规模有助于提升非受限环境下的人脸识别性能，然而传统的方法不利于处理大规模训练数据集。

大数据和计算机计算能力的提高共同推动着深度学习的前进：大规模的数据集可以让科研工作者训练出更深的网络模型，而 GPU 的迅速发展，大大减少了训练深度网络模型所需的时间。这些因素均为科研人员设计并深度网络模型进而提高模型的性能提供了可能。因而近年来相关科研人员对神经网络，尤其是卷积神经网络在图像分类方面的研究不断深入。主要从以下方面进行了研究。

为了能够在小样本中有效的处理图像分类问题，2005 年 LeCun 提出能够利用 CNN 分别提取并计算两张同时输入的图像的特征向量之间相似度的孪生网络（SiameseNet）^[16]，该方法能够判断输入的图像是否“一样”，对解决那些类别数目比较多但是每个类别样本数量较少的情况或者整个训练样本无法使用传统方法训练的分类问题做出了很大贡献。2010 年间，Zeiler 等人提出使用反卷积网络（Deconvolutional Networks）^[17,18]来学习图像中的中级高级特征，从而充分提取图像中包含的信息。之后伴随着大数据和 GPU（Graphics Processing Unit）的迅速发展，2012 年，Alex Krizhevsky 等人通过设计的卷积神经网络（AlexNet）在 2012 年的大规模视觉识别挑战赛（ImageNet Large Scale Visual Recognition Challenge, ILSVRC）中以领先第二名使用传统机器学习方法 10.9 个百分点的优

异成绩取得冠军，充分体现出了 CNN 超强的特征提取能力，这使得 CNN 越来越受到科研人员的重视。在随后的 5 年期间，研究人员更是提出了各种形式的 CNN 架构，进一步增强了 CNN 模型的特征提取能力，比较有代表性的网络架构有：从 2013 年 Lin M 等人利用多层感知器（Multilayer Perceptron，MLP）构成的微型网络对局部感受野进行复杂运算来近似逼近非线性函数的 NIN 网络（Network In Network，NIN）^[19]、到 2014 年 ILSVRC 竞赛的冠军获得者 GoogLeNet^[20]以及其后续衍生得到的各种 Inception 变体模型^[21,22,23]，这一家族的网络架构的设计特点是通过增加网络模型的宽度来增加模型的鲁棒性，同一年，ILSVRC 竞赛亚军的获得者 VGGNet^[24]则通过增加模型的深度来提高网络的性能，这两种提升网络性能的原则均为后续研究人员设计网络提供了极大的指导。随着网络模型越来越大，层数越来越深，梯度消失的问题愈发严重，使得训练网络变得困难，为解决这一问题，2015 年的 ResNet^[25]应运而生，与此同时，科研工作者发现经典的 CNN 中的输入图像的尺寸必须固定，而固定尺寸的输入图像限制了网络最后的全连接网络，进而使得整个网络的输入尺寸无法改变，考虑到 CNN 的特性是可以接受任意尺寸的输入图像，因而设计出了全卷积网络（Fully Convolutional Networks，FCN）^[26]，该网络模型能够上采样最后一卷积层输出的特征图并使其恢复到与输入图像相同的尺寸，进而可以保留原图的空间信息并预测出每一个像素值。最近几年研究人员试图从空间维度的层面来提升网络的性能，如 Inception 结构嵌入了多尺度信息，聚合了多种不同感受野上的特征来获得性能的增益；在 Inside-Outside^[27]网络中考虑了空间中的上下文信息，Xception^[28]和 ResNeXt^[29]两种网络则从通道入手，认为既然不同的卷积参数会产生不同的卷积效果，而当卷积参数比较有限时，产生的输出结构难免会有一定的相关性，如果将这种相关的特征放在一起考虑，有时并不一定会产生更好的效果，反而可能会造成一定程度的过拟合，因而这两种网络在设计时，并不把处理通道得到的所有信息都考虑在内，而是进行分组卷积，二者的实验也充分验证了通道也具有局部相关性。以及 2017 年 Imagenet 竞赛冠军模型 SENet^[30]能够自动学习出每个特征通道的重要程度，进而强化有用特征，抑制用处不大的特征。DenseNet^[31]和 ResNet 则是通过将前面层得到的特征信息直接跨层输入到下一层，从而实现特征信息的整合，避免了层间传递时导致的信息丢失和梯度消失问题。

人脸识别作为图像分类中的一个重要应用，2014 年时，香港中文大学的团队^[32]将卷积神经网络应用到人脸识别上，获得了 97.45% 的识别精度，在 LFW 上第一次超过了人类水平的识别精度，这是人脸识别发展历史上的又一座里程碑，自此以后，研究者不断改进网络架构，同时扩大训练样本的规模，将人脸识别的精度进一步提高，如通过大规模数据集、更深的卷积神经网络模型并使用 triplet

loss 进行度量学习的 VGG face^[33], 通过使用 MaxOut 作为激活函数过滤了噪声、保留了有用信号并借鉴 NIN 网络来减少网络架构的参数量而设计出的 Light CNNs^[34]、谷歌提出的 FaceNet^[35]则采用 Inception 架构进行提取特征的同时，直接使用 Triplets 损失函数直接将提取的人脸图像特征映射到欧几里得空间，保留了原有特征的结构信息，进而提高了面部识别的精度。

与此同时，为了便于在现实场景中检测出人脸，部分科研工作者在人脸检测对齐方面做出了很大贡献：为了得到更精确的面部关键点的位置，2013 年 Sun 等人将不同卷积神经网络级联在一起且每个级联层通过使用多个不同 CNN 来计算不同的关键点，并将其进行融合，进而获得更精确的面部关键点的位置。2016 年 Zhang 等人在构建图像金字塔的基础上，利用全卷积网络进行检测，同时使用边界框和非极大值抑制等方法进行修正，提高了人脸检测的精度。

尽管科研工作者对面部识别做了很多工作并取得了很多成果，但是由于获取人脸图像的角度和距离的不同、面部表情的变化以及所处背景的复杂性，使得面部识别仍面临着极大的挑战，其主要影响因素具体如下：

(a) 光照的影响

被识别对象所处环境中光照的强弱直接影响着算法最终的识别性能。当光照太强时，被识别对象面部的关键点特征区分性不大；当光照太弱时，采集到的图像信息的特征不易于算法的提取，有时算法甚至无法检测到人脸。这些因素都严重影响了后续算法的识别性能。针对这些问题，主要通过对被采集的图像进行适当的光照补偿或者直接采用红外图像作为算法的输入图像。

(b) 年龄因素的影响

同一个人在不同年龄段的面部结构也会有细微的差别，从而导致提取的特征不一样，如面部五官的变化、皮肤纹理的变化等。随着时光的流逝，这些细微差别会产生巨大的特征差异，使机器正确分辨出每个人变得困难。

(c) 姿态的影响

由于目前建立的大多数人脸数据库多数是只包含正面姿态的面部图像，这样训练出来的识别算法可能对训练集具有很好的识别效果，但是现实生活中摄像头采集到的人脸图像可能处于各种姿态，这就大大降低了算法的鲁棒性。虽然科研工作者提出使用能够改变模型姿态的 3D 人脸来降低多姿态问题的影响^[36]，但这对获取人脸的设备提出了苛刻性要求。

(d) 遮挡因素的影响

面部被遮挡如口罩遮挡嘴部、墨镜遮挡双眼、毛发遮挡额部等都会造成被遮挡部位的特征不能被提取，进而增加了机器识别人脸的难度。

(e) 人脸表情的影响

同一个人在不同时段的喜怒哀乐使得五官发生一定程度扭曲变形，进而使面部呈现出不同的图像，这就使得提取的人脸图像的特征向量发生变化，从而增加了机器识别人脸的难度。

(f) 目标图像的分辨率影响

图像采集设备的分辨率越高，采集的人脸图像丢失的信息越少，越逼近真实人脸，但是对获取人脸图像的硬件设备提出了苛刻要求，而且某些特殊场合只能获取低分辨率的图像。大大降低了机器识别的准确性。

(g) 人脸所处背景的影响

人类可以轻松的从具有复杂背景的图像中将背景和识别对象分辨出来。但是机器很难提取出具有过于复杂背景的面部图像，因为机器并不会像人类那样去思考这幅图像中的不同目标对象之间的差别与联系，这就给机器后续的识别任务带来了困难。

(h) 尺度变化的影响

人脸图像与图像采集设备之间的距离越远，采集出的图像包含的信息越少，越不利于机器准确将其识别出来；另一方面，距离目标过近，目标过大，也可能无法准确的将其识别出来，因此，待识别人脸图像的尺度也给人脸识别带来了困难。

因此，由于这些因素的影响，如何让机器准确的识别出人脸，仍然是一个极具挑战的课题。鉴于上述影响因素，本文主要针对尺度变化、遮挡、复杂背景下的人脸图像等因素对人脸识别的影响进行了探究。

1.2.2 深度学习框架研究现状

近年来，深度学习发展迅速，除了大数据和计算机计算能力的提高对其发展产生了巨大的推动作用之外，各种深度学习框架的出现也对其发展产生了积极促进作用。通过使用这些深度学习框架，科研人员可以将更多的精力用于算法的研发上，而不必每次都从零开始写代码实现相应的算法，从而进一步推动深度学习的不断进步。近年来，各种开源深度学习框架也层出不穷，如图 1-2 所示，其中包括 Caffe^[37]、Caffe2^[38]、Tensorflow^[39]、PyTorch^[40]、Keras^[41]、CNTK^[42]等等。下面本文首先对几个目前主流的开源深度学习框架研究现状进行简短介绍。





图 1-2 主流深度学习框架

Tensorflow

Google 在 2015 年推出新一代的人工智能学习系统—Tensorflow。它作为一个异构分布式深度学习框架，具有高度的灵活性、较强的可移植性、能够自动求微分、不仅可以运行在服务器端，而且也可以直接运行在手机客户端，构建深度网络架构只需调用并组合相应的 API 接口即可。

CNTK

CNTK（Computational Network Toolkit）是微软研究院用于搭建深度神经网络的跨平台深度学习系统，但其主要应用于语音识别领域。CNTK 支持各种前馈网络、自动求解梯度、支持多机、多 GPU 的分布式训练、Linux、MacOS、Windows，但是目前不支持 ARM 架构，限制了其在移动设备上的发挥。

Keras

Keras 是一个依赖 Tensorflow 或 CNTK 作为后台的高层神经网络 API，其相对于标准的 GPU 建模环境，比如 CNTK、Tensorflow、Caffe、Theano 等，有一个非常大的优势是使用简单，代码高度模块化。其缺点是只能使用目前已有的结构来搭建自己的神经网络且只提供一个建模环境。

Pytorch

Pytorch 是 2017 年 Facebook 新开源的深度学习框架，能够动态的构建深度神经网络，能够有效处理时序问题，如对 RNN 变化时间长度的输出提供了很好的支持，提供了具有强大 GPU 加速的张量计算，同时将深度神经网络建立在基于磁带的自动调整系统上，能够重复使用相同的结构。因而 PyTorch 很快成为目前主流的深度学习框架之一。

Caffe/Caffe2

Caffe 作为目前主流的深度学习框架，在 Ubuntu 系统下，用户不仅可以使用 Shell 脚本来实现相应的算法，而且支持 Matlab、Python，用户只要预先编译好相应的接口，就可以轻松的使用相应的 API 来构建自己的网络架构，并且 Caffe 框架能够配合 CUDA 进行高效的并行计算，从而大大缩短了模型的训练时间。其自带了许多很好的例子，如 Lenet、Alexnet、GoogleNet 等模型以及训练好的模型文件，方便初学者在此基础上按照自己的需求微调成适合自己的模型，以及用于特征提取与分类、将图像数据转换成 Lmdb、Leveldb 等格式的数据文件工具，Caffe 凭借其简单易用的源码、易于设计模型的模块化设计在国内外有比较活跃的社区。

目前，贾扬清同其团队对原有 Caffe 源码进行了重构，并在 2017 年 4 月 18

开源了第二代 Caffe 并将其命名为 Caffe2，同其他深度学习框架相比，更注重模块化设计的 Caffe2 不仅能够在服务器端表现出优越性能，而且可以快速的部署到移动端，如同 Tensorflow 一样，Caffe2 使用 C++、Eigen 库，支持 ARM 架构，并且 Caffe 和 Caffe2 之间可以方便的转换，使用 Caffe 框架训练的模型可以轻松的转换成 Caffe2 支持格式的模型，且增加了对递归神经网络、长短期记忆网络的支持。

其他深度学习框架

除了上述主流的深度学习框架之外，百度开源的 PaddlePaddle、英伟达公司开源的 DIGITS 以及使用 Java 开发的 Deeplearning4J、Chainer 还有亚马逊 DSSTNE 等等这些框架也有一定的用户使用。

1.3 本文的主要工作及组织结构

本论文主要基于 Caffe 平台，对卷积神经网络架构设计以及惩罚函数设计进行了研究并将其应用于人脸识别。全文共分五章，具体内容安排：

- 第 1 章 绪论。概述了为什么人脸识别会逐渐取代传统的身份鉴别手段、为什么要用卷积神经网络进行人脸识别、国内外人脸识别及卷积神经网络的研究进展以及目前主要用于构建深度卷积神经网络的常用框架。
- 第 2 章 卷积神经网络及其设计基础。首先对卷积神经网络理论：卷积神经网络的特性、卷积层、池化层、损失函数进行了相关研究，然后以 Caffe 平台为例，阐述了在 Caffe 中如何有效管理输入的数据，输入的数据如何进行加工计算以便获得需要的特征，如何将数据和加工数据组合成能够执行相应任务的网络。
- 第 3 章 基于捷径的卷积神经网络的人脸识别。本章从特征损失及损失函数角度出发，对卷积神经网络进行了研究，提出了一种带有附加惩罚函数的捷径连接的人脸识别模型，综合考虑了浅层次特征和深层次特征，降低了多次池化操作带来的信息损失程度，通过惩罚函数降低已学习到的深度特征之间的类内距离，提高了分类的准确率。最后，在 CASIA-Web 和 Facescrub 两个人脸库上验证了算法的可行性，并给出了实验结果对比。
- 第 4 章 基于多尺度残差网络模型的人脸识别。本章深入研究了卷积核的工作机制，针对目前大多数算法对图像尺度变化、遮挡问题比较敏感的问题及深度神经网络参数多计算量大的问题进行了研究，提出了多尺度残差网络模型，同一卷积模块层并联多种尺寸的卷积核来提高鲁棒性，通过卷积核分解来降低学习参数，首先在 Mnist 数据集上逐个验证提出的几种设计原则的有效性，然后同时综合使用提出的卷积核设计原则，设计了

更深的 MResNet 网络，并在 MS-Celeb-1Mv1 数据集上进行了实验，最后使用设计的 MResNet 实现了实时人脸识别系统，进一步验证了本算法对尺度变化、遮挡问题的有效性。

第 5 章 总结与展望。对已有工作进行了归纳总结，并对目前存在的问题，最后对人脸识别、深度学习的前景进行了展望。

第 2 章 卷积神经网络及其设计基础

CNN 的局部连接、权值共享、池化操作及强大的自动提取特征的特性使其更加接近于现实世界中的神经网络，能够有效降低网络结构的复杂度，并获得对图像的缩放、平移、扭曲不变性具有良好鲁棒性的特征，而被广泛应用于目标检测、图像分类、目标跟踪等领域。为了探究卷积神经网络提取特征的机理以及为了能够更好的使用 Caffe 快速构建出自己的卷机神经网络架构，本章首先阐述了卷积神经网络的相关原理，然后对本文构建卷积神经网络的 Caffe 框架的数据结构进行了深入研究。

2.1 卷积神经网络概述

传统的卷积神经网络模型架构一般包括进行滤波运算的卷积层、用于降低维数并提取显著性特征的池化层、度量模型优劣的损失函数层以及起到分类器作用的全连接层等。池化层一般是交替出现在卷积层之后与全连接层相连接。CNN 采用局部连接、权值共享和池化方法，从而获得很好的平移、缩放、扭曲不变性使提取的特征更具有区分性。

(1) 局部连接

局部连接是 CNN 中比较重要的一个特性，由于图像本身的统计特性：图像一个区域的统计特性跟其他区域的统计特性相同，因此可以利用同一个核对图像各个区域分别进行卷积运算。局部连接有助于提取图像的边缘信息等局部特征、减少相邻层之间神经元的连接数目、降低模型的计算复杂度等。如图 2-1 所示

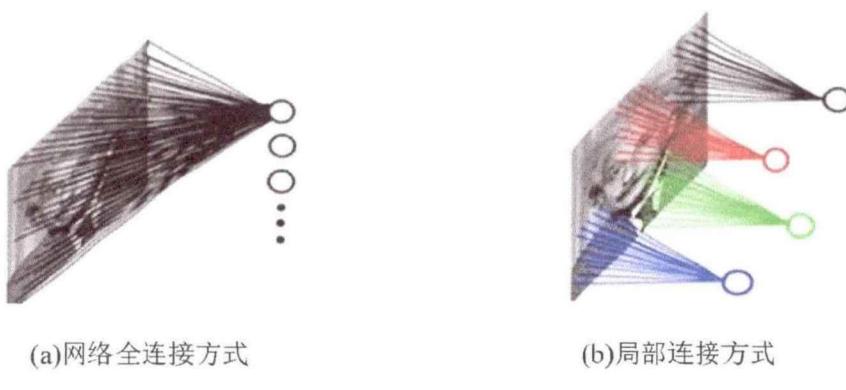


图 2-1 网络连接方式

假定图像尺寸为 400×400 像素，网络中有 100 个隐藏神经元，那么图 2-1(a) 表示的全连接网络需要 $400 \times 400 \times 100 = 1.6 \times 10^7$ 个连接，假设局部连接使用的局部区域尺寸为 10×10 像素，则图 2-1 (b) 的局部连接网络只需要

$10 \times 10 \times 100 = 10^4$ 个连接。因而采用局部连接减少了网络结构的连接数目，有助于提高计算速度。如果图像尺寸很大，采用全连接的方式对硬件的要求较高，而采用局部连接不仅可以有效降低计算复杂度，还可降低对硬件的要求。

(2) 权值共享

为了减少可训练参数数目，卷积神经网络采用了权值共享的机制，同一个特征图使用相同的滤波器对前一层进行计算操作。在进行网络架构设计时，一般同时使用多种尺寸的滤波器以便提取出图像的多级特征，增加模型的鲁棒性。如图 2-2 所示。

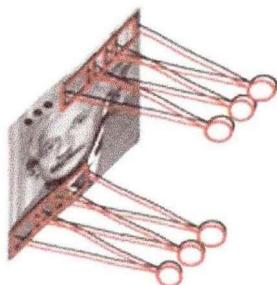


图 2-2 权值共享连接

同样此处假定图像尺寸大小依然为 400×400 像素，网络中有 100 个隐藏单元，滤波器的尺寸依然为 10×10 ，如果按照图 2-1 (a) 所示的全连接方式需要学习 $400 \times 400 \times 100 = 1.6 \times 10^7$ 个参数，而如果采用权值共享的方式，只有 100 个参数需要学习。因此权值共享机制大大减少了网络结构中的可训练参数的数目，减小了网络的学习难度。

(3) 空间相关池化

基于图像空间相关性原理，卷积神经网络加入了池化机制，池化一般紧跟在卷积层之后，虽然卷积层通过局部连接的方式减少了神经元结构的连接数目，但是由于特征图数目的增加，使得特征的维度过大，网络训练难度增加，并容易产生过拟合，通过池化操作如最大池化、平均池化等可以有效降低特征的维度并保留图像的有效信息，加快网络的训练速度，在一定程度上避免了过拟合现象。

2.1.1 卷积层

卷积层是卷积神经网络中主要进行卷积操作较为核心的网络层。其定义如式(2-1)所示

$$y_{ij} = \sum_u^m \sum_v^n f_{uv} \cdot x_{u+i-1, v+j-1} + b \quad (2-1)$$

在式(2-1)中 $1 \leq i \leq M, 1 \leq j \leq N, M \times N$ 为卷积层的输入维度大小， $m \times n$

为卷积核的大小， $1 \leq u \leq m, 1 \leq v \leq n$ ， f_{uv} 表示卷积核的第 u 行第 V 列的元素， b 为偏置项， $x_{u+i-1, v+j-1}$ 为对应的 $u+i-1$ 行 $v+j-1$ 列元素的卷积层输入。

对于图像中的每一个像素点，计算它的邻域像素和滤波器矩阵对应位置元素乘积的累加和并将其作为对应像素的输出值，具体计算如图 2-3 所示。

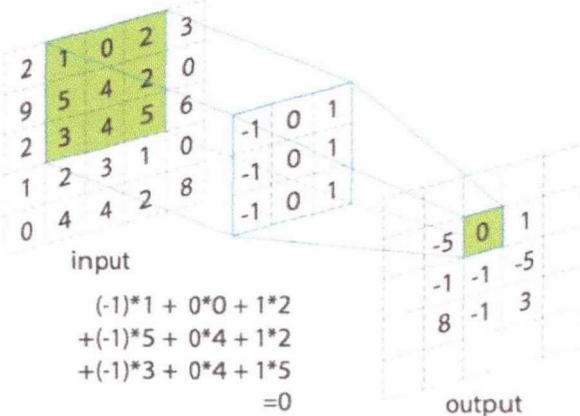


图 2-3 卷积操作示意图

从图 2-3 中可以看到，卷积核就是一个权值矩阵，表示了如何处理单个像素与其邻域像素之间的关系。卷积核中各个元素的相对差值越小，相当于每个像素都与周围像素取了个平均值，就越有模糊去噪的效果，而卷积核元素差值越大，就拉大了每个像素与周围像素的差距，也就越能提取边缘，达到锐化的效果。

在实际应用中，卷积核的尺寸一般比较小，主要因为卷积核的大小与计算量成正比，卷积核尺寸增大使得计算量成倍增加。同时卷积核矩阵一般为正方形，并且边长一般为奇数，比如 1×1 、 3×3 、 5×5 等，这样才能保证有且只有一个中心，以便于使原图像的像素与输出结果相对应。

卷积核矩阵中元素的取值，除了影响输出效果以外，还会影响输出图像的亮度。若卷积核内所有元素的累加和等于 1，那就基本保持了与原图形相同的亮度。变亮的输出图对应于卷积核所有元素累加和大于 1 的情况，变暗的输出图则说明卷积核所有元素的累加和小于 1。当卷积核中元素累加和为 0 时，输出的图像亮度非常低，仅保持了部分图案边缘的亮度，这样的卷积核可以用于边缘提取。卷积效果如图 2-4 所示。



图 2-4 卷积效果图（从左到右依次为原图、卷积核所有元素累加和等于 1、卷积核所有元素累加和大于 1、卷积核所有元素累加和小于 1、卷积核所有元素累加和为 0）

2.1.2 池化层

池化层也称为下采样层，其是将图像按窗口大小划分成不重叠的区域，然后对每一个区域内的元素进行聚合。池化层的滤波器同卷积层滤波器类似，也需要人工设定滤波器的尺寸、是否使用全零填充、滤波器的移动步长，但是池化层的滤波器只需要在一个通道上进行滤波操作，并且需要同时在长、宽和深度三个维度上进行移动操作。一般采用最大值池化、平均值池化等，其计算方式如式(2-2)和式(2-3)，图 2-5 展示了一个最大池化层前向传播计算过程。

$$Pool_{\max}(x) = \max(x_{ij}) \quad (2-2)$$

$$Pool_{\text{aver}}(x) = \frac{1}{m \times n} \sum_{x_{ij} \in R_f} x_{ij} \quad (2-3)$$

其中 $Pool_{\max}(x)$ 表示求最大值， $Pool_{\text{aver}}(x)$ 表示进行求平均值， x_{ij} 对应于滤波器第 i 行第 j 列的元素， $m \times n$ 表示需要求最大值或者均值的区域范围。

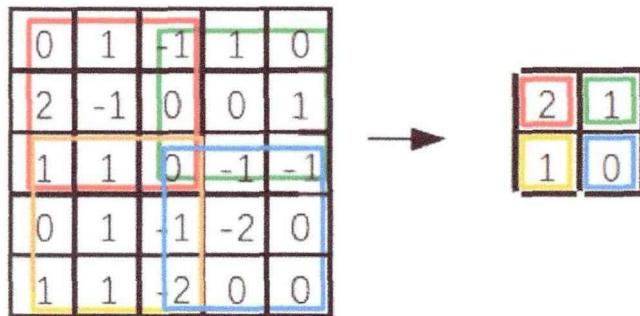


图 2-5 5×5 矩阵步长为 2 的最大池化过程示意图

2.1.3 损失函数

模型的优劣通常通过使用损失函数进行衡量。比如 0-1 损失函数、平方差函数、Softmax 损失函数等。

(1) 0-1 损失函数

0-1 损失函数比较的是预测值 $f(x)$ 与真实值是否相同，其形式如式(2-4)所示

$$J = \begin{cases} 1, & y = f(x) \\ 0, & y \neq f(x) \end{cases} \quad (2-4)$$

0-1 损失函数能够很好的判断出预测值与真实值之间的差别，但是其非凸性不利于优化求解。

(2) 平方差损失函数

平方损失函数是线性回归模型常用的最优化目标函数，其度量了预测值 $f(x)$ 与真实值 y 之间的距离的总体关系，其定义为：

$$J = (y - f(x))^2 \quad (2-5)$$

(3) Softmax 损失函数

本文第三章和第四章设计的网络模型使用 Softmax 损失函数对模型的优劣进行度量，该函数是由 Logistic 回归模型在多分类问题上的推广，能够对多类别标签进行分类。

在一般的针对二分类的 Logistic 回归问题中，训练集由 m 个已标记的样本构成： $\{(x^{(1)}, y^{(1)}), \dots, (x^{(m)}, y^{(m)})\}$ ，其中输入特征 $x^{(i)} \in R^{n+1}$ ，（此处将截距项一并归入到特征向量的维度中），类标记 $y^{(i)} \in \{0, 1\}$ ，预测函数如下：

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}} \quad (2-6)$$

对模型参数进行训练，使其能够使式(2-7)所示的代价函数最小化：

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \log h_{\theta}(x^{(i)}) + (1 - y^{(i)}) \log (1 - h_{\theta}(x^{(i)})) \right] \quad (2-7)$$

对于 k 类别分类问题，为方便叙述，此处假定训练集为 $\{(x^{(1)}, y^{(1)}), \dots, (x^{(m)}, y^{(m)})\}$ ，相应的类别标签为 $y^{(i)} \in \{1, 2, \dots, k\}$ ，针对给定测试输入 X ，通过预测函数估测出 X 属于第 j 类别的概率值 $p(y = j | x)$ ，最后通过输出的 k 个概率值组成输出向量，其计算过程如式 (2-8) 所示。

$$\begin{aligned} h_{\theta}(x^{(i)}) &= \begin{bmatrix} p(y^{(i)} = 1 | x^{(i)}; \theta) \\ p(y^{(i)} = 2 | x^{(i)}; \theta) \\ \vdots \\ p(y^{(i)} = k | x^{(i)}; \theta) \end{bmatrix} \\ &= \frac{1}{\sum_{j=1}^k e^{\theta_j^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \vdots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix} \end{aligned} \quad (2-8)$$

其中 $\theta_1, \theta_2, \dots, \theta_k \in R^{n+1}$ 是模型的参数， $\frac{1}{\sum_{j=1}^k e^{\theta_j^T x^{(i)}}}$ 为归一化概率分布，

其所有概率之和为 1。

为了方便起见, 此处依然使用符号 θ 来表示全部模型的参数, 在实现 softmax 回归时, 将 θ 用一个和 $k \times (n + 1)$ 的矩阵来表示会很方便, 该矩阵的形式如式(2-9)所示:

$$\theta = \begin{bmatrix} \theta_1^T \\ \theta_2^T \\ \vdots \\ \theta_k^T \end{bmatrix} \quad (2-9)$$

因而一般形式 Logistic 可以如下形式:

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{j=1}^k 1\{y^{(i)} = j\} \log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^k e^{\theta_l^T x^{(i)}}} \right] \quad (2-10)$$

当 $y^{(i)} = j$ 时, $1\{y^{(i)} = j\} = 1$, 否则, $1\{y^{(i)} = j\} = 0$ 。

本文将一般的 Logistic 进一步变形, 可以得到 softmax 损失函数形式, 即

$$\begin{aligned} J(\theta) &= -\frac{1}{m} \left[\sum_{i=1}^m (1 - y^{(i)}) \log (1 - h_\theta(x^{(i)})) + y^{(i)} \log h_\theta(x^{(i)}) \right] \\ &= -\frac{1}{m} \left[\sum_{i=1}^m \sum_{j=0}^1 1\{y^{(i)} = j\} \log p(y^{(i)} = j | x^{(i)}; \theta) \right] \end{aligned} \quad (2-11)$$

$$\text{其中 } p(y^{(i)} = j | x^{(i)}; \theta) = \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^k e^{\theta_l^T x^{(i)}}}$$

比较式(2-10)和式(2-11), 可以发现, softmax 和一般的 Logistic 非常相似, 只是在 softmax 中对类别标记 k 个可能值进行了累加。

对于 $J(\theta)$ 的最小化问题, 通过迭代的方法来对其进行求解, 对 $J(\theta)$ 求导后可以得到其梯度式:

$$\nabla_{\theta_j} J(\theta) = -\frac{1}{m} \sum_{i=1}^m [x^{(i)}(1\{y^{(i)} = j\} - p(y^{(i)} = j | x^{(i)}; \theta))] \quad (2-12)$$

如果直接将式(2-11)作为损失函数, 常常会出现参数冗余问题, 将参数向量 θ_j 减去一个向量 ϕ 得到式(2-13):

$$\begin{aligned}
 p(y^{(i)} = j | X^{(i)}; \theta) &= \frac{e^{(\theta_j - \theta)^T x^{(i)}}}{\sum_{l=1}^k e^{(\theta_l - \theta)^T x^{(i)}}} \\
 &= \frac{e^{(\theta_j)^T x^{(i)}} e^{(-\theta)^T x^{(i)}}}{\sum_{l=1}^k e^{(\theta_l)^T x^{(i)}} e^{(-\theta)^T x^{(i)}}} \\
 &= \frac{e^{(\theta_j)^T x^{(i)}}}{\sum_{l=1}^k e^{(\theta_l)^T x^{(i)}}}
 \end{aligned} \tag{2-13}$$

从式 (2-13) 中可以看出, 虽然参数向量被减去一个向量, 但是最终结果一样。这就致使不同组的参数向量可以得到相同的预测函数, 最终得到多组最小值。

为了解决上述问题, 通常加入惩罚项 $\frac{\lambda}{2} \sum_{i=1}^m \sum_{j=0}^n \theta_{ij}^2$, 对原有损失函数进行惩罚, 最终的损失函数就变成式 (2-14) 所示的形式。

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{j=1}^k 1\{y^{(i)} = j\} \log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^k e^{\theta_l^T x^{(i)}}} \right] + \frac{\lambda}{2} \sum_{i=1}^m \sum_{j=0}^n \theta_{ij}^2 \tag{2-14}$$

通过对参数项的惩罚, 避免了参数项的多解问题, 为了使用优化算法, 对(2-14)求导得:

$$\nabla_{\theta_j} J(\theta) = -\frac{1}{m} \sum_{i=1}^m \left[X^{(i)} (1\{y^{(i)} = j\} - p(y^{(i)} = j | X^{(i)}; \theta)) \right] + \lambda \theta_j \tag{2-15}$$

通过最小化 $J(\theta)$, 就可以实现图像分类问题。

2.2 卷积神经网络设计基础

对 CNN 工作机理有了比较深入的了解, 自己如何去实现这一具有强大自动提取特征的网络呢? 本小节以 Caffe 平台来阐述 CNN 模型的相关理论。在 Caffe 框架中, 输入的数据信息以及前向传播的数据信息、反向传播的数据信息以及梯度信息以 Blobs 结构的形式进行存储; 每一个函数被单独定义成一层, 比如用于卷积计算的 Convolution 层、用于增加非线性表达能力的 ReLU 层、用于降维的 Pooling 层等, Caffe 中直接将这些层 (Layers) 作为最基本的计算单元, 这样科研人员只需按需求组合这些层就可以轻松的设计出自己的网络 (Nets) 架构。

2.2.1 数据的存储与交换

要训练出一个网络模型，首先需要有一个优质的数据管理机制，在 Caffe 中，这个数据管理机制通过 Blobs 数据封装包来完成，其能够有效的在 CPU 和 GPU 之间同步处理数据。Blobs 主要负责对需要处理的图像数据信息、前向传播过程涉及的数据信息和反向传播过程中的梯度信息进行管理，如输入的图像数据会被转换成维数为 $N \times K \times H \times W$ 的向量， N 、 K 、 H 、 W 别代表图像的数量、通道数、图像的高度、图像的宽度。Blobs 按照行为主进行存储，所以一个 4 维的 Blobs 中，坐标为 (n, k, h, w) 的物理位置为 $((n \times K + k) \times H + h) \times W + w$ ，这也使得最后面的维度更新最快。

2.2.2 层的计算和连接

卷积神经网络模型，对输入的图像只进行数据管理还远远不能实现预期的功能，我们还需要对输入的图像数据进行加工处理，在 Caffe 中，通过不同的层来实现这些独立加工数据的功能单元，如卷积层（Convolution）、池化层（Pooling）、全连接层（InnerProduct）、ReLU 和 Sigmoid 等非线性运算、批规一化处理（BatchNorm）、Softmax 损失计算等，包含了绝大部分目前最前沿深度学习任务所需要的层类型。

Caffe 中每个单元分工明细，如数据的接收工作由 Bottom 层来完成，经过中间的加工处理后，会将已处理好的数据输送给 Top 层以便进行下一功能单元的数据加工，每一个层级单元都要经过 setup、forward、backward 三级运算。通过 setup 重置层与层之间的连接信息；通过 forward 对从 Bottom 中接收的数据进行相应的计算并将其传递给 Top 中；backward 将计算出相对输入的梯度并将其传递给 Bottom 中。如图 2-6 所示。

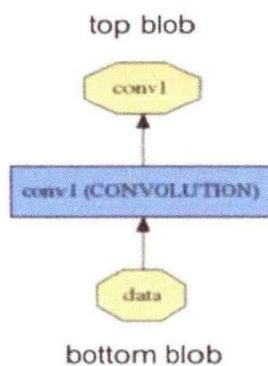


图 2-6 卷积层操作层

2.2.3 网络的定义和操作

设计一个完整的卷积神经网络，除了需要有管理数据的机制、负责数据加工计算的层级单元，还需要将这二者合理的组合搭配才算的上一个完整的网络架构，在 Caffe 中，这二者被组合成有向无环的形式，来执行相应的任务。整个计算过程的中间数据均会被保存下来，确保了前向传播和反向传播计算的准确性。通常一个网络架构的计算过程为：从硬盘读入数据并依次输送给中间的各层级计算单元，如卷积计算、池化操作、数据归一化操作、非线性处理操作等，最后通过损失函数来度量网络计算的输出与期望输出之间的差别。如图 2-7 所示。

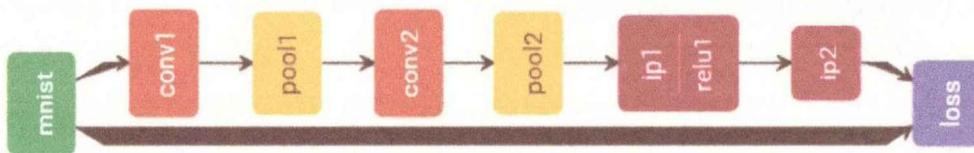


图 2-7 LeNet 网络的实例图

2.3 本章小结

本章首先对卷积神经网络的特性：局部连接、权值共享、池化操作进行了详细的阐述，以及深入研究了卷积层、池化层的工作机理、常用衡量模型好坏的损失函数，尤其是本论文后面即将使用的 Softmax 损失函数。然后以 Caffe 平台为例，说明了设计卷积神经网络架构时，如何对图像数据进行存储，输入给卷积神经网络的数据如何计算传递，以及 Caffe 是如何把存储的数据与各种层级单元组合成相应的网络，以便执行相应的任务等。为后续算法的研究奠定了理论和实验基础。

第3章 基于捷径的卷积神经网络的人脸识别

卷积神经网络在特征提取方面表现出优越的性能，它的局部连接、权值共享、池化操作等特性不仅使网络能够提取出对图像具有平移、缩放、扭曲不变性的特征而且极大的降低了参数量。但是经过多次池化操作会致使部分细节特征丢失。而从人类识别图像的角度出发，给定一幅自然场景下的图像，人眼会首先从这幅图像中找到感兴趣的区域，进而利用有区分性的细节特征来识别图像中的物体^[43]，并且如果能够综合利用高层次深度特征和浅层次的边缘特征能够改善识别性能^[44]。虽然深度卷积神经网络（DCNN）对不同任务，如目标识别^[45]，目标检测，目标跟踪^[46]等有着极其出色的特征提取性能。然而基于传统的卷积神经网络难以有效的将低层次特征与高层次特征进行融合，进而提取出不仅具有可识别性而且具有可区分性的特征。为了增加深层次特征的识别能力，本章借鉴 DeepID 网络^[47]及深度残差网络结构的思想，在传统的 Alexnet 卷积神经网络的基础上，提出一个基于捷径连接和惩罚函数的卷积神经网络模型，提高人脸识别的准确性。

3.1 基于捷径卷积神经网络算法

卷积神经网络一般由输入层、多个交替连接的卷积层和池化层、全连接层、以及输出层组成的特殊的深层前向神经网络模型。典型连接方式如图 3-1 中的所示。

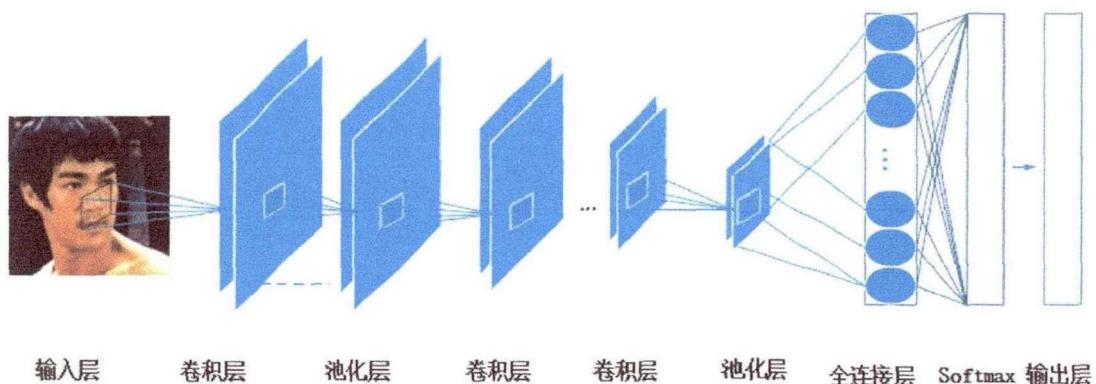
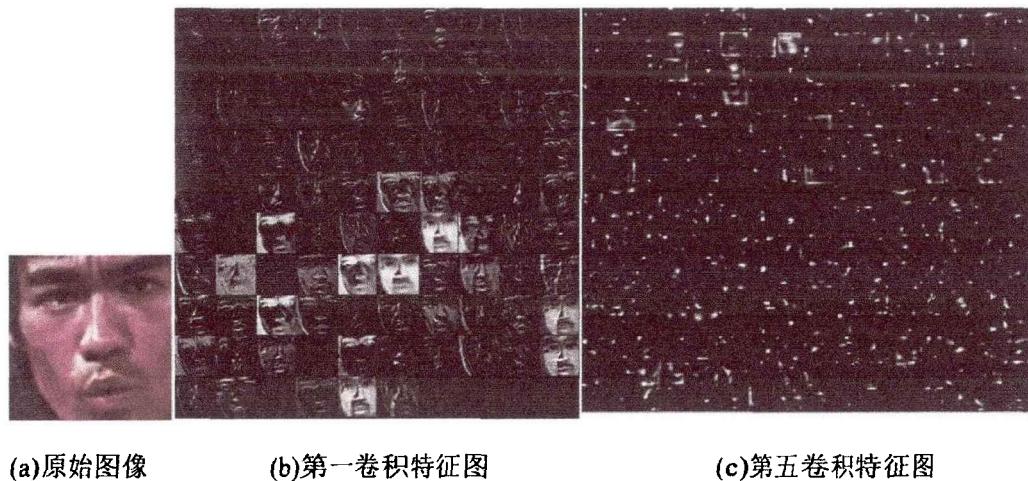


图 3-1 CNN 连接框图

为了分析每层卷积在整个卷积神经网络提取什么样的特征，将卷积层特征进行了可视化处理，如图 3-2 所示。



(a)原始图像 (b)第一卷积特征图 (c)第五卷积特征图

图 3-2 卷积特征可视化效果

从图 3-2(b)知第一卷积层主要提取人脸图像的轮廓、边缘等浅层信息，从第一卷积层到第五卷积层，卷积层提取的特征由整体特征逐渐变化到提取局部特征，从浅层特征逐渐变化为深度特征。人脸识别获得的特征不仅要具有可分性而且要具有可识别性，虽然基于单独的 softmax 损失函数的分类器能将已获得的特征分离，但是这些分离好的特征对于人脸识别任务还不够有效，特征的可识别性相对较差。Alexnet 网络经过多个池化进行降维的同时，减少了后面各卷积层包含神经元的数量，使部分信息在传播过程中损失掉，这样就会降低识别的准确率。本章提出的基于附加惩罚函数捷径连接的卷积神经网络模型将前面一层的卷积特征直接越过中间的卷积层和池化层，直接连接到最后的全连接层，其连接效果图如图 3-3 所示。而在第一个卷积层和最后的隐含层之间使用捷径连接可以有效降低传播信息的损失。

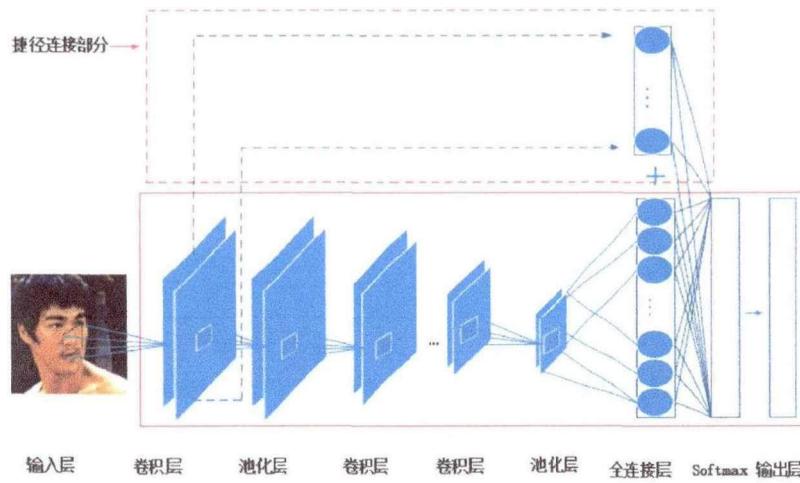


图 3-3 捷径连接框图

3.1.1 捷径连接网络

对于输入第 k 个人脸样本，捷径连接网络模型的学习过程如式 (3-1)、(3-2)、(3-3) 和 (3-4) 所示。由于池化操作对前面的特征图信息进行了聚合统计操作，但这种聚合操作在一定程度上降低计算量的同时也会带来部分细节信息的丢失，例如对前面的特征图进行 $n \times n$ 的最大池化操作后，仅保留了值最大的那个特征信息，其余 $n \times n - 1$ 个特征信息均丢弃掉了。而通过式 (3-2) 将第 k 个样本第一隐含层的输出特征同第 k 个样本第 9 隐含层进行特征融合后可以有效降低经过多次池化操作带来的信息损失，从而可以得到更完备的深度特征，进而可以在一定程度上提高识别的准确率。

$$h_{1,j}^k = \delta(w_{1,j}^k * x^k + b_{1,j}^k) \quad (3-1)$$

$$H_{10}^k = (h_{1,1}^k, h_{1,2}^k, \dots, h_{1,32}^k, H_9^k) \quad (3-2)$$

其中 $h_{1,j}^k$ 表示第 k 个样本的第一个隐层的第 j 个卷积面， $w_{1,j}^k$ 和 $b_{1,j}^k$ 分别表示第 k 个样本第一隐含层的第 j 个卷积面的卷积核和偏置， H_9^k 为第 9 层的输出向量， H_{10}^k 表示第 k 个样本的第 10 隐含层，将第 k 个样本第一个隐含层（卷积层）的输出和第 k 个样本第 9 隐含层的输出进行融合后结果向量作为第 k 个样本第 10 个隐含层的输入。

$$H_{11}^k = \delta(W_{11}^k * H_{10}^k + b_{11}^k) \quad (3-3)$$

$$H_{12}^k = \delta(W_{12}^k * H_{11}^k + b_{12}^k) \quad (3-4)$$

H_{11}^k 和 H_{12}^k 分别表示第 k 个样本的第 11 个隐含层和第 12 个隐含层的输出向量， W_{11}^k 和 b_{11}^k 分别表示第 10 层和第 11 层之间的连接权重和偏置。 H_{10}^k 表示第 k 个样本第 10 个隐含层的输出向量， $\delta(\bullet)$ 为 PReLU (Parametric ReLU) 激活函数。

3.1.2 附加惩罚函数

人脸识别任务中，通过 CNN 学习到的特征类间距不像场景、行为识别等识别任务中得到的特征类间距那么大，因此在单独使用 softmax 损失函数时，识别的性能会有所下降。为了增强对已学习到的深度特征的识别能力，对原有的 softmax 损失函数 J_s 添加一个惩罚项如式(3-5)所示。

$$J = J_s + J_c = -\sum_{i=1}^n \log \frac{e^{y_i}}{\sum_{j=1}^n e^{y_j}} + J_c \quad (3-5)$$

$$J_c = \frac{1}{2} \sum_{i=1}^m \|x_i - c_i\|_2^2 \quad (3-6)$$

$$c_{i+1} = c_i + \Delta c_i = c_i + \alpha * \frac{\partial J_c}{\partial x_i} \quad (3-7)$$

$$\alpha = \frac{\sum_{j=1}^m \Gamma(j = i)}{1 + \sum_{j=1}^m \Gamma(j = i)} \quad (3-8)$$

增加惩罚项的目的在于将已学习到的同类特征到该类特征中心之间的距离最小化。 J_c 通过计算深度特征与该类深度特征中心之间的距离来对原有的 softmax 损失函数加以惩罚，其形式如式 (3-6) 所示，并采用式 (3-7) 和式 (3-8) 所示的形式进行特征中心的更新。通过式 (3-5) 并采用随机梯度下降法进行求解优化，随着不断迭代，可使同类特征距离该类特征中心的距离逐渐减小，每类特征更加汇聚于一个区域，对于一般样本，同时会使不同特征之间的距离进一步增大，进而更容易把每种类别区分出来，也就在一定程度上提高了识别的准确性。式(3-6)中表示第 i 类深度特征， c_i 表示第 i 类深度特征的中心，其值为每次迭代时相应类别的平均特征，同时为了避免少数已失去标签样本数据的干扰，添加一个参数 α 来控制深度特征中心的学习率， α 按照式 (3-8) 的形式选取，采用式 (3-7)、(3-8) 的方法进行更新，式 (3-8) 中 $\Gamma(\bullet)$ 为条件函数，如果条件满足函数值为 1，如果条件值不满足函数值为 0。这样经过多次迭代后，同一类别的特征同 c_i 之间的距离逐渐缩小并逼近于 c_i ，因此每类特征可识别性更强，更有利 softmax 的分类。损失函数 J 可以通过 SGD (Stochastic Gradient Descent) 算法进行优化求解，求解过程如算法 1 所示。其算法的整体流程如图 3-4 所示。

算法 1. 惩罚函数的优化算法

Input: 训练数据集特征向量 $\{x_i\}$ ，初始化卷积层参数 θ 、每类特征的中心 c_i 、学习率 μ 、迭代次数 $t:=0$

- 1: **while** not converge **do**
- 2: $t:=t+1$
- 3: 计算损失函数 J 和 J_c
- 4: 计算反向传播误差 $\frac{\partial J}{\partial x_i}$ 和 $\frac{\partial J_c}{\partial x_i}$

5: 通过 $c_{i+1} = c_i - \Delta c_i = c_i - \frac{\sum_{j=1}^n \Gamma(j=i)}{1 + \sum_{j=1}^n \Gamma(j=i)} * \frac{\partial J_c}{\partial x_i}$ 更新 c_i

6: 通过 $\theta_{i+1} = \theta_i - \mu * \sum_i \frac{\partial J}{\partial x_i} \frac{\partial x_i}{\partial \theta_i}$ 更新 θ_i

7: end while

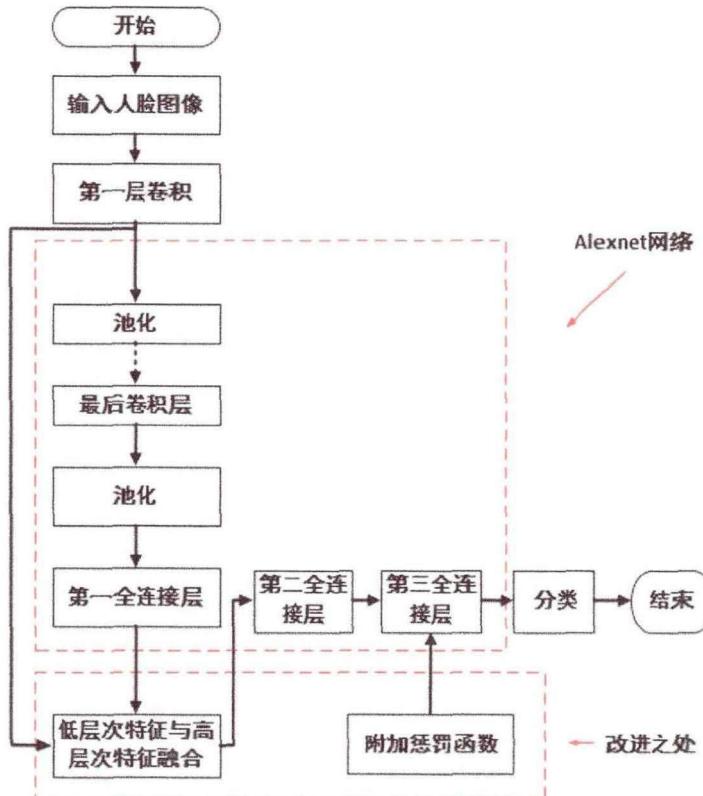


图 3-4 人脸识别算法流程图

3.2 图像预处理

为了验证基于附加惩罚函数捷径连接网络结构的性能，本文将其与具有相同卷积层和池化层的网络结构进行了对比实验。实验采用了 CASIA-webface^[48]、Facescrub^[49]人脸数据集。其中 CASIA-webface 数据集包含了 10575 个人，共计 494414 张人脸图像，每个人均有多张不同的照片，Facescrub 数据集是由 265 位男性和 265 位女性组成的 106863 张人脸图像，每人大概有 200 张图像。

CASIA-webface、Facescrub 原始人脸数据集中包含了一些与人脸无关的信息，因此首先利用 Sun^[50]等人提出的面部检测算法检测出每张人脸图像的关键点，包

括 2 个眼部中点、鼻尖和 2 个嘴角，从而得到每张人脸的边框。然后把只包含人脸面部的图像剪切出来作为网络模型的训练集和测试集。其中每个人脸集均是随机选取 70% 作为训练集剩余的 30% 作为测试集。两个人脸数据集中的人脸图像检测和剪切效果如图 3-5 和图 3-6 所示。



(a)人脸库的原图



(b)剪切后的人脸面部图

图 3-5 CASIA-web 人脸预处理



(a)人脸库原图



(b)剪切后的人脸面部图

图 3-6 Facescrub 人脸预处理

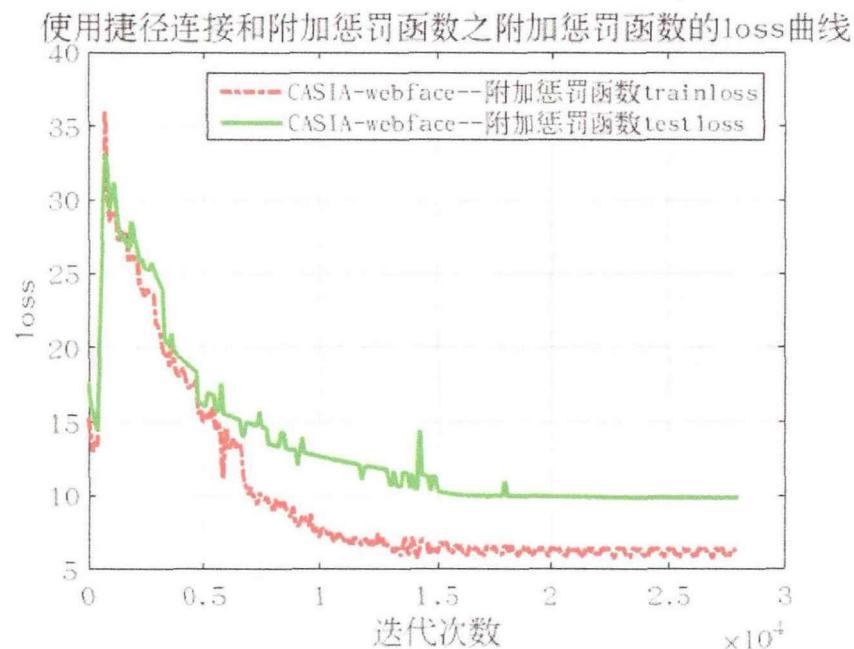
3.3 实验结果与分析

3.3.1 CASIA-web 数据集实验结果

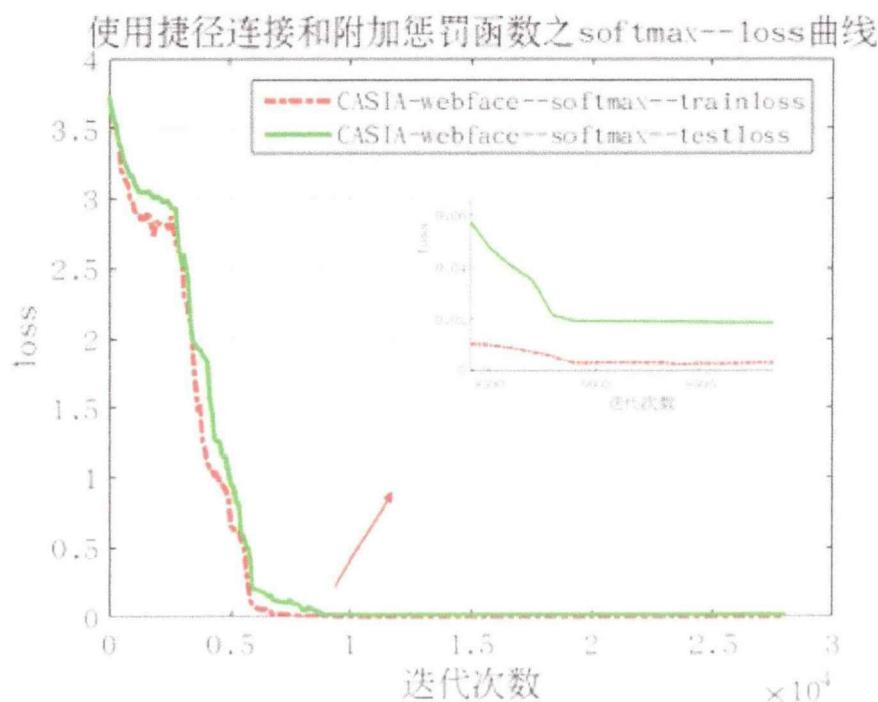
表 3-1 为本文模型及现有的算法模型在 CASIA-web 数据集上的识别效果，图 3-7 为本文模型在 CASIA-web 数据集上损失函数的收敛情况。

表 3-1 各种模型在 CASIA-web 数据集上的识别准确率

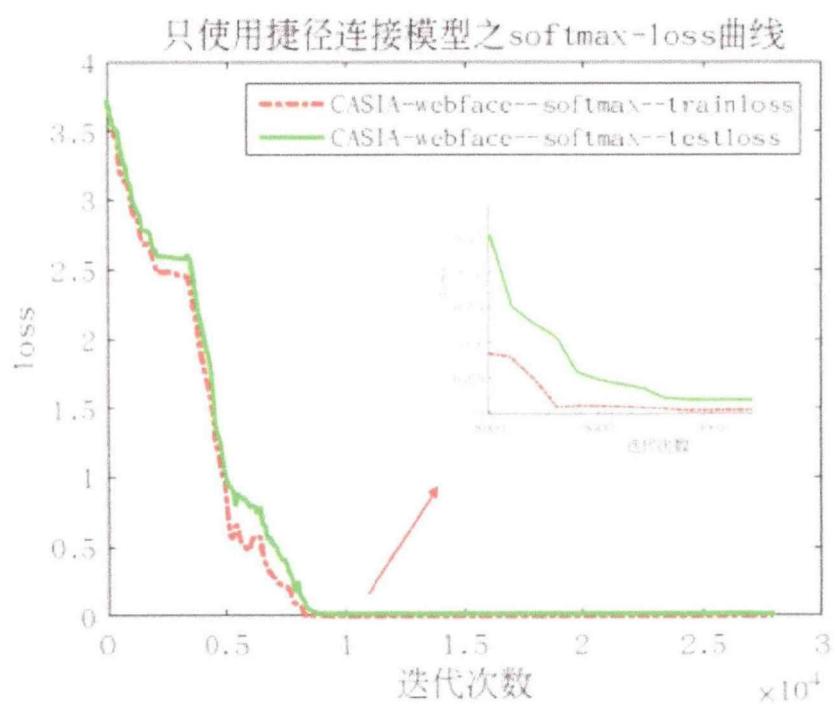
模型	CASIA-web
本文算法	98.90%
无附加惩罚函数的捷径连接模型	98.20%
Alexnet 算法	97.50%
DCNN ^[51]	97.45%
Yi et al. ^[52]	97.73%
Wang et al. ^[53]	98.23%



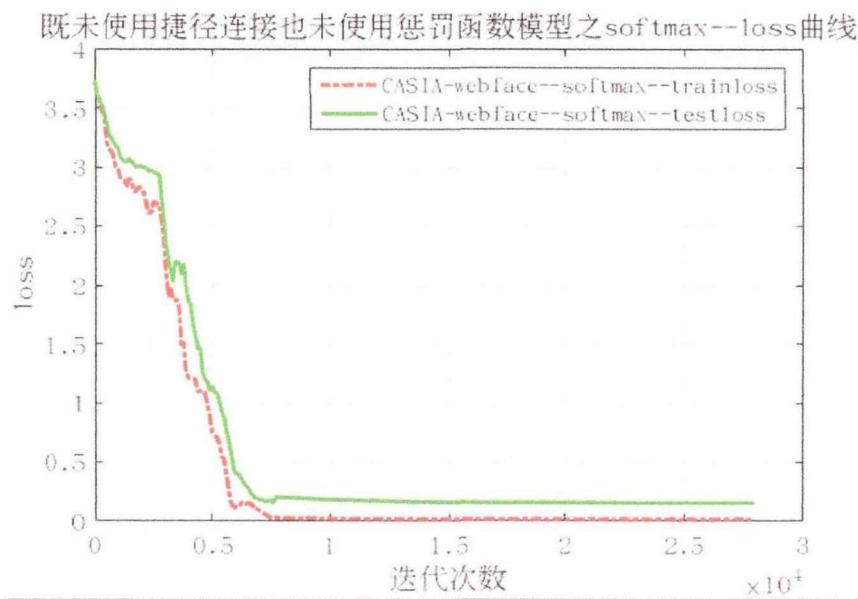
(a)本文算法惩罚函数的 loss 曲线



(b) 本文算法的 softmax-loss 曲线



(c) 只使用捷径连接的 softmax-loss 曲线



(d) Alexnet 算法的 softmax-loss 曲线

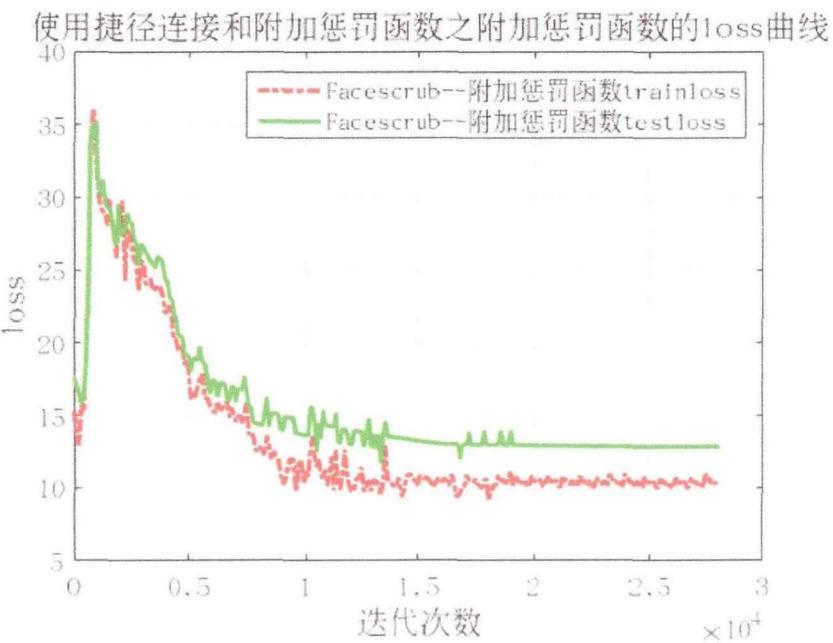
图 3-7. CASIA-web 数据集上的损失函数曲线

3.3.2 Facescrub 数据集实验结果

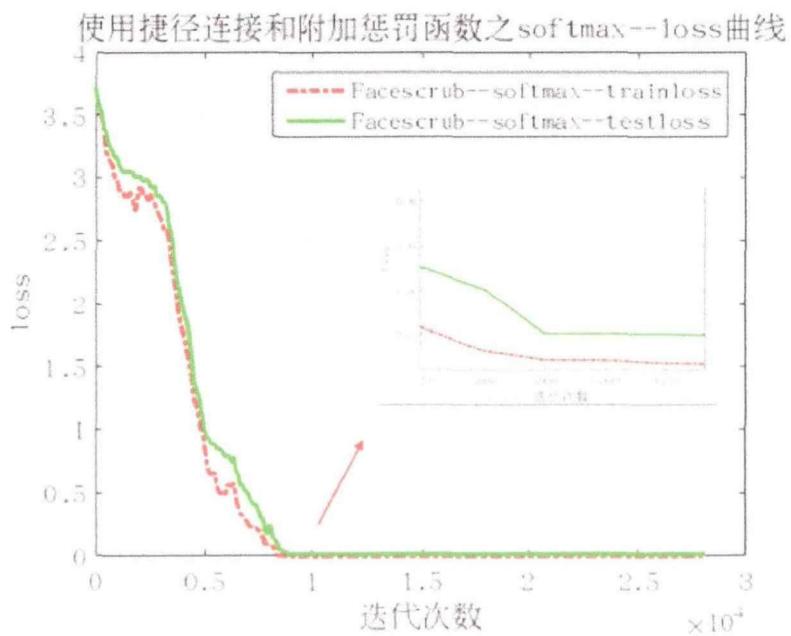
表 3-2 为本文模型及现有的算法模型 Facescrub 数据集上的识别效果，图 3-8 为本文模型在 Facescrub 数据集上损失函数的收敛情况。

表 3-2 各种模型在 Facescrub 数据集上的识别准确率

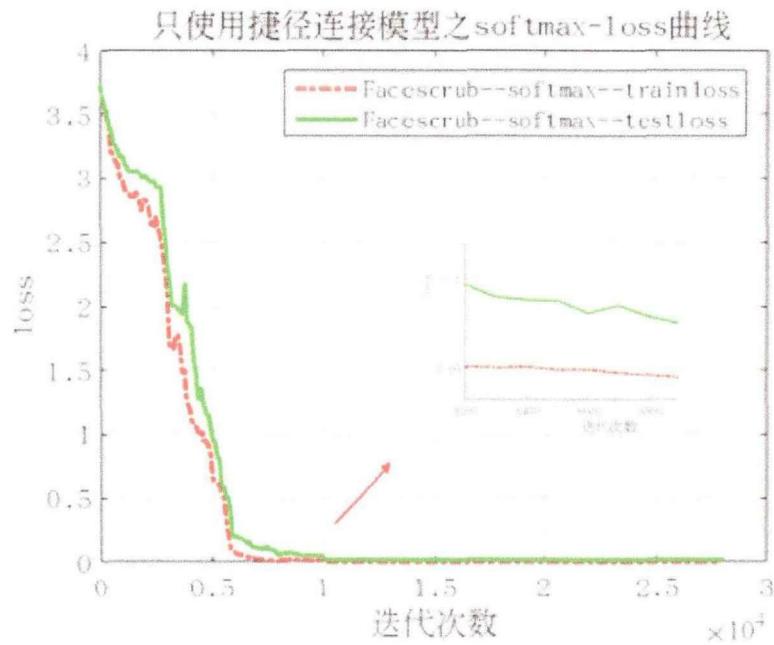
模型	Facescrub
本文算法	98.50%
无附加惩罚函数的捷径连接模型	98.00%
Alexnet 算法	97.00%
one-vs-one ^[54]	80.10%
one-vs-rest ^[55]	80.30%
Hayat et al. ^[55]	91.50%
GRCC ^[55]	92.67%
NEC ^[56]	78.66%



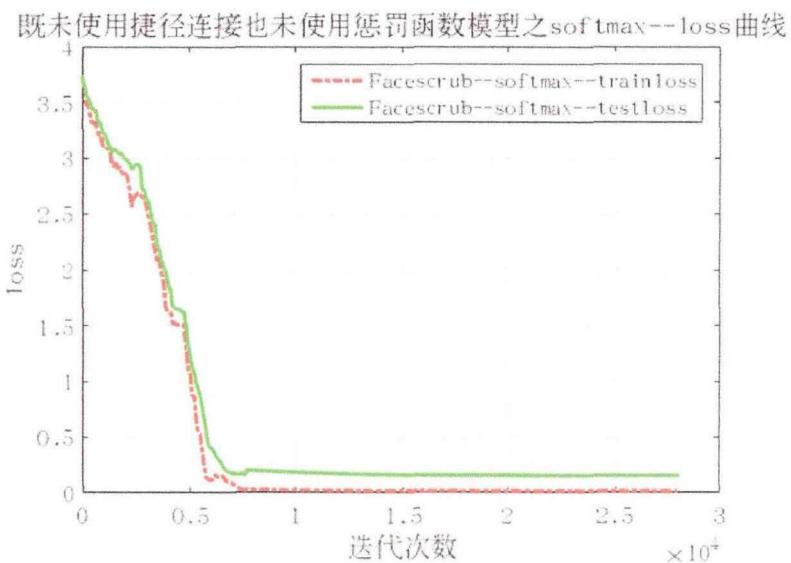
(a) 本文算法惩罚函数的 loss 曲线



(b) 本文算法的 softmax-loss 曲线



(c)只使用捷径连接的 softmax-loss 曲线



(d)Alexnet 算法的 softmax-loss 曲线

图 3-8 Facescrub 数据集上的损失函数曲线

3.3.3 实验结果分析

由图 3-7(a)和图 3-8(a)可以发现，由于开始时的类中心是通过随机初始化获

得，这也使得开始时类特征点距离类中心的平均距离相对比较大，如图 3-7(a)和 3-8(a)所示，但是随着不断地迭代，可以通过学习得到一个比较理想的类中心，通过式(3-5)的使用，同类特征之间的距离最终是在减小的，同类特征与该类特征的中心距离收敛处的值比开始降低很多，达到了使同类特征之间汇聚的更紧凑的效果，从而更有利识别出不同的类别。对比图 3-7(b)和图 3-7(c)、图 3-8(b)和图 3-8(c)可以发现，惩罚函数的使用降低了 softmax 损失函数的收敛值。对比图 3-7(b)、图 3-7(c)、图 3-7(d)以及图 3-8(b)、图 3-8(c)、图 3-8(d)可知，未使用捷径连接的网络模型的 softmax 损失函数测试曲线收敛值要比使用捷径连接模型的损失函数测试曲线收敛值大很多，证明图像经过多次卷积池化后部分特征会出现丢失的现象。算法使用了式(3-6)所示的惩罚函数，使得同类特征易汇聚在一起，更易于分类器对其进行分类，故如图 3-7(b)、图 3-7(c)和图 3-8(b)、3-8(c)所示，带有惩罚函数模型的损失函数值要比没有使用惩罚函数模型的损失函数值低，降低了错误分类的概率。

3.4. 本章小结

本章从特征丢失及损失函数的角度出发，对卷积神经网络进行了研究，提出了一种带有附加惩罚函数捷径连接的人脸识别模型，综合考虑了浅层特征和深层次特征，并将二者进行了融合，降低了经过多次池化操作过后部分特征的丢失程度，通过添加的惩罚函数来降低前面已学习到的同类深层次特征的类内距离，然后再使用分类器对其进行分类，提高了分类的准确率。

第4章 基于多尺度残差网络模型的人脸识别

由于获取人脸图像的远近、获取角度的不同，致使同一个人脸图像的特征会出现很大的差异，从而不利于进行人脸识别，而传统的 CNN 不能够充分利用人脸图像的多尺度信息^[56,57]，每层只能提取一种尺度的特征信息，为了得到更多的特征，通常的做法是加深网络层数，容易出现过拟合，也意味着计算资源的增加，导致网络的训练优化变得复杂。针对这些问题，本章提出多尺度残差网络模型（Multi-scale Residual Nets, MResNet），网络每一层通过并联不同大小的卷积核，以充分提取不同尺度的人脸特征信息，同时采用残差模块的思想，使得网络更容易训练优化。

4.1 基于多尺度残差网络模型

在图像特征提取中，单一尺寸的滤波器一次只能提取一种尺度的特征，如果在同一模块层同时使用多种尺寸的卷积核，可以增加对图像变化的鲁棒性。同时将一个大卷积核分解为非对称小卷积核，可以有效降低需要学习的参数。根据上述设计思想，本章提出如下四种设计卷积神经网络的策略：

- a. 使用 1×1 的卷积核，构成跨通道卷积，压缩特征数量。
- b. 同一模块并联使用多种卷积核，构成多尺度模块来增加鲁棒性。
- c. 将大的卷积核分解成小卷积核，降低参数计算量。
- d. 使用跨层连接，缓解梯度消失问题。

4.1.1 多尺度跨通道卷积

对一幅图像而言，图像数据的临近区域相关性极高，如图 4-1(a)的嘴周围的图像特征的相关性要远高于嘴部与眼睛部位的图像特征的相关性；对同一幅图像的不同通道而言，每一通道对应的空间位置的图像特征的相关性也要高于不同空间位置特征向量的相关性，如图 4-1(b)、4-1(c)、4-1(d)对应图像中嘴部的特征相关性要远高于三幅图像中非相同空间位置的特征的相关性。因而可以通过 1×1 的卷积核将这些相关性高的特征连接在一起，进而可以通过构成跨通道卷积来压缩输出特征图的数量。激活函数采用非线性修正单元（ReLU），网络结构如图 4-2 所示。其中 1×1 卷积得到的特征图可以按式(4-1)计算得到。



(a)原始图像

(b)R 通道图像

(c)G 通道图像

(d)B 通道图像

图 4-1 RGB 多通道图像

$$f_{i,j,k} = \max(W_k^T * x_{i,j,k} + b_k, 0) \quad (4-1)$$

$f_{i,j,k}$ 表示 1×1 卷积核对上一层特征图卷积得到并经过激活函数处理后的特征图, $x_{i,j,k}$ 表示第 k 通道上第 i 行第 j 列对应的元素值, b 表示偏执项, k 表示通道的索引。 W_k^T 表示对应第 k 通道第 i 行第 j 列元素对应卷积核的权重。

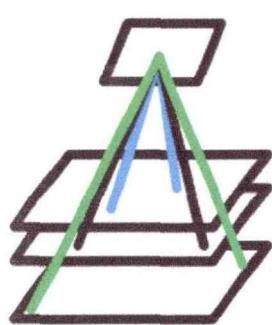


图 4-2 跨通道卷积特征图

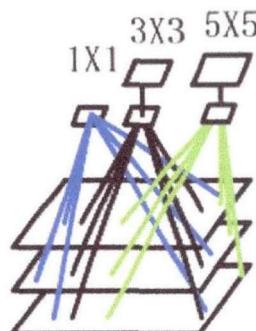
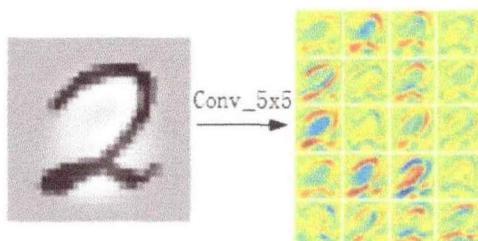
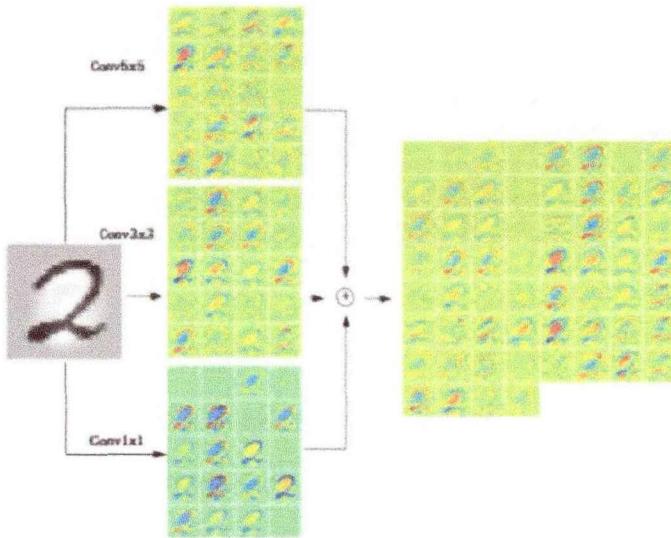


图 4-3. 多尺度跨通道卷积特征图

为了在同一层能够得到多尺度的特征, 在 Mnist 数据集上通过同一模块层同时使用不同尺寸的卷积核, 提取不同尺度的特征, 提高对尺度变化的适应性, 不同尺寸卷积核提取特征的过程如图 4-3 所示。为了验证设计原则 a、b 的有效性, 本文对 Lenet^[58] 网络进行改进, 分别并联使用 1×1 、 3×3 、 5×5 三种不同尺寸的卷积核, 网络模型的第一层卷积特征图可视化如图 4-4 所示, 可知不同尺寸卷积核得到的特征图不一样, 充分说明可以提取不同尺度的特征。网络结构和测试结果如表 4-1 所示, Lenet2 网络的准确率最高, 故并联使用不同尺寸的卷积核的网络性能要优于原始使用单一尺寸网络的性能。



(a)原始 Lenet 第一卷积模块可视化特征图



(b)Lenet2 第一卷积模块可视化特征

图 4-4. 原始 Lenet 以及多尺度变体第一卷及模块层可视化特征图

表 4-1 原始 Lenet 及其多尺度跨通道卷积变体网络的参数与结果

Layers	Conv1	Conv2	Accuracy
Original Lenet	$\begin{bmatrix} k = 5 \times 5 \\ p = 0 \times 0 \end{bmatrix}$	$\begin{bmatrix} k = 5 \times 5 \\ p = 0 \times 0 \end{bmatrix}$	99.08%
Lenet1	$\begin{bmatrix} k = 5 \times 5 \\ p = 0 \times 0 \end{bmatrix}$	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 5 \times 5 \\ p = 2 \times 2 \end{bmatrix}$	99.13%
Lenet2	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 5 \times 5 \\ p = 2 \times 2 \end{bmatrix}$	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 5 \times 5 \\ p = 2 \times 2 \end{bmatrix}$	99.19%

4.1.2 卷积核分解

卷积操作是卷积核和特征图进行的全连接操作(如图 4-5 所示)。 3×3 卷积核的级联可以获得 5×5 卷积核同样大小的感受野, 但需要计算的参数是原来的 $9/25 \approx 0.36$, 显著降低了需要学习的参数。如果在两个 3×3 卷积核之间再使用一次激活函数, 其特征表达能力比单个 5×5 的卷积核更好。本文提出将 3×3 的卷积核分解为 3×1 和 1×3 的卷积核级联, 这样将卷积核需要计算 9 个参数降为只需要计算 6 个参数, 而且比 3×3 卷积核多了一层非线性处理, 可以增强细小特征的表

达能力^[59], 3×3 卷积核分解如图 4-6 所示。 5×5 卷积核的分解如图 4-7 所示。

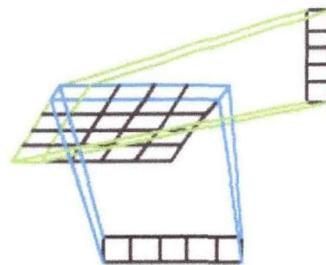
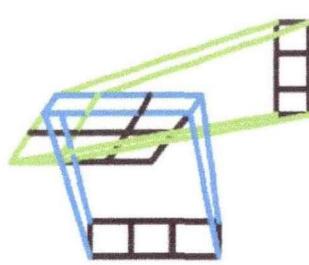
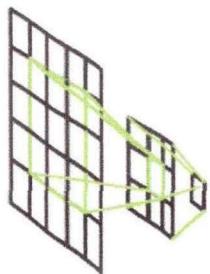
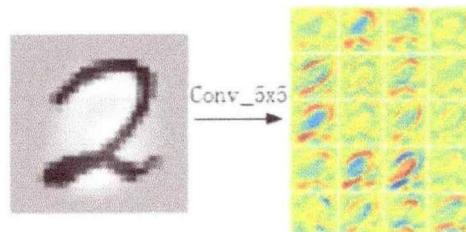
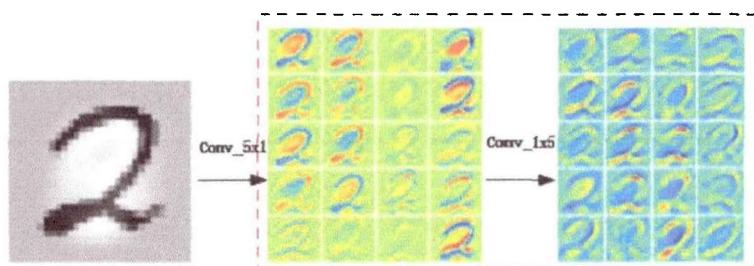


图 4-5. 卷积操作示意图 图 4-6. 3×3 卷积核分解示意图 图 4-7. 5×5 卷积核分解示意图

为了研究非对称大卷积核、对称小卷积核、非对称小卷积核对网络性能的作用, 按照前面原则 c 的设计思想, 对原始的 Lenet 网络进行了改进, 将 5×5 卷积核进行了多种形式的分解, 第一卷积模块层的可视化特征图如图 4-8 所示, 图 4-8(b)表示直接原始 Lenet 中的 5×5 的卷积核分解成 5×1 和 1×5 卷积核, 图 4-8(c)为直接将原始的 Lenet 中的每个 5×5 卷积核均分解成两个 3×3 的卷积核对应的特征图, 图 4-8(d) 是将图 4-8(c)中的卷积核进一步分解成非对称小卷积核得到的特征图, 网络参数和测试结果如表 4-2 所示。



(a). 原始 Lenet 模型



(b). Lenet3 模型

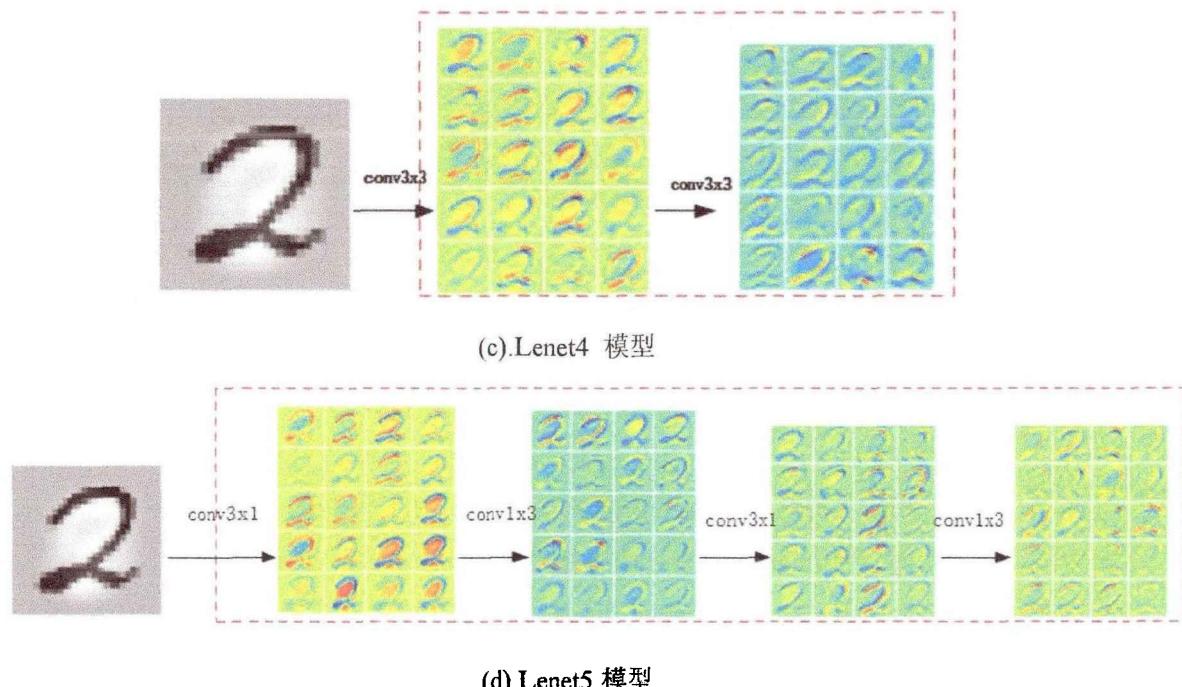


图 4-8. 原始 Lenet 以及分解卷积核变体第一卷积模块层可视化特征图

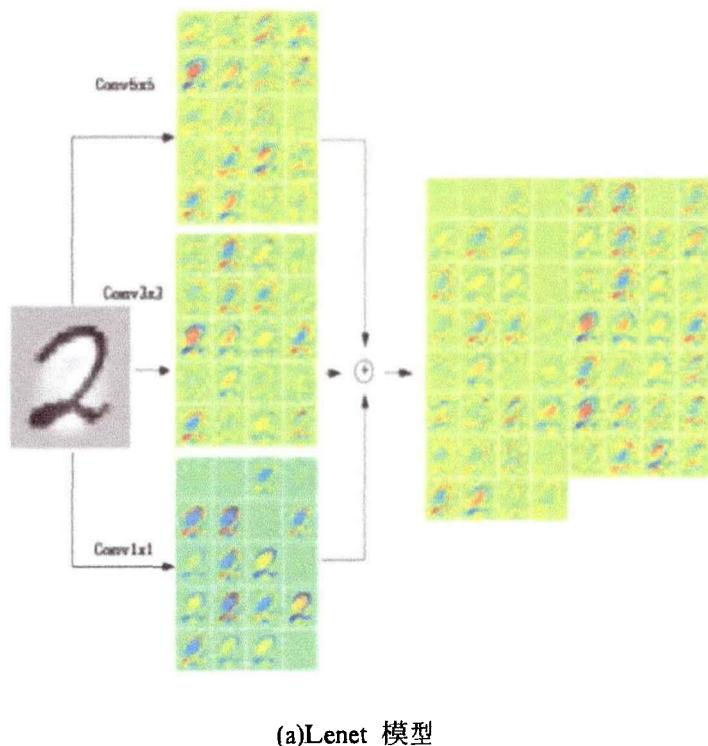
表 4-2. 原始 Lenet 及其分解卷积核变体网络架构参数与结果

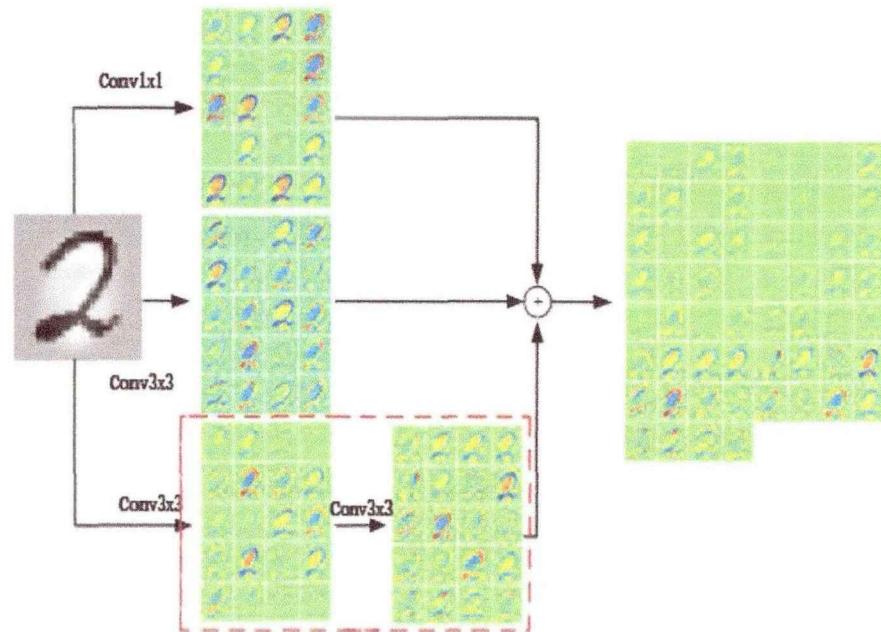
Layers	Conv1	Conv2	Accuracy
Original Lenet	$\begin{bmatrix} k = 5 \times 5 \\ p = 0 \times 0 \end{bmatrix}$	$\begin{bmatrix} k = 5 \times 5 \\ p = 0 \times 0 \end{bmatrix}$	99.08%
Lenet3	$\begin{bmatrix} [k = 5 \times 1] \rightarrow [k = 1 \times 5] \\ p = 0 \times 0 \end{bmatrix} \times 1$	$\begin{bmatrix} [k = 5 \times 1] \rightarrow [k = 1 \times 5] \\ p = 0 \times 0 \end{bmatrix} \times 1$	99.07%
Lenet4	$\begin{bmatrix} [k = 3 \times 3] \times 2 \\ p = 0 \times 0 \end{bmatrix}$	$\begin{bmatrix} [k = 3 \times 3] \times 2 \\ p = 0 \times 0 \end{bmatrix}$	99.16%
Lenet5	$\begin{bmatrix} [k = 3 \times 1] \rightarrow [k = 1 \times 3] \\ p = 0 \times 0 \end{bmatrix} \times 2$	$\begin{bmatrix} [k = 3 \times 1] \rightarrow [k = 1 \times 3] \\ p = 0 \times 0 \end{bmatrix} \times 2$	99.22%

对比表 4-2 中的四种模型以及图 4-8 中的可视化特征图可以发现：将 5×5 的卷积核直接分解成 5×1 和 1×5 卷积核，特征图的噪声增强了，模型性能低于原始 Lenet 模型。将 5×5 卷积核分解成两个 3×3 卷积核级联得到的特征图比原始 Lenet 模型的第一层卷积模块得到的特征图更平滑，有更低的噪声，分类准确率有提高。如果进一步将 3×3 的卷积核分解成两个 3×1 和 1×3 的卷积核级联，可以得到噪声更小、更平滑的特征，分类准确率最高。

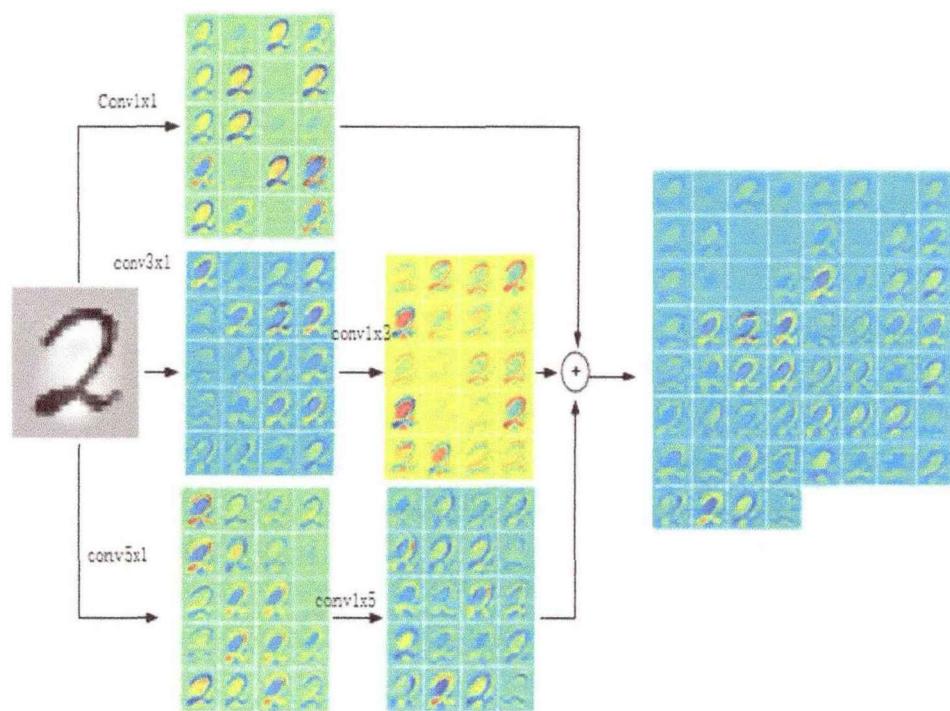
4.1.3 多尺度小卷积核

从 4.1.1 节和 4.1.2 节得知，同一卷积模块同时并联 1×1 、 3×3 、 5×5 三种感受野大小的卷积核和把 5×5 的卷积核分解成两个 3×3 或者分解成两对 3×1 、 1×3 的卷积核均有有助于网络模型性能的提升，因而将 Lenet2 网络中的 5×5 的卷积核分解成两个 3×3 的卷积核得到 Lenet7 网络，将 Lenet2 中的 3×3 、 5×5 分解成非对称的 3×1 和 1×3 、 5×1 和 1×5 得到 Lenet8，进而将 Lenet7 中的 3×3 卷积核分解成 3×1 和 1×3 得到 Lenet9，对应的第一卷积模块可视化特征如图 4-9 所示，图中虚线框的感受野同 5×5 卷积核等效，但比单个 5×5 卷积核的特征表达能力更强，同时需要学习的参数大幅度降低。图 4-9 (d) 中同时并联使用不同尺寸的卷积核能提取的特征数量更多，而且最后得到的特征层次相对更抽象。网络架构参数和测试结果如表 4-3 所示。可知 Lenet9 模型使用的多尺度非对称小卷积核得到的特征图噪声最低，分类准确率最高，故并联使用 1×1 、 3×1 、 1×3 非对称的小卷积核的性能最优。比较表 4-1 中的 Lenet1 和表 4-3 中的 Lenet6 可以发现，将 5×5 的卷积核分解成 3×3 的卷积核提高了分类准确率，从表 4-3 中的 Lenet7 和 Lenet8 以及图 4-9 中对应的可视化特征图，可知 Lenet7 卷积模块得到的特征图噪声低于 Lenet8 的卷积模块，且特征图更平滑。

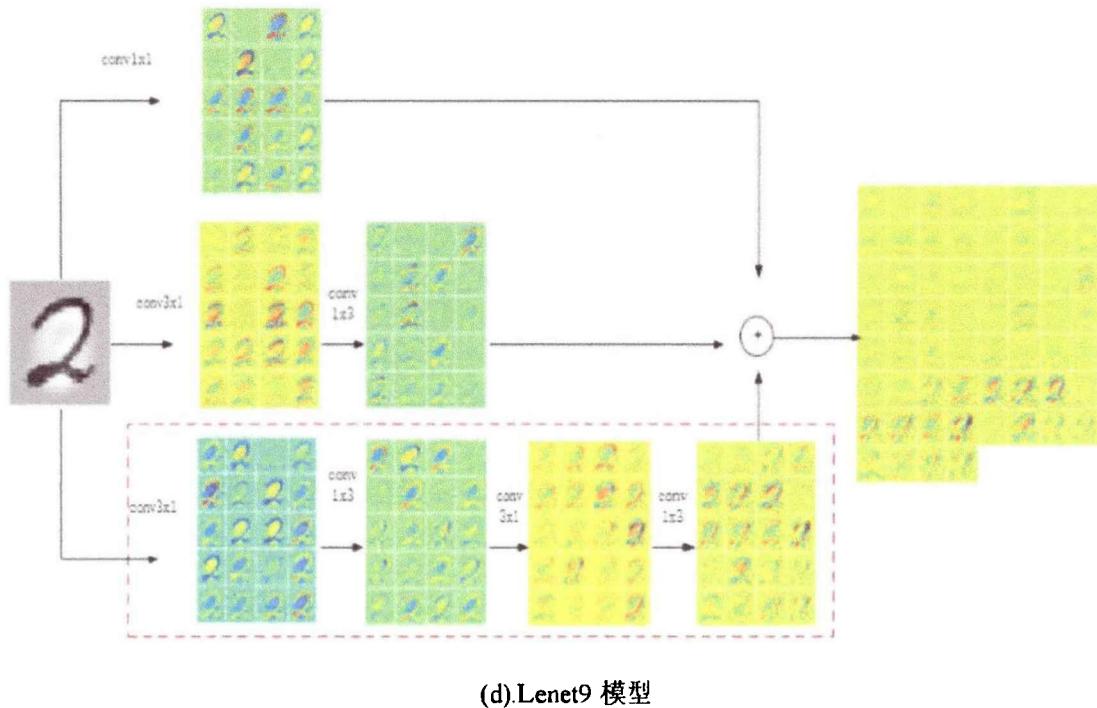




(b)Lenet7 模型



(c)Lenet8 模型

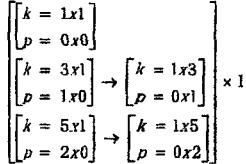
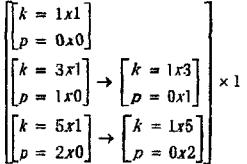
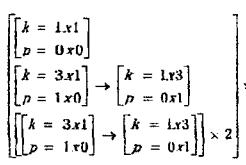
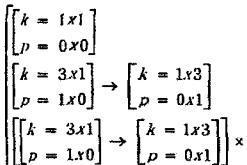


(d) Lenet9 模型

图 4-9 原始 LeNet 以及多尺度小卷积核变体第一卷积模块层可视化特征图

表 4-3. 多尺度小卷积核变体网络架构与结果

layers	Conv1	Conv2	Accuracy
Lenet2	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 5 \times 5 \\ p = 2 \times 2 \end{bmatrix}$	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 5 \times 5 \\ p = 2 \times 2 \end{bmatrix}$	99.08%
Lenet6	$\begin{bmatrix} k = 5 \times 5 \\ p = 0 \times 0 \end{bmatrix}$	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 2$	99.15%
Lenet7	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 2$	$\begin{bmatrix} k = 1 \times 1 \\ p = 0 \times 0 \end{bmatrix}$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 1$ $\begin{bmatrix} k = 3 \times 3 \\ p = 1 \times 1 \end{bmatrix} \times 2$	99.20%

Lenet8 		99.15%
Lenet9 		99.32%

4.1.4 多尺度残差网络

4.1.1 节、4.1.2 节和 4.1.3 节验证了多尺度小卷积模块对 Lenet 网络性能提升的有效性，然而浅层网络模型对人脸图像的特征表达能力远远不够，但是网络深度的增加会导致网络参数的求解变得困难^[60]，针对此问题，提出将上一层的网络特征直接跨过多层卷积模块映射到下层网络并学习其残差来有效缓解梯度消失问题。因此首先根据 Lenet2 的设计思想和 d 设计原则，设计了图 4-10(a)的多尺度残差模块，标有 Maxout 的表示使用 Maxout 激活函数，未标出的表示使用 ReLU 激活函数。为了降低特征损失，采用步长为 2 的卷积核和步长为 2 的最大池化层对网络进行降维操作，模块如图 4-10(b)所示。图中的 n 值表示相应卷积核得到的特征图个数。本文将这种网络架构命名为 MResNet-A，其网络模型如图 4-11 所示，其中的模块 A 和模块 C 的架构如图 4-10(a)和图 4-10(b)所示。

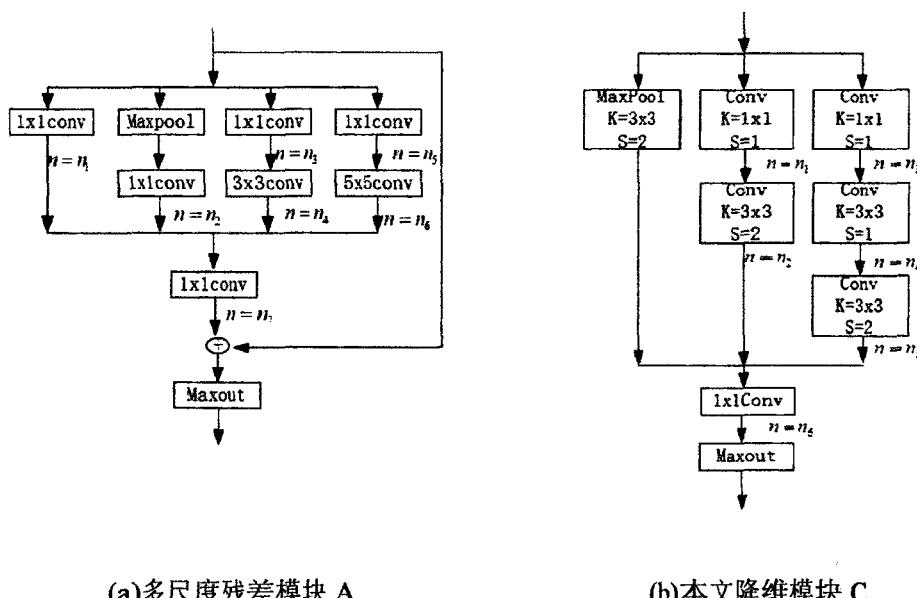


图 4-10 MResNet-A 的多尺度残差模块和降维模块

其中各个模块卷积核输出特征图个数以及每个模块的参数如表 4-4 所示, 其中 n_i , $i=1,2,\dots,7$ 表示图 9 中相应卷积核的输出特征图的个数, A_m , $m = 1, 2, 3$ 表示网络中使用的第 m 个多尺度残差模块, C_n , $n = 1, 2, 3, 4$ 表示网络中使用的第 n 个降维模块。

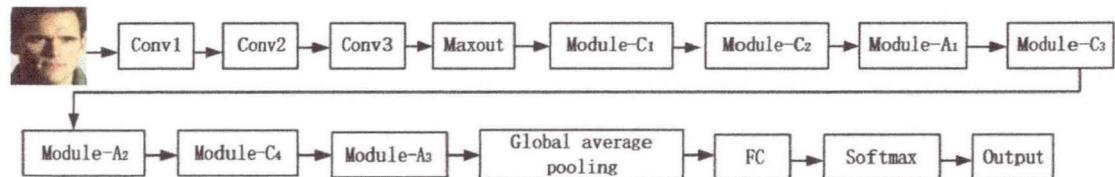


图 4-11 MResNet-A 网络模型架构示意图

表 4-4. MResNet-A 各模块卷积核的输出特征图个数及需要学习参数

Model	n_1	n_2	n_3	n_4	n_5	n_6	n_7	Parameter
C_1	32	64	32	64	128	192	--	159.744k
C_2	96	192	96	128	192	384	--	571.392k
A_1	96	96	96	128	96	192	384	841.728k
C_3	96	256	96	128	256	512	--	1024k
A_2	128	128	128	192	128	256	512	1531.904k
C_4	192	384	192	256	512	1024	--	3563.52k
A_3	256	256	256	384	256	512	1024	6127.616k

从表 4-4 可以看出, 每个模块的参数量很大, 为了进一步降低 MResNet-A 网络架构的参数, 综合 a、b、c、d 四种设计原则, 按照 Lenet9 模型架构的设计方法, 设计了如图 4-12 的多尺度小卷积残差模块 MResNet-B, 降维模块仍然使用图 4-10(b)所示的降维模块架构。图 4-12 中的多尺度小卷积残差模块各卷积核对应的输出特征个数及参数如表 4-5 所示。可以看出, 虽然模块 B_1 、 B_2 、 B_3 比对应的模块 A_1 、 A_2 、 A_3 使用了更多的卷积核, 但是参数量均要小于 A_1 、 A_2 、 A_3 的参数量, 图 4-10(a)和图 4-12 中的 1x1 的卷积核步长均为 1, 3x3、5x5、1x3、3x1 采用零填充策略。

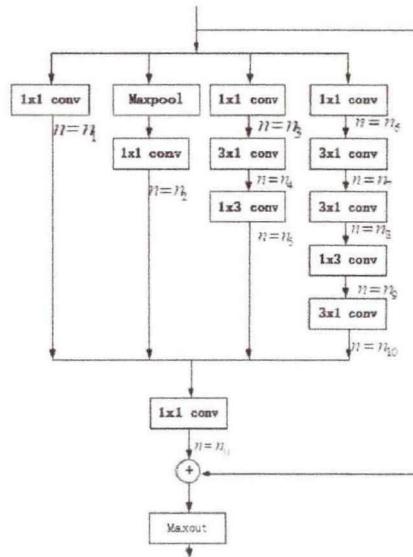


图 4-12 MResNet-B 多尺度小卷积残差模块

表 4-5. MResNet-B 各模块卷积核的输出特征图个数及需要学习参数

Model	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	Parameter
C_1	32	64	32	64	128	192	--	--	--	--	--	159.744k
C_2	96	192	96	128	192	384	--	--	--	--	--	571.392k
B_1	96	96	96	128	128	96	128	128	192	192	384	1290.688k
C_3	96	256	96	128	256	512	--	--	--	--	--	1024k
B_2	128	128	128	192	192	128	192	192	256	256	512	1204.224k
C_4	192	384	192	256	512	1024	--	--	--	--	--	3563.52k
B_3	256	256	256	384	384	256	384	384	512	512	1024	4816.896k

4.2 实验与数据分析

4.2.1 数据预处理

为了验证提出的算法性能，采用 MS-Celeb-1Mv1^[61]人脸数据集来训练 MResNet-A 和 MResNet-B 模型，MS-Celeb-1Mv1 数据集包含了 99891 个人，每个人有多张不同的图像，一共大约 10M 张图像。然后在 LFW 数据集上来评估模型的性能，LFW 数据集包含 5749 个人，一共 13233 张图像。

首先利用 MTCNN^[62]算法对 MS-Celeb-1Mv1 和 LFW 数据集检测面部的关键点，包括 2 个眼部中点、鼻尖、两个嘴角，以及每张人脸的边框信息，然后利用检测到的边框信息将人脸的面部图像裁剪出来（如图 4-13 所示）。实验将图像裁剪成 165x165，作为网络的输入训练 MResNet-A 和 MResNet-B 网络。



(a) MS-Celeb-1Mv1 人脸数据集裁剪对齐 (b) LFW 人脸数据集裁剪对齐

图 4-13 人脸图像预处理

4.2.2 模型训练与性能评估

随机选取 MS-Celeb-1Mv1 数据集的 75% 作为训练，剩下的 25% 作为验证，并将输入的 165×165 大小的图像进行随机裁剪成 149×149 的图像作为第一个卷积层的输入。初始学习率从 0.01 开始，按照步长逐渐降低学习率。

将 LFW 数据集分割成 6000 对人脸图像，每对包含两张图像，直接使用在 MS-Celeb-1Mv1 数据集上训练的模型文件来提取每张图像的特征，为了同现有算法进行性能比较，采用 10 倍交叉验证计算这两个特征的余弦距离的相似性。表 4-6 展示了算法以及现有算法在 LFW 上的识别效果，图 4-14 为算法在 LFW 数据集上的 ROC 曲线。

表 4-6 各算法在 LFW 上的评估性能

Algorithm	Accuracy
MResNet-A	98.26%
MResNet-B	99.13%
DeepFace ^[63]	98.95%
Webface ^[52]	97.73%
Web-scale ^[64]	98.37%
VIPLFaceNet ^[65]	98.62%

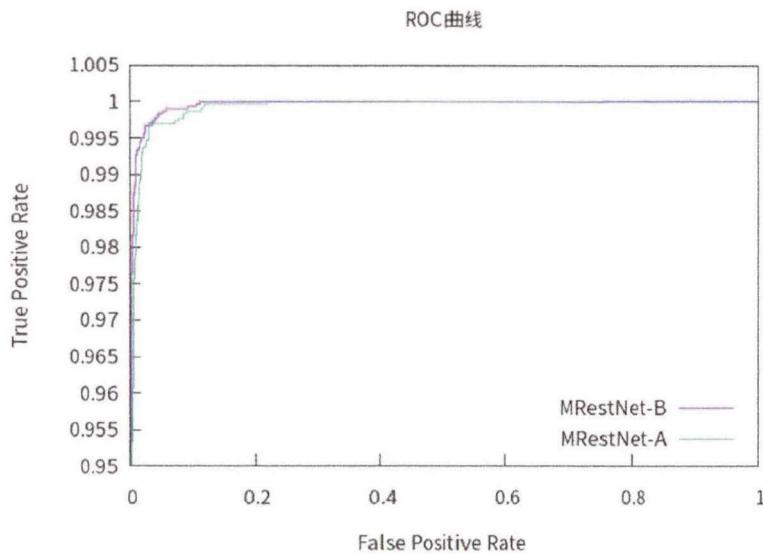


图 4-14. MResNet-A 和 MResNet-B 在 LFW 上的 ROC 曲线

4.2.3 实验结果分析

通过图 4-10、图 4-12，可以发现模型 MResNet-B 中的每个残差模块比模型 MResNet-A 的每个残差模块使用了更多的卷积核，级联了更深的层数，从而提高了特征的表达能力，从表 4-6 可以得知 MResNet-B 的性能要比 MResNet-A 提高近 1%，而对比表 4-4 和表 4-5 可以发现，MResNet-B 的参数计算量要小于模型 MResNet-A，充分说明将大卷积核分解成小卷积核，不仅能够降低参数计算，而且能够提高网络架构的特征表达能力，图 4-14 的分析可以发现，MResNet-B 模型的 ROC 曲线与横坐标所围成的面积要高于 MResNet-A 模型的 ROC 曲线与横坐标围成的面积，间接说明了 MResNet-B 网络模型的诊断准确性高于 MResNet-A 模型的准确性。

4.3 人脸识别系统测试

4.3.1 机器识别一般过程

机器识别的一般流程如图 4-15 所示，需要先给机器一幅事先标好标签的图像，机器通过相应的算法学习出图像的特征，摄像头采集到一幅未知图像，机器会利用同样的算法提取出该图像的特征，然后将未知图像的特征与已知图像的特征进行比较，如果二者之间的距离在给定阈值范围内，则机器认为采集到的未知

图像和同它进行比较的已知图像是同一个人或者物。因此，首先应该对图像进行采集，并将其标号标签放进相应的数据库。

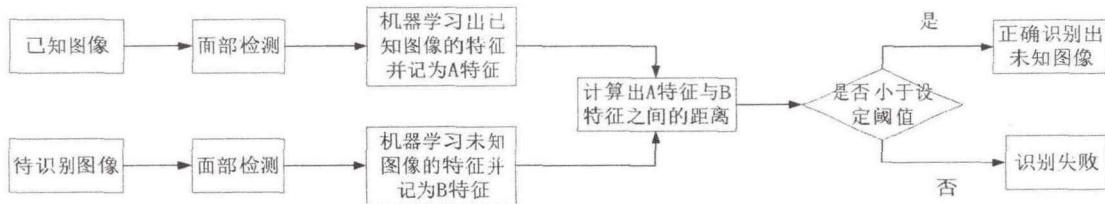


图 4-15 机器识别过程

由于采集好的图像可能不仅包含面部，而且有时还包含人体的其他部位以及人所处的环境中的其他干扰物体，这些干扰因素不仅会增加一些无关的特征向量而且会大大降低最终算法的识别性能。因而对图像进行特征提取之前，我们还需要先对图像进行面部检测，此处采用第 4.2.1 小节中的方法进行面部关键点检测。然后将图像中的人脸定位出来。之后将定位裁剪出的人脸送入特征提取器，本测试系统的特征提取器采用 MResNet-B 模型。

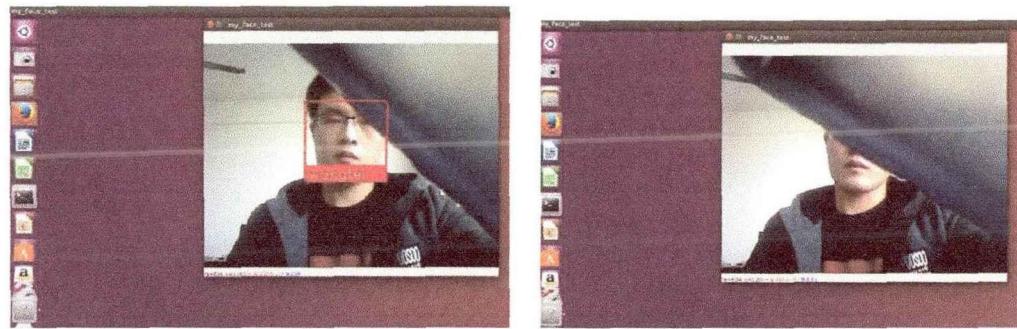
4.3.2 识别系统测试

按照 4.2 节的思想，本小节综合使用 Caffe、Python，在 Ubuntu16.04 系统下编写了人脸识别系统，通过摄像头读取人脸图像，分别从不同角度、不同尺度、对目标图像是否遮挡等场景下对多尺度残差网络模型进行了测试，测试结果如图 4-16 所示。



(a) 远距离人脸图像测试效果

(b)近距离不同角度测试效果



(c) 目标图像被遮挡小部分测试效果 (d) 目标图像多半被遮挡测试效果

图 4-16 多尺度残差模型软件测试效果

对比图 4-16(a)和图 4-16(b)，可以发现，本算法对远近不同的目标图像及不同角度的图像均能够正确识别出来目标图像，证明算法具有很好的鲁棒性，对比图 4-16(c)和图 4-16(d)可以发现，当目标图像少部分被遮挡时，目标图像仍能够被正确识别出来，而当目标图像一半以上被遮挡时，目标图像识别失败，因此多尺度残差网络模型对遮挡问题也具有一定的鲁棒性。

4.4 本章小结

本章提出在每一卷积模块并联使用多个不同大小的卷积核提取不同尺度的特征，增加了模型对复杂环境变化的适用性，并通过将大卷积核分解成小卷积核级联降低模型参数的计算量，且根据残差网络的思想直接将上一模块层的特征直接跨过多尺度模块映射到下一层来减少梯度消失问题带来的影响，提高了图像分类的准确率，然后针对不同的设计原则分别进行了实验，分别给出了实验结果对比，最后通过编写的人脸识别系统进一步验证了算法的性能。

第5章 总结与展望

5.1 研究工作总结

本文主要针对卷积神经网络的网络架构及惩罚函数进行了研究，主要做了以下几个方面的工作：

- 1) 针对传统的卷积神经网络经过多次池化进行降维致使部分信息损失问题，本文提出了基于附加惩罚函数的捷径连接模型：综合考虑浅层次特征和深度特征，并将二者进行融合，进而提取出更加完备的特征；对原有的 softmax 损失函数添加惩罚项，使得到的深度特征中的同类特征之间的距离逐渐减小，不同类特征之间的距离逐渐加大，在一定程度上提高了识别的准确率。
- 2) 针对图像获取角度的不同，图像获取距离远近不同，使得同一幅人脸图像差异很大，以及网络层数的增加网络学习参数急剧增加等问题，提出了非对称多尺度小卷积核网络模型：在同一模块层并联使用多种尺寸的卷积核，提取不同尺度的特征，提高网络模型对图像变化的鲁棒性；并将大卷积核分解成小卷积核、小卷积核分解成非对称卷积核的方法来降低参数量，提高网络模型的非线性表达能力，然后通过将上一层的特征直接映射到下一层并学习其残差，缓解了由于网络深度增加带来的梯度消失问题。
- 3) 为了便于验证算法的有效性，本文使用自主设计的多尺度残差网络模型、Caffe、Ubuntu、Python 实现了实时人脸识别系统，结果显示该算法对获取图像的角度、获取图像的远近、图像的遮挡等问题有很好的鲁棒性。

5.2 展望

卷积神经网络以其强大的特征提取能力而被广泛用于各种领域，但是在人脸识别仍存在以下问题需要解决：

- 1) 目前基于卷积神经网络的学习方法需要大量的有标注的数据集，而在实际应用中，标记大量的数据是非常困难的，因而未来可以对训练方法进行改进：采用无监督的训练方法可以让卷积神经网络模型省去对数据集的标注过程，在输入数据量较大时可以显著降低运算量。
- 2) 目前多靠经验或实验来设计用于人脸识别的卷积神经网络架构，如网络架构中特征图的数量、学习率的选择等问题，针对这些问题还需要进一步研究。
- 3) 对于某些目标识别任务，如人脸识别任务，为了获得更好的性能，研究人员

不断的加深网络的层数，有的网络层数多达千层，但是这样的网络模型的大小，参数的计算量都相当大，而一般的移动设备的存储能力有限，限制了网络模型在移动设备上的应用。因而今后有必要针对如何降低模型的大小、参数的计算量等问题进一步深入研究。

致 谢

光阴似箭，日月如梭，三年的硕士生活即将结束，这三年的研究生学习与生活，让我获益良多，在论文完成之际，首先发自内心的感谢我的导师张莹副教授，本论文的研究工作之所以能够顺利完成，离不开张莹老师的悉心指导，张老师思维活跃，治学严谨，待人和蔼可亲。三年的研究生学习期间，在科研道路上，感谢张老师为我指明了研究方向，指引着我前行；生活上，张老师的关怀与照顾，更是让我感觉到亲人般温暖。在此，我要对我的导师张莹老师以最真挚的感谢！

其次，感谢控制点的导师们，他们是黄辉先老师、张东波老师、汤红忠老师、兰永红老师、印峰老师、许海霞老师，感谢他们在专业课程中的耐心指导；感谢盛孟刚副教授、姚志强教授给予我再次选择研究方向的机会。

感谢同门师兄师姐师弟师妹们，他们分别是张柱、高赢、闫璠、涂勇涛、邵豪、成超鹏、薛亮、牟清萍。感谢他们在科研工作和生活上给予的关爱与帮助；感谢一同呆过三年的室友：邓文振、黄响兵、张宇平，感谢他们为我营造了一个良好的舒适的寝室氛围以及日常生活中的关心与照顾；感谢在科研与生活上一起走过的小伙伴们：高炳春、陈蓉、邵豪、袁盼盼、李雅芳、黄巍等，感谢他们的风雨与共；感谢控制点的所有同学日常的关爱；感谢316通信团队成员们：田力、郭慕文、李晗等人在学术上与生活上的帮助。

感谢我的父母与家人，感谢父母二十多年来养育和最无私付出，让我有了一个温馨舒适的家庭，让我能够在校园里追逐自己的梦想！

最后，感谢各位评审专家和老师能够抽出宝贵的时间对本论文进行评审和指导！

硕士期间学术论文及研究成果

- [1]. 王飞, 张莹, 张东波, 邵豪, 成超鹏. 基于捷径的卷积神经网络在人脸识别中的应用研究.[J].电子测量与仪器学报,2018.32(04):80-86.
- [2] 王飞, 张莹, 邵豪, 张东波, 牟清萍. 多尺度残差网络模型的研究及其应用 [J].计算机研究与发展 (外审中)

参考文献

- [1] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 2012: 1097-1105.
- [2] Phillips P J, Wechsler H, Huang J, et al. The FERET database and evaluation procedure for face-recognition algorithms[J]. Image and vision computing, 1998, 16(5): 295-306.
- [3] Turk M A, Pentland A P. Face recognition using eigenfaces[C]//Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on. IEEE, 1991: 586-591.
- [4] Belhumeur P N, Kriegman D J. What is the set of images of an object under all possible illumination conditions? [J]. International Journal of Computer Vision, 1998, 28(3): 245-260.
- [5] Belhumeur P N, Hespanha J P, Kriegman D J. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection[J]. IEEE Transactions on pattern analysis and machine intelligence, 1997, 19(7): 711-720.
- [6] Rikert T D, Jones M J, Viola P. A cluster-based statistical model for object detection[C]//Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on. IEEE, 1999, 2: 1046-1053.
- [7] Valentin D, Abdi H, O'Toole A J, et al. Connectionist models of face processing: A survey[J]. Pattern recognition, 1994, 27(9): 1209-1230.
- [8] Hjelmås E, Wroldsen J. Face recognition with multi-layer perceptrons[J]. 1997.
- [9] Cortes C, Vapnik V. Support vector machine[J]. Machine learning, 1995, 20(3): 273-297.
- [10] Lu J, Plataniotis K N. Boosting face recognition on a large-scale database[C]//Image Processing. 2002. Proceedings. 2002 International Conference on. IEEE, 2002, 2: II-II.
- [11] Blanz V, Vetter T. A morphable model for the synthesis of 3D faces[C]//Proceedings of the 26th annual conference on Computer graphics and interactive techniques. ACM Press/Addison-Wesley Publishing Co., 1999: 187-194.
- [12] Vetter T, Blanz V. Estimating coloured 3D face models from single images: An example based approach[C]//European Conference on Computer Vision. Springer, Berlin, Heidelberg, 1998: 499-513.
- [13] Liu C, Wechsler H. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition[J]. IEEE Transactions on Image processing, 2002, 11(4): 467-476.
- [14] Ahonen T, Hadid A, Pietikainen M. Face description with local binary patterns: Application to face recognition[J]. IEEE transactions on pattern analysis and machine intelligence,

2006, 28(12): 2037-2041.

- [15] Learned-Miller E, Huang G B, RoyChowdhury A, et al. Labeled faces in the wild: A survey[M]//Advances in face detection and facial image analysis. Springer International Publishing, 2016: 189-248.
- [16] Chopra S, Hadsell R, LeCun Y. Learning a similarity metric discriminatively, with application to face verification[C]//Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE, 2005, 1: 539-546.
- [17] Zeiler M D, Taylor G W, Fergus R. Adaptive deconvolutional networks for mid and high level feature learning[C]//Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011: 2018-2025.
- [18] Zeiler M D, Krishnan D, Taylor G W, et al. Deconvolutional networks[C]//Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010: 2528-2535.
- [19] Lin M, Chen Q, Yan S. Network In Network[J]. Computer Science, 2013..
- [20] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.
- [21] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//International Conference on Machine Learning. 2015: 448-456.
- [22] Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the inception architecture for computer vision[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2818-2826.
- [23] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning[C]//AAAI. 2017: 4278-4284.
- [24] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
- [25] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
- [26] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 3431-3440.
- [27] Bell S, Lawrence Zitnick C, Bala K, et al. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2874-2883.
- [28] Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017, pp. 1800-1807

- [29] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks[C]//Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. IEEE, 2017: 5987-5995.
- [30] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[J]. arXiv preprint arXiv:1709.01507, 2017.
- [31] Huang G, Liu Z, Weinberger K Q, et al. Densely connected convolutional networks[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017, pp. 2261-2269.
- [32] Sun Y, Wang X, Tang X. Deep learning face representation from predicting 10,000 classes[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1891-1898.
- [33] Parkhi O M, Vedaldi A, Zisserman A. Deep Face Recognition[C]// British Machine Vision Conference. 2015:41.1-41.12.
- [34] Wu X, He R, Sun Z, et al. A Light CNN for Deep Face Representation with Noisy Labels[J]. Computer Science, 2015.
- [35] Schroff F, Kalenichenko D, Philbin J. FaceNet: A unified embedding for face recognition and clustering[J]. 2015:815-823.
- [36] Wang Y, Liu J, Tang X. Robust 3D Face Recognition by Local Shape Difference Boosting[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2010, 32(10):1858-1870.
- [37] <https://github.com/BVLC/caffe>
- [38] <https://github.com/caffe2/caffe2>
- [39] <https://github.com/tensorflow/tensorflow>
- [40] <https://github.com/pytorch/pytorch>
- [41] <https://github.com/keras-team/keras>
- [42] <https://github.com/Microsoft/CNTK>
- [43] Zhong S, Liu Y, Ren F, et al. Video Saliency Detection via Dynamic Consistent Spatio-Temporal Attention Modelling[C]//AAAI. 2013.
- [44] 齐美彬, 檀胜顺, 王运侠, 等. 基于多特征子空间与核学习的行人再识别[J]. 自动化学报, 2016, 42(2): 299-308
- [45] Wen Y, Zhang K, Li Z, et al. A discriminative feature learning approach for deep face recognition[C]//Proc of European Conf on Computer Vision. Berlin:Springer, 2016: 499-515

- [46] Sun C, Lu H, Yang M H. Learning spatial-aware regressions for visual tracking[J]. arXiv preprint arXiv:1706.07457, 2017.
- [47] Sun Y, Wang X, Tang X. Deep learning face representation from predicting 10,000 classes[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1891-1898.
- [48] <http://www.cbsr.ia.ac.cn/english/CASIA-WebFace-Database.html>
- [49] <http://vintage.winklerbros.net/facescrub.html>
- [50] Sun Y, Wang X, Tang X. Deep convolutional network cascade for facial point detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. 2013: 3476-3483.
- [51] Chen J C, Patel V M, Chellappa R. Unconstrained face verification using deep cnn features[C]//Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on. IEEE, 2016: 1-9.
- [52] Yi D, Lei Z, Liao S, et al. Learning Face Representation from Scratch[J]. Computer Science, 2014.
- [53] Wang D, Otto C, Jain A K. Face Search at Scale: 80 Million Gallery[J]. Computer Science, 2015.
- [54] Hayat M, Khan S H, Bennamoun M. Empowering simple binary classifiers for image set based face recognition[J]. International Journal of Computer Vision, 2017: 1-20
- [55] Nech A, Kemelmacher-Shlizerman I. Level Playing Field for Million Scale Face Recognition[C]// Computer Vision and Pattern Recognition. IEEE, 2017:3406-3415.
- [56] 周晔, 张军平. 基于多尺度深度学习的商品图像检索[J]. 计算机研究与发展, 2017, 54(8): 1824-1832
- [57] Zhu C, Zheng Y, Luu K, et al. CMS-RCNN: contextual multi-scale region-based CNN for unconstrained face detection[M]//Deep Learning for Biometrics. Springer, Cham, 2017: 57-79
- [58] https://github.com/BVLC/caffe/blob/master/examples/mnist/lenet_train_test.prototxt
- [59] Ma J, Wang W, Wang L. Irregular convolutional neural networks[J]. arXiv preprint arXiv:1706.07966, 2017.
- [60] Srivastava R K, Greff K, Schmidhuber J. Highway Networks[J]. Computer Science, 2015.
- [61] Guo Y, Zhang L, Hu Y, et al. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition[C]// Proc of European Conf on Computer Vision. Berlin:Springer, 2016: 87-102

- [62] Zhang K, Zhang Z, Li Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks[J]. IEEE Signal Processing Letters, 2016, 23(10): 1499-1503
- [63] Parkhi O M, Vedaldi A, Zisserman A. Deep face recognition[C]//BMVC. 2015, 1(3): 6
- [64] Taigman Y, Yang M, Ranzato M A, et al. Web-scale training for face identification[C]// Proc of the IEEE conf on Computer Vision and Pattern Recognition. Piscataway,NJ:IEEE,2015: 2746-2754.
- [65] Liu X, Kan M, Wu W, et al. VIPLFaceNet: an open source deep face recognition SDK[J]. Frontiers of Computer Science, 2017, 11(2): 208-218.