单位代码：10293　密　级：公开

南京邮电大学

硕士学位论文

论文题目：基于深度卷积神经网络的面部表情识别研究

学　　号　1019020922

姓　　名　邹翔翔

导　　师　张昀

学 科 专 业　电路与系统

研 究 方 向　智能信息处理

申请学位类别　工学硕士

论文提交日期　二〇二二年五月

Research on Facial Expression Recognition Based on Deep Convolutional Neural Network

Thesis Submitted to Nanjing University of Posts and Telecommunications for the Degree of Master of Science in Engineering

By

Xiang Xiang Zou

Supervisor: Prof. Yun Zhang

May 2022

摘要

面部表情是人们表达情感的重要途径，近年来随着计算机领域的发展，面部表情识别成为了当前的研究热点并取得了显著的进展，可应用于人机交互、情感计算等计算机视觉领域，人工智能和深度学习的发展则更好地促进了面部表情识别的研究。基于机器学习的传统面部表情识别算法采用人工的方式进行特征提取，所提取的面部表情特征存在人为因素的干扰，以至于训练完成的分类器不能有效地解释表情信息，最终导致模型泛化能力不足，识别准确率较低。深度学习算法中的卷积神经网络在面部表情识别任务中的优异性能表现吸引了众多研究学者的目光。但是现实场景中的基于深度学习的面部表情识别仍然受到人体姿势、面部不同程度的遮挡、背景环境与光线干扰等因素的影响，识别的准确率仍需进一步提高，因此本文提出一种改进的基于深度卷积神经网络的面部表情识别模型，用来提高面部表情识别的精度。具体工作和创新点如下：

（1）针对传统面部表情识别模型准确率较低的问题，本文提出了两个改进的卷积神经网络模型用于对比选择。第一个模型是针对面部表情识别任务进行优化改进的 VGG12 网络，VGG12 网络结构基于 VGGNet，改变了其系统结构并添加了 Dropout 避免过拟合；第二个模型是基于深度可分离卷积的深度卷积神经网络（DCNN），DCNN 的结构参考了 VGG12 中卷积块的叠加，但将其核心的卷积层替换成了深度可分离卷积层，同时搭配卷积残差块的使用，使网络能够有效减少参数的情况下，能够提取多尺度上的特征信息，从而有效的保留了细节特征。通过实验发现 DCNN 不仅拥有更少的参数数量，而且准确率也略高于 VGG12，因此，本文选择基于深度可分离卷积的 DCNN 作为基础特征提取网络。

（2）卷积神经网络的输入数据中缺乏多样性和易于分辨的特征信息可能会影响网络的性能，导致面部表情特征提取不足，为了解决以上问题，可以在网络中引入注意力机制使其忽略图片中无关特征而专注于有效特征。因此，在 DCNN 模型的基础上添加了卷积块注意力模块构成了新的 DCNN-CBAM 网络，使网络可以依次沿着通道和空间两个维度对给定的输入特征映射计算注意力图，然后再将输入的特征映射与其注意力图相结合以完成特征的精细化从而提高网络模型的特征表达能力。在不同数据集上的实验结果表明，在 DCNN 网络中引入注意力机制，可以有效地提高网络的性能和识别的准确率。

（3）针对 ReLU 激活函数会在网络训练过程中产生负值神经元坏死，以及传统的 Softmax 损失函数无法解决在面部表情中训练数据存在同类表情差异较大，不同类别的表情差异较小的情况，在 DCNN-CBAM 网络中引入了 Mish 激活函数和 AM-Softmax 损失函数。Mish 激活函数可以避免神经元坏死的情况，而 AM-Softmax 损失函数则可以最大化类间差异，实验

证明，引入 Mish 激活函数和 AM-Softmax 损失函数可以改善 DCNN-CBAM 的训练情况，使其具有更好的特征表达能力。

关键词：面部表情识别，深度学习，深度可分离卷积，注意力机制，Mish 激活函数，AM-Softmax

Abstract

Facial expression is an important way for people to express their feelings. In recent years, with the development of computer realm, facial expression recognition has become a current research hot spot and made remarkable progress, which can be applied to human-computer interaction, emotional computing and other computer vision fields. The development of artificial intelligence and deep learning has better facilitated the study of facial expression recognition. The facial expression features extracted by the traditional facial expression recognition algorithm based on machine learning are interfered by human factors, so that the training model has insufficient generalization ability and low recognition accuracy. The excellent performance of convolution neural network in deep learning algorithm in facial expression recognition task has attracted the attention of many scholars. However, facial expression recognition based on deep learning in real scenes is still affected by several factors such as human posture, facial occlusion, background environment and light interference, thus the accuracy of recognition still needs further improvement. Therefore, this paper proposes an improved facial expression recognition model based on deep convolution neural network for improving the accuracy of facial expression recognition. The specific work and innovations are as follows:

(1) Aiming at the low accuracy of traditional facial expression recognition models, two improved convolution neural network models are proposed in this paper for comparison and selection. The first model is the optimized VGG12 network which is applied to facial expression recognition task. The structure of VGG12 network is based on VGGNet, with its system structure changed and Dropout added to avoid overfitting. The second model is the deep convolution neural network (DCNN) based on depth separable convolution. The structure of DCNN refers to the superposition of convolution blocks in VGG12, but the core convolution layer is replaced by the depth separable convolution layer, together with the use of convolutional residual blocks. In the case of reducing the parameters, the network can extract multi-scale feature information and effectively retain the detailed features. Through experiments, it is found that DCNN not only has fewer parameters, but also has a slightly higher accuracy than VGG12. Therefore, DCNN is chosen as the basic feature extraction network.

(2) The lack of diversity and easily distinguishable feature information in the input data of the convolution neural network may affect the performance of the network and lead to insufficient extraction of facial expression features. In order to solve the problems above, the attention mechanism

can be introduced into the network to make it ignore irrelevant features in the images and focus on effective ones. Therefore, this thesis adds the convolution block attention module to the DCNN model to form the new DCNN-CBAM network. So that the network can calculate the attention map for the given input feature mapping along the channel and space dimensions, and then combine the input feature mapping with its attention map to complete the refinement of features so as to improve the feature expression ability of the network. The experimental results on different datasets show that the introduction of the attention mechanism into the DCNN network can effectively improve the performance of the network.

(3) In view of the problem that the ReLU activation function will cause negative neuron necrosis in the process of network training, and the traditional Softmax loss function can not solve the situation that there are large differences between similar expressions and small differences between different types of expressions in the training data of facial expressions, Mish activation function and AM-Softmax loss function are introduced into DCNN-CBAM network. The Mish activation function can avoid neuronal necrosis, while the AM-Softmax loss function can maximize the difference between classes. The experiment proves that the introduction of Mish activation function and AM-Softmax loss function can improve the training of DCNN-CBAM and make it have better feature expression ability.

Key words: Facial expression recognition, Deep learning, Depthwise separable convolution, Attention mechanism, Mish activation function, AM-Softmax

目录

第一章 绪论

1.1 研究背景及意义

近年来，随着计算机技术和人工智能的飞速发展，计算机视觉成为研究热点。计算机视觉相关任务的研究目的是利用图像处理方法，让机器能够实现符合人类期望的机器视觉功能。目前，计算机视觉涉及众多领域，例如目标检测[1]、语义分割[2]、人脸识别[3]、以及姿态识别[4]等。其中面部表情识别因其广阔和极具商业价值的应用前景成为当前研究的热点。

一般来说，人们通过面部表情[5]和声调[6]来推断他人的情绪状态，如喜悦、悲伤和愤怒[7]。在众多的非语言成分中，面部表情承载着情感的意义，是人际交流的主要信息渠道之一。心理学家 Mehrabian[8]通过大量的实验研究指出人们在日常沟通交流中，仅有 7%的情感信息交流来自于语言表达，语言表达中伴随着的语音语调则占据了 38%，而约为 55%的情感交流信息是通过人们的面部表情进行传达的。

面部表情识别技术主要就是通过对静止图像或动态视频序列中的面部表情特征进行提取，然后根据提取的表情特征进行分析后使用分类器进行面部表情的分类识别。随着计算机技术和人工智能的发展，计算机实现面部表情识别成为了现实[9]，人机交互也更好地促进了面部表情识别的研究[10]。将面部表情识别应用于我们日常生活中，通过识别人们的面部表情，分析感情状态，并通过利用这些感情信息可以帮助人们提升生活质量和幸福感，一些具体应用如下：

(1)汽车安全驾驶。安全驾驶一直受到社会广泛的关注，汽车的行驶安全不仅仅是依靠汽车的安全性能，驾驶员的情绪状态也是影响汽车安全驾驶的重要因素之一[11]。驾驶员需要在驾驶汽车时保持平稳的情绪，否则当驾驶员产生强烈的情绪波动容易引发产生一些非理性的驾驶行为最终可能会导致交通事故的发生。将面部表情识别应用到安全驾驶领域，通过监测驾驶员的情绪状态，在驾驶员情绪波动时发出警告，提醒驾驶员安全驾驶，从而可以有效地避免交通事故的发生。伴随着新能源汽车的飞速发展，汽车都配备了智能系统，这让实现面部表情识别的驾车辅助系统成为了现实。

（2）线上教学监测。由于疫情的影响，国内很多地区的学校都采取了网上授课的形式，但是线上教学有其局限性，老师在授课时无法关注到每个学生的情绪状态并及时收到教学反馈，学生也通过视频上课也无法保持住专注度，从而影响授课质量。而通过面部表情识别则可以实时监测每个学生的情绪状态，并由此判断学生的上课专注度，从而进行提醒。将面部

表情识别系统应用于线上教学，从而可以使老师提高课堂质量，学生保持学习专注度。

（3）短视频推送。伴随着智能手机的不断更迭，而如今短视频又十分盛行，手机短视频已经成为了人们生活的一部分，但是在短视频平台上仅凭点赞与否无法准确判断用户的喜好，大数据就无法准确推送用户感兴趣的内容，而通过检测用户在观看视频时的情绪变化就可以准确的判断用户对此类视频的喜恶，从而可以精准推送相关内容，提高用户体验的同时也为短视频平台带来收益。

（4）智能机器人。根据资料显示，20 世纪 90 年代以来，中国的老龄化[12]进程不断加快，而在 21 世纪前期中国人口老龄化将会达到顶峰，这些老人不可避免地会缺少关心和陪伴。而如果通过将面部表情识别技术与智能机器人相结合，通过实时监测老人的情绪变化，及时判断他们的心理和身体健康状态，然后智能采取措施解决问题，让智能机器人具备人的共情能力可以更好地陪护老人。

此外，面部表情识别还可以应用到许多其他领域服务和改善人们的生活，例如医疗辅助[13]、刑侦安防[14]、游戏娱乐等等。由此可见面部表情识别具有研究的价值和意义。

1.2 国内外研究现状

从 18 世纪起，许多学者就开始了关于面部表情识别的研究，其中最有名的当属英国的生物学家达尔文，在他撰写的《人类和动物的表情》[15]一书中通过对表情原理的阐述，来介绍人类和动物的表情特性以及不同的表情所蕴含的深层含义。在书中的研究证明了种族、地域甚至性别都会对人的表情产生影响。达尔文的这本著作使西方心理学的主流思想摆脱了哲学的范畴而进化成为了生物科学。进入到 19 世纪，美国心理学家 Ekman 等人[16]通过人类不同表情的差异性将面部表情进行了分类，具体可以分为开心、伤心、生气、厌恶、恐惧、惊讶和中性这七类表情，这也是人类的七种基本面部表情。随后 Ekman 又研究了人的不同表情与面部肌肉运动状态的联系，总结并提出了面部行为编码系统（Facial Action Coding System, FACS）[17]。FACS 描述了不同面部动作单元之间的联系与区别以及不同的面部表情中动作单元的组成成分，这也预示着面部表情识别从之前的生物科学的理论观察研究进入到了计算机技术分析。随着 Mase[18]在 1991 年提出使用光流的信息来推测面部动作单元，从而实现表情识别，至此面部表情识别正式进入了新的实践阶段。一般的面部表情识别系统包含三个步骤，分别是人脸检测、特征提取和面部表情分类，如图 1.1 所示，下面我们将从基于传统的机器学习和基于深度学习这两方面介绍面部表情识别的最新研究进展。

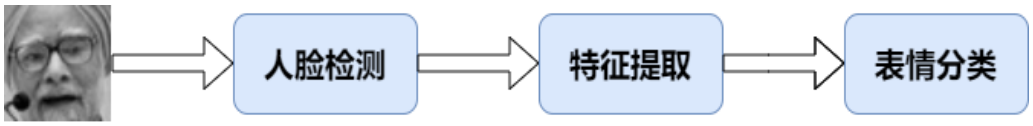


图 1.1 一般的面部表情识别过程

1.2.1 基于传统机器学习的面部表情识别

在传统机器学习的面部表情识别研究中，特征提取主要是利用计算机对数字化的面部表情数据进行计算和处理，然后再提取特征的数学方法，其应用到的特征提取算法因为提取了面部图像的部分特征，所以在一定程度上也起到了降低特征维度的作用，基于机器学习的面部表情特征提取一般分为两种方法：整体提取和局部提取。

整体提取法的核心思想就是将人的面部表情整体化进行特征提取。当人产生情绪变化时，人的面部器官会发生较为明显的形变，会对面部的全局信息产生影响，就可以从全局角度提取表情特征。整体法中的一些经典的基础算法有线性判别法（Linear Discriminant Analysis, LDA）[19]，独立分量分析法（Independent Component Analysis，ICA）[20]和主成分分析法（Principal Component Analysis，PCA）[21]，针对这些传统算法研究者们也进行了改进。文献[22]提出了一种新的移位γ加速线性判别（Shifted Delta Acceleration Linear Discriminant Analysis, SDA-LDA）算法来提取视频序列中最具辨别力、动态和鲁棒性的表情特征。文献[23]将 PCA 特征提取算法和粒子群优化算法（Particle Swarm Optimization，PSO）进行了结合，主成分分析法用于为每个分类的表情提取有效的特征向量，粒子群优化算法对 PCA 提取的特征进行优化，最终的方法在 JAFFE 数据集上实现了 94.97%的分类准确率。

面部表情的变化不仅仅体现在整体上，也存在局部的差异，这就需要局部提取特征。将人的面部分为几个部分，每一个部分的重要性也不尽相同，使用局部法提取重要性相对较高的部分的特征，对重要性较少的部位进行的特征分析就减弱些。在传统的机器学习中主要使用的局部特征提取方法有 Gabor 小波法[24]和局部二值模式（Local Binary Pattern, LBP）[25]。文献[26]通过将拉东（Radon）变换和 Gabor 小波变换结合用于提取面部表情的可变特征从而提高识别精度。王等人[27]将局部二值模式改进为双局部二值模式（Double local binary pattern, DLBP）再与泰勒展开式（Taylor expansiion, TE）融合提出了基于改进 LBP 的人脸面部表情特征提取方法，在 JAFFE 面部表情数据集上的实验验证了这种新算法的有效性。

在传统的机器学习中，在特征提取的工作之后还需要单独设计分类器才可以判断提取到的表情特征所归属的表情类别。一般使用的分类器有基于贝叶斯网络（Bayesian network）和基于支持向量机（Support Vector Machines, SVM）。

基于贝叶斯网络的分类器是以贝叶斯公式为基础，从已知的表情分类信息推测出未知表情。基于贝叶斯网络的分类方法其中又主要分为各种贝叶斯网络分类算法的改进[28]和隐马尔可夫模型（Hidden Markov Model, HMM）算法。文献[29]提出基于 Gabor 滤波器和改进的隐马尔可夫模型进行特征提取并分类的新框架，使用 Gabor 滤波器提取局部面部表情特征，并通过独立分量分析（ICA）对提取到的局部特征进行降维操作，最后将降维后的特征输入到两层 HMM 中进行分类，通过实验验证了框架的高效和健壮性。

基于支持向量机的分类器的原理是通过不断地优化目标函数，直至不同类别样本的特征之间差距最大。文献[30]研究了在 SVM 中使用线性核、二次核和三次核这三种不同的核函数对面部表情识别性能的影响。文献[31]中将基于支持向量机的分类方法与遗传算法（Genetic Algorithm, GA）相结合，提出了一种使用两个新定义的表情几何特征：标记曲率和矢量化标记的面部情感识别技术。

这些传统的基于机器学习的面部表情识别方法为面部表情识别的研究做出了重要贡献。然而，传统的人工特征提取方法使用会存在人为因素的干扰，以至于训练完成的分类器不能有效地解释表情信息，最终导致模型泛化能力不足从而无法实现较高的识别准确率。而基于深度学习的面部表情识别方法使用深度卷积神经网络可以自动提取形成更抽象的高层特征或属性特征[32]，从而提高最终的分类或预测精度，下面一节对基于深度学习的面部表情识别进行详细介绍。

1.2.2 基于深度学习的面部表情识别

深度学习（Deep Learning, DL）这一名词首先是由加拿大的 Hinton 提出的，他也因此被称为神经网络之父。Hinton 和 Ruslan 在 2006 年的一篇论文[33]中解决了如何消除深层网络训练中梯度消失的问题，至此深度学习的浪潮便开始兴起。而在 2012 年，Hinton 的研究小组为了证明深度学习与传统的机器学习相比在解决图像分类与识别问题上的优越性，在 ImageNet 图像识别比赛中提出并构建了名为 AlexNet[34]的卷积神经网络（Convolutional Neural Networks, CNN）。AlexNet 不仅夺得了比赛冠军，并且其分类准确率远超过第二名基于传统机器学习的 SVM 方法，这个网络首次使用了电脑 GPU 用来加快模型的计算速度，且在模型中使用了 ReLu 激活函数，在解决网络梯度消失问题同时也加快了模型收敛速度。自 2012 年开始，基于深度学习的不同网络结构层出不穷，促使其在其他领域也大放异彩。2016 年，一场非比寻常的围棋比赛吸引了全世界的目光，对战的双方分别是由谷歌公司基于深度学习设计并开发的智能 AI 软件 AlohaGo 和围棋世界冠军李世石，这场人机大战最终以李世石一比四惨败于 AlohaGo

结束，次年 5 月，它又零封了中国棋手柯洁。至此，基于深度学习的研究在医疗、金融、无人驾驶等领域迅速发展。

Hinton 和他的团队将深度学习定义为任何可以训练一个系统并具有两层非线性隐藏层以上的学习方法[35]。目前图像分类与深度学习的关系密不可分，而面部表情识别又是图像分类的子任务，将基于深度学习的算法与面部表情识别结合，相较于传统的基于机器学习的表情识别方法有许多益处。首先，基于机器学习的面部表情识别算法中特征提取与特征分类是两个独立的步骤，需要分别进行独立研究，而在深度学习中特征提取与特征分类是在同一个算法中进行设计与优化的，可以简化算法，降低算法的复杂度；其次，传统的机器学习方法中进行特征提取依赖于手动提取特征，这种手动方式不仅繁琐而且提取的特征也容易受人为因素的干扰，而在深度学习中是由对图像具有较好提取特征能力的神经网络自动提取特征，这使得基于深度学习的面部表情识别方法具有更好的特征表达能力。基于深度学习的面部表情识别方法中最常见的网络是卷积神经网络[36],许多国内外研究学者都对其进行了深入的研究。

文献[37]针对大多数现有面部表情识别方法中为了训练的简便只考虑正面图像而忽略侧面图的问题，提出了一种通过迁移学习（Transfer Learning，TL）技术的深度卷积神经网络（Deep Convolutional Neural Networks, DCNN），其中预训练的 DCNN 模型的密集层需要与面部表情识别兼容，并且使用面部表情数据集对模型进行微调。所提出的面部表情识别系统使用了八种不同的预训练 DCNN 模型（VGG-16、VGG-19[38]]、ResNet-18、ResNet-34、ResNet-50、ResNet-152[39]、Inception-v3[40]和 DenseNet-161[41]）在 KDEF 和 JAFFE 数据集上进行实验验证，最终预训练模型的在两个数据集上都取得了较高的准确率，在具有不同的侧面表情图像和正面表情图像的多样性的 KDEF 数据集上获得如此良好的性能对面部表情识别系统的开发更有意义。Jain[42]等人提出一种包含卷积层和残差块的深度神经网络模型，通过在两个公开数据集上对该模型进行的测试证明残差块的使用可以有效地改善模型性能。Saurav 等人[43]提出一种情感网络（Emotion NetworkEmnet, EmNet）的架构来实时自动识别自然环境中的面部表情，EmNet 架构由两个结构上类似的深度卷积神经网络（DCNN）模型使用联合优化方法构成，EmNet 会对输入的面部表情图像给出三个预测，并使用加权最大融合和平均融合对这三种预测结果进行融合从而得到最终的分类结果。此外，为了能够将面部表情识别系统部署在资源受限的嵌入式平台，使用 TensorRT[44]软件开发工具包对 EmNet 模型和人脸检测器进行了优化处理，最终该网络在 RAF-DB 数据集[45]上获得了 87.16%的识别率，与当前最先进的技术在准确性上相比有着显著提高。虽然这些卷积神经网络在面部表情识别任务中都能达到较高的识别准确率，但是却忽略了网络的过多参数会减缓网络的收敛速度，为此本文将重点放在了轻量化的深度卷积神经网络的研究中。

文献[46]使用深度可分离卷积来构建轻量级深度神经网络并将提出一系列网络架构称为：MobileNets，通过在对象检测、细粒度分类以及人脸检测等任务的实验证明了 MobileNets 的优越性。Sadik 等人[47]使用迁移学习将 MobileNet 模型应用于面部表情分析之中，实验结果表明，采用迁移学习方法的 MobileNet 模型可以在识别任务中提供令人满意的结果。为此本文决定通过深度可分离卷积对传统的卷积神经网络进行改进，从而达到模型的轻量化。

Abhinav 等人[48]提出了两种新颖的卷积神经网络架构，并研究了 CNN 中的卷积核的大小和过滤器的数量对分类精度的影响，通过在 Fer2013 数据集上的实验验证了这两个指标对网络的准确性存在影响，在搭建 CNN 过程中可以通过调整这些网络参数以提高网络的性能表现。这对我们的深度卷积神经网络中卷积核和过滤器的选择具有参考意义。

为了更好的发挥深度学习在面部表情识别这项任务上的优势，在不改变网络模型本身架构的前提下，很多研究学者对卷积神经网络进行了改进以提高网络的特征表达能力，其中较为合适的方法就是引入注意力机制或者使用改进的激活函数或损失函数。

在图像分类中我们需要关注输入的特征图与上下文相关的特征，即注意力机制。注意力机制[49]可以使神经网络忽略无关信息而专注于有效信息。文献[50]中通过注意力机制提出了空间变换器模块，将图像的空间域信息变换到另一个相对应的空间，从而提取图像中的感兴趣区域。文献[51]中提出了挤压-激励(Squeeze-and-Excitation, SE)块，SE 块在训练过程中会对输入数据中的通道特征进行处理。但 SE 块仅仅考虑通道注意力机制在网络训练过程中的作用，忽略了图像数据中的空间域信息，空间域转换网络也存在着相似的问题，它忽视了通道注意力的影响。为此文献[52]提出了卷积块注意力模块(Convolutional Block Attention Module, CBAM)，与空间变换器模块和 SE 块不同，它是一种将空间注意力和通道注意力进行有效结合的注意力机制模块。文献[53]通过将卷积块注意力模块(CBAM)与 VGGNet 相结合用以解决网络特征提取能力不足的问题，最终通过在 CK+和 FER-2013 数据集上的实验验证了该方法的可行性与有效性。为此本文决定将注意力机制与我们的轻量化模型相结合，以此提升模型的识别准确率。

此外，对于卷积神经网络模型，激活函数和损失函数是其核心，激活函数可以激活神经元的特征来解决非线性问题，而损失函数可以用来表现预测与实际数据的差距程度。文献[54]研究了激活函数在 CNN 模型中的影响。根据 CNN 模型中激活函数的设计原则，提出了一种新的分段激活函数应用于面部表情识别任务中。文献[55]将一种新的附加角度引入到 Softmax 损失函数中，提出了一个概念上简单且几何上可解释的新的损失函数，并将其应用于人脸检测之中。考虑激活函数和损失函数能进一步提升卷积神经网络的性能，因此，本文接着探索研究了如何将激活函数和损失函数同深度卷积神经网络进行结合，从而提高模型特征表达能

力和识别精度。

1.3 论文主要工作和章节安排

本文的研究工作获得了国家自然科学基金“基于深度学习的移位 MIMO‘鬼’成像方法研究”（项目批准号：61871234）的支持。本文主要工作是基于深度卷积神经网络对面部表情进行识别。本文的主要工作和创新点如下：

（1）针对面部表情识别任务提出两个改进的卷积神经网络模型。第一个模型是基于 VGGNet 的改进的经典卷积神经网络 VGG12，VGG12 在 VGGNet 的基础上减少了卷积层的层数和卷积层和全连接层中的滤波器数量，并添加了 Dropout 防止网络过拟合；第二个模型是基于深度可分离卷积和卷积残差块的 DCNN 网络，与经典的卷积层相比，深度可分离卷积层拥有更少的卷积参数量和更低的运算成本，卷积残差块的使用是为了防止网络出现梯度爆炸或梯度消失的情况。

（2）针对卷积神经网络的输入中缺乏多样性和类别可辨别信息可能会影响网络的性能，而注意力机制可以使神经网络忽略图片中无关特征而专注于有效信息。研究了不同注意力机制的性质和优缺点，在 DCNN 中引入了一种结合了空间注意力和通道注意力的混合注意力机制：卷积块注意力模块（CBAM），CBAM 会按序对输入特征图的通道注意力图和空间注意力图进行计算处理，从而有效地提高网络的特征表达能力和识别准确率。

（3）针对不同的激活函数和损失函数会对网络的训练和模型的特征表达能力产生影响，研究了不同激活函数和损失函数的性质和优缺点，将 Mish 激活函数和 AM-Softmax 损失函数引入到 DCNN-CBAM 模型中。与 ReLU 激活函数相比，Mish 激活函数可以增强正则化效果以及有效地避免神经元坏死的情况；AM-Softmax 损失函数则可以最大化不同类别之间的差异，从而显著提高模型的特征表达能力。

本文的研究点及主要工作如图 1.2 所示。

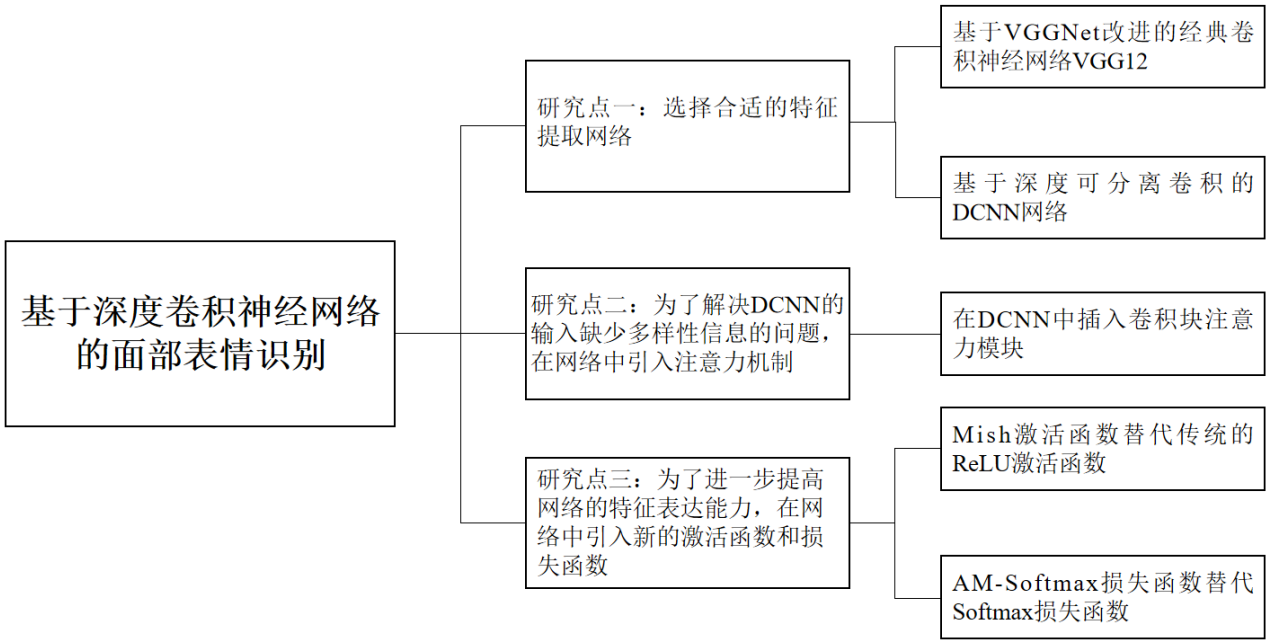


图 1.2 本文的研究点及主要工作

本文分为五个章节。主要的章节安排为：

第一章绪论，主要介绍了面部表情识别的研究背景及意义，同时对面部表情识别的国内外研究现状进行了阐述。

第二章叙述了本课题的基础理论知识，首先介绍了卷积神经网络的基本组成结构以及其原理知识，然后具体阐述了目前应用广泛的基于卷积神经网络的面部识别的流程。

第三章是面部表情识别的特征提取网络的选择，首先介绍了改进的经典卷积神经网络 VGG12 的结构，然后介绍了基于深度可分离卷积的 DCNN 网络的结构。通过仿真实验对两个模型进行性能对比，最终选择了 DCNN 作为面部表情识别系统的特征提取网络。

第四章首先对目前图像分类中不同的注意力机制进行原理介绍并对其进行分析对比，然后将结合了空间注意力和通道注意力的卷积块注意力模块引入到 DCNN 模型中，接着介绍了网络训练情况，最后通过实验验证了注意力机制可以有效地提高模型性能。

第五章以上文所提出的算法为基础，首先阐述了目前在深度学习中较为常用的激活函数和损失函数及其变体，并通过数学公式对其进行原理分析，然后提出了将 Mish 激活函数和 AM-Softmax 损失函数引入到模型之中。然后用实验定量且定性的验证了该算法能够显著地提升面部 表情识别的准确度。最后将模型应用于面部表情识别系统中。

第六章是对本文的总结以及对未来工作的展望。通过对全文的工作内容进行总结，由此引出对本课题研究点在未来可能进行拓展的方向的探讨。

第二章 基础理论介绍

2.1 卷积神经网络

卷积神经网络是深度学习的代表算法之一，它拥有丰富的数据表征学习能力。与传统算法不同的是，CNN 可以通过参数的自动调整从而学习到最有用的特征。并且，传统算法需要先对数据进行预处理，后续才能进行特征的提取和定位，而 CNN 则直接将图像的原始信息作为输入，从而缩短了算法的时间。如图 2.1 所示为 CNN 的结构图，为了方便后续步骤的运算，通常会在输入层对数据预处理工作，然后经过卷积层和激励层进行特征提取，池化层再负责对提取到的特征进行数据压缩，最后通过全连接层实现分类。下面我们将具体介绍卷积神经网络。

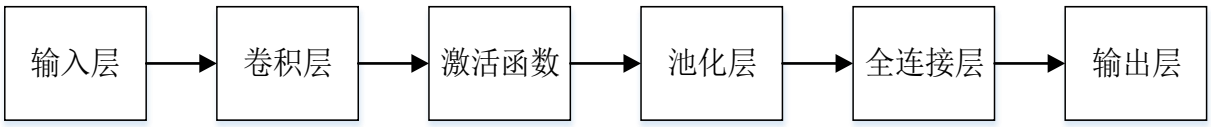


图 2.1 CNN 结构图

2.1.1 卷积层与全连接层的对比

在全连接神经网络（Fully Connected Neural Network, FC）中假设第n层的神经元数量为n^n，第$n-1$层的神经元数量为n^{n-1}，那么将第$n-1$层与第n层相连接需要$n^{n-1} \times n^n$个连接边，即权重矩阵所含有的参数数量为$n^{n-1} \times n^n$。当神经元数量巨大时，权重矩阵的参数量就会很多，导致神经网络的训练效率降低。

而在卷积层中，第n层的输入z^n为第$n-1$层的活性值a^{n-1}与滤波器的卷积，如式（2.1）。

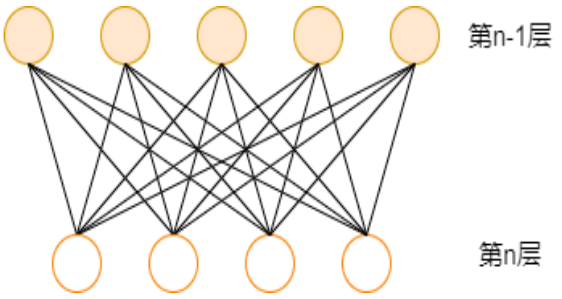
$$z^n = w^n \otimes a^{n-1} + b^n \tag{2.1}$$

其中滤波器w^n为权重向量，b^n为偏置量。

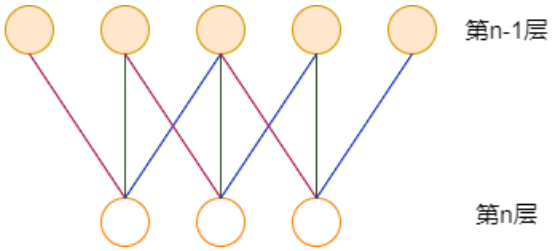
与全连接神经网络相比，卷积神经网络中有两个很重要的性质，它们分别是局部连接和权重共享，图 2.2 展示了全连接层与卷积层的对比。

由于局部连接的存在，卷积层（我们假设为第$n-1$层）中的每一个神经元都只和下一层（第n层）的某些特定的神经元相连，这些特定的神经元由局部窗口所确定。与全连接层相比，卷积层与下一层的连接数远远减少，由原来的$n^{n-1} \times n^n$的连接数减少为$n^{n-1} \times m$，卷积层中的滤波器的大小表示为m。

由于权重共享的存在，滤波器可以共同作用于所有的神经元。如图 2.2b 所示，相同颜色的连接的权重都是一致的。而在权重共享中，一个滤波器仅能提取一种局部特征，因此，为了能够提取多种特征，往往需要在卷积层中使用多个滤波器。



(a) 全连接层



(b) 卷积层

图 2.2 全连接层与卷积层的对比

在局部连接和权重共享的共同作用下，卷积层只有m维的权重w^n和一维的偏置量b^n，共计$m+1$个参数。参数的数量与神经元个数不存在关系。此外第n层的神经元数量n^n与上一层的神经元数量n^{n-1}有关，它们满足$n^n = n^{n-1} - m + 1$。

2.1.2 卷积层

卷积层（convolutional layer）是卷积神经网络中的核心基础结构。卷积的作用就是提取输入数据的一个局部特征，卷积层中的每一个卷积核都是一个特征提取器，因此可以充分提取特征。目前主要是将卷积神经网络应用于图像处理之中，而图像在数据处理中属于二维数据，因此为了充分提取图像的数据特征，需要将卷积层重构成三维的结构，其尺寸为$M \times N \times D$，其中M为宽度，N为宽度，D为深度，图 2.3 给出了卷积层的三维结构表示，如图 2.3，D个$M \times N$的特征映射共同组成了卷积层的基本结构。特征映射（Feature Map）是图像经过积操作后提取到的特征，每个单独的特征映射都代表了图像的一种特征。因此需要在每一个卷积层使用多个特征映射，从而提高网络的特征表示能力。

卷积（Convolution）是数学中一种重要的运算方式，卷积操作的本质是一个滤波过程，在图像和信号处理中被普遍使用，主要的应用形式有一维卷积和二维卷积。

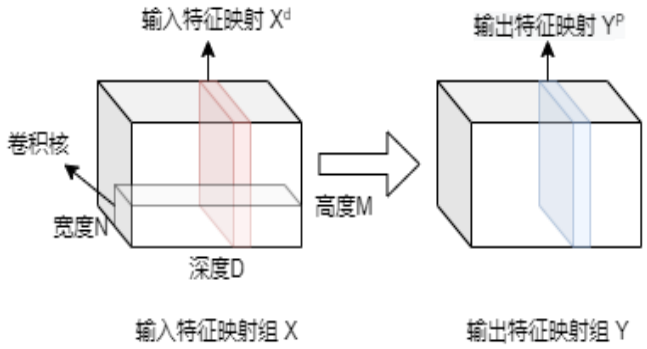


图 2.3 卷积层的三维结构

一维卷积最初是在信号处理中使用，通常用于计算各种信号的延迟累计。我们假设一个一维卷积层的输入信号序列x为$x_1, x_2, \cdots, x_t$，卷积层的卷积核或滤波器ω为$\omega_1, \omega_2, \cdots, \omega_m$，那么滤波器和输入信号序列的卷积如式（2.2）。

$$y_t = \sum_{k=1}^{m} \omega_k \cdot x_{t-k+1} \tag{2.2}$$

信号序列x和滤波器的ω的一维卷积定义为式（2.3）。

$$y = \omega * x \tag{2.3}$$

其中$*$为卷积运算符。

图 5.1 给出了一维卷积示例，其中一维卷积层的滤波器为$[-1, 0, 1]$，如图，不同颜色的连接边上的数字代表了滤波器中不同的权重。

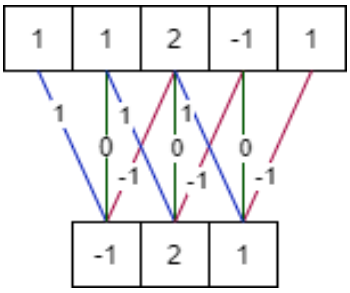


图 2.4 一维卷积示例

在图像处理中，二维卷积比一维卷积更为常用，因为图像数据本身就是二维结构。二维卷积是一维卷积的拓展，假设输入图像数据为$X \in \mathbb{R}^{M \times N}$，卷积层的滤波器为$W \in \mathbb{R}^{m \times n}$，$m$和$n$远大于$M$和$N$，其卷积为式（2.4）。

$$y_{i,j} = \sum_{u=1}^{m} \sum_{v=1}^{n} \omega_{uv} \cdot x_{i-u+1, j-v+1} \tag{2.4}$$

则输入图像X和滤波器W的二维卷积定义为式（2.5）。

$$Y = W * X \tag{2.5}$$

其中$*$为卷积运算符。

二维卷积示例如图 2.4，输入是一个4×4二维图像，使用一个卷积核为3×3的卷积对其进行卷积操作得到一个2×2的特征映射。具体操作为：在对输入图片数据的特征平面上，

卷积核使用步长为 1 在特征平面上依次按照从左向右、从上往下的顺序滑动，再进行加权求和得到最终的特征映射。

在卷积的标准定义上，为了增加卷积的多样性和特征提取的灵活性，还引入滤波器或者卷积核的滑动步长和零填充。

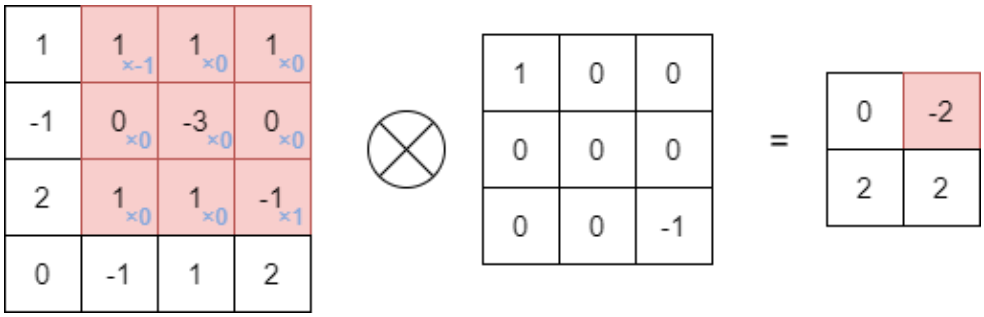
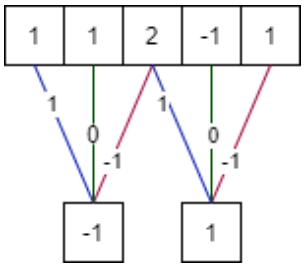


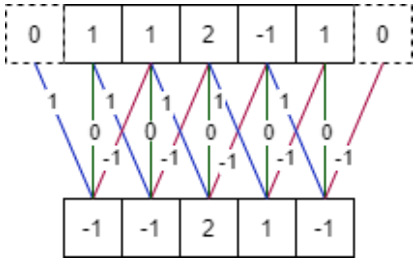
图 2.5 二维卷积示例

卷积层中的卷积核的步长（Stride）是指卷积核在进行特征提取时的滑动的间隔，在卷积操作中，默认步长为 1，上文中的一维和二维卷积的示例中卷积核的步长皆为 1。步长为 2 的一维卷积示例如图 2.5a。

零填充（Zero Padding）是指对输入数据的边缘进行补零操作，零填充可以有效地防止像素丢失的情况。将一维卷积进行两端补零操作后示例如图 2.5b，图中p是指补零的数量。



(a) 步长为 2 的一维卷积示例



(b) 零填充$p = 1$的一维卷积示例

图 2.6 卷积的步长和零填充

我们假设某一卷积层的从上层传入神经元的数目为n，卷积核的大小为m，步长为s，进行零填充的数量为p，那么该卷积层的神经元数量sum为式（2.6）。

$$sum = \frac{n-m+2p}{s} + 1 \tag{2.6}$$

2.1.3 激励层

卷积层的作用是对输入进行线性处理，具有局限性，将卷积层进行简单的叠加无法解决非线性问题时。在网络中添加激励层就可以解决非线性问题，激励层可以增强神经网络的学习能力和表示能力，从而提升网络对数据的特征表达能力。激活函数是激励层的核心，激活函数需要具有以下几点特性：

（1）激活函数的特性要求它是一种连续并可导的非线性函数，但是并不需要严格可导。由于激活函数在定义域上可微，所以能够使用数值优化的原理来对网络参数进行学习。

（2）激活函数以及其导函数要避免复杂，简单的激活函数可以提高网络的运算效率；

（3）激活函数的导数的值域也有限制，需要将其控制在一个合适的范围之内，过大或过小的导数值都会对训练的效率和稳定性产生不良的影响。

关于激活函数的详细内容我们将在第五章进行展开叙述。

2.1.4 池化层

池化层（Pooling Layer）的别名也称为子采样层（Subsampling Layer），具体作用是对特征映射进行筛选压缩，从而可以通过减少特征的数量达到减少参数量的效果。上文中提到卷积层虽然可以削减神经元之间的连接数，但是存在于特征映射之中的神经元数量却没有改变，倘若在卷积层后直接连接一个分类层，会导致较高的分类层输入维度，网络易产生过拟合的现象，而如果将池化层加在卷积层后就解决这一问题，池化层会将卷积层的输出的特征映射进行降维，从而有效地避免了过拟合。

池化（Pooling）操作就是将输入至池化层的特征映射进行区域划分，划分的区域可以重叠，但是为了计算的简便最好是划分成不重叠的区域，然后对每个区域进行下采样（Down Samping），最终得到一个值作为这个区域的代表值。目前较为常用的池化操作一般是平均池化（Mean Pooling）和最大池化（Max Pooling）。平均池化的输出是将输入特征映射中某个区域内所有值加和取平均值，而最大池化的输出是选取某个区域的最大值。

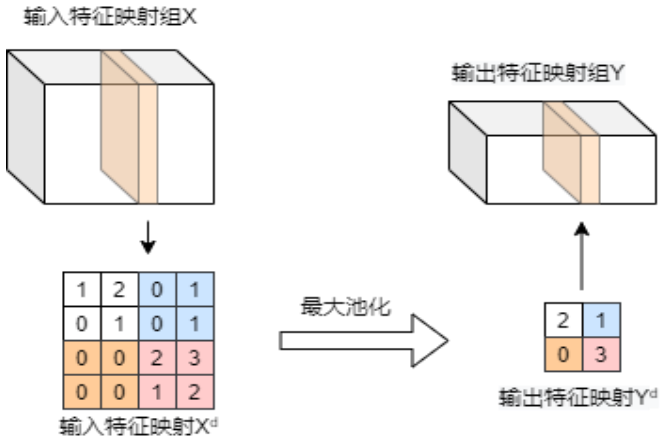


图 2.7 最大池化操作过程示例

图 2.7 给处理最大池化操作过程示例，由图可知，池化层不仅可以有效地对特征映射进行降维，而且也可以使得神经网络对一些局部特征保持不变并使其拥有更大的感受野。当然池化层也可以看作卷积核为取最大值或取平均值函数。卷积核大小围殴$m \times m$，步长为$s \times s$的特殊的卷积层。一般的池化层对输入特征映射的采样区域为2×2大小，过大的采样区域虽然可以使神经元的数目成倍缩小，但是也会造成特征信息的过分丢失。

2.1.5 全连接层

全连接层（Fully Connected Layers, FC）一般也用作分类层，它把前面经过卷积层、池化层和激活函数得到的特征信息进行整合并在样本空间进行映射。经过全连接层输入图像后得到的在样本空间的分数范围是从$-\infty$到$+\infty$的，需要通过分类器将进行归一化才能实现最终的分类。

现如今，在卷积神经网络中常用的分类器有 Logistic 回归（Logistic Regression，LR）和 Softmax[56]。

Logistic 回归是用于解决二分类问题的线性模型，将多个 Logistic 回归进行组合也可解决多分类任务。逻辑回归在神经网络中主要是通过 Sigmoid 函数实现的，既可以用于概率预测，也可用于各种分类问题。虽然 Sigmoid 函数计算成本较低且易于实现，但是它很容易产生过拟合的现象，无法进行准确的分类。

Softmax 是如今卷积神经网络中最基础也是最常用的分类器，比逻辑回归相比，使用单个 Softmax 分类器就可以直接解决多分类问题。对于多分类问题，对于一个输入样本x，Softmax 预测其属于类别c的条件概率计算公式为式（2.7）。

$$p(y = c \mid x) = \frac{e^{\omega_c^T x}}{\sum_{j=1}^{C} e^{\omega_j^T x}} \tag{2.7}$$

其中 C 表示总类别数，ω_c代表第c类的权重向量，p 代表了该类别属于类别c的条件概率。经过 Softmax 得到的条件概率的值在 0 至 1 之间，且所有类别的概率值相加的和应为 1。

2.2 面部表情识别

面部表情识别的流程在第一章已经简要介绍过了，下面我们对基于卷积神经网络的面部表情识别流程进行详细的介绍，基于卷积神经网络的面部表情识别流程如图 2.8，一共包含了四个部分，它们分别是人脸检测、图像预处理、特征提取和表情分类。人脸检测是为了确定并标出图像或视频中的人脸，图像预处理是为了对面部表情数据进行数据增强或归一化处理，最后将处理后的面部表情数据送入卷积神经网络中进行特征提取与分类。

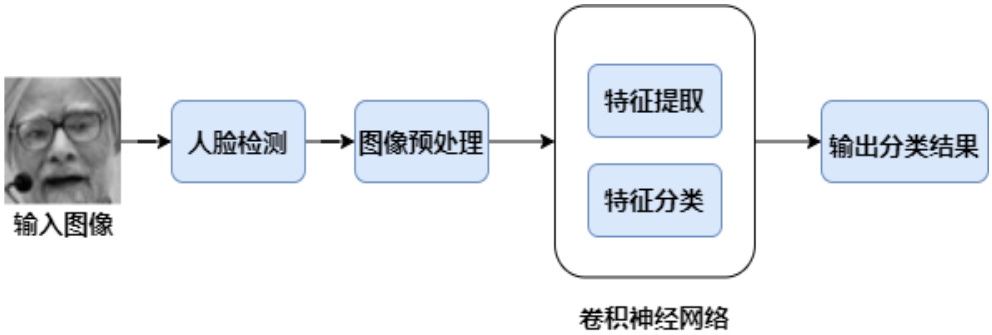


图 2.8 基于卷积神经网络的面部表情识别流程

2.2.1 人脸检测及面部跟踪

进行人脸检测的目的是确定输入图像中有无人脸并对人脸进行面部跟踪，如果存在人脸便框出人脸并实时进行面部追踪，人脸检测的性能直接影响着面部表情识别的准确性。

人脸检测在早期是通过基于模板匹配的方法实现的，首先确定以一个人脸作为模板，然后确定一个模板大小的窗口，将此窗口在待检测图像上进行滑动判断人脸的存在。但是由于不同图像尺寸不一致，且其中的人脸的占比又有差异，所以还需对其进行缩放处理，然后再滑动窗口进行人脸检测。这种算法的实时性很差而且无法识别有遮挡或有一定旋转角度的人脸。但是随着机器学习的诞生，人脸检测与机器学习相融合，诞生了基于多层感知机的模板匹配算法。其中以 Rowley 等人[57]提出的基于神经网络的二分类多层感知机模型最有代表性，该方法使用人脸和非人脸图像数据进行训练。但是由于复杂的模型结构设计和滑动窗口的运算密集，导致人脸检测速度还是不快。

目前使用最多的也是最基础的人脸检测及面部跟踪算法是基于 AdaBoost 框架的方法，其

首次出现于 2001 年，由 Viola 和 Jones 联合提出[58]，其思想是使用 Haar-like 特征弱分类器和多个级联的 AdaBoost 分类器构建成一个人脸检测器，此后此算法也被称为 Viola-Jones 算法。使用多个弱分类器进行人脸检测的优势在于可以迅速地排除非人脸的窗口，同时当某个窗口通过分类器的判定检测为人脸时，则会该窗口作为人脸框进行输出并实时的对面部进行追踪，这种操作可以在保证检测精度的同时大大提高检测的速度。基于深度学习的方法与基于 AdaBoost 框架的方法类似，也是使用了多个分类器级联，不同的是其中的分类器使用的是神经网络训练得到的模型。基于深度学习的人脸检测算法相比于基于 AdaBoost 框架在识别精度上会有所提升，但是实时性却有所下降，所以具体使用何种人脸检测及面部跟踪算法还需根据具体使用场景再进行选择判断。

2.2.2 图像预处理

目前现有的面部表情数据集的规模较小，而卷积神经网络的训练往往需要大量的数据集，否则会因为数据的缺失造成网络的过拟合，影响网络的识别准确率。为了能够利用有限的数据集，需要对数据集进行图像预处理工作。在图像处理中通常采用的预处理方式有两种，一种是几何变换方法，具体操作是对图片进行水平或垂直翻转或随机旋转一定的角度达到数据增强的目的；另一种是像素变换方法，具体操作对图片添加噪声或者对图片的亮度进行调整模拟光照变化。一般数据集在训练之前还会进行归一化处理，具体操作是将输入的数据转化为 0 至 1 之间或-1 至 1 之间，归一化操作可以加快梯度下降的优化速度，而且也会对网络的精度有些许提升。本文采取的图像预处理方式是在训练过程中使用几何变换方法对图片进行数据增强，具体实现在第四章会进行概述。下面对第二种图像预处理方式进行讨论研究。

（1）图像噪声

在图像处理中，图像噪声在图像上以一些孤立的像素点或者像素块出现，一般噪声是指会对图像的有用信息进行干扰致使图像变得不清晰的无用信息。图像的噪声通常是在图像获取过程中或在图像传输过程中产生的。

图像常见噪声有高斯噪声和椒盐噪声等。当噪声的概率密度服从高斯分布时这一类的噪声统称为高斯噪声，高斯噪声一般产生于在温度较高或者光线较弱的环境。椒盐噪声又称脉冲噪声，当图像的一些像素值被随机改变时就会产生椒盐噪声。椒盐噪声在图像中一般以白点或者黑点的形式随机出现。图 2.9 展示了原图以及添加了高斯噪声和椒盐噪声的图像。

（a）原图 （b）加入高斯噪声的图片 （c）加入椒盐噪声的图片

图 2.9 添加图像噪声前后图像对比

（2）光照变化

在图像处理中，一般可以改变图片的亮度或者对比度来模拟现实环境中的光照变化，图像中的亮度或者对比度调节主要体现在对图片的明暗程度的的改变。图 2.10 展示了在调节亮度或对比度后图像的变化。

图 2.10 调节亮度和对比度前后图像对比

2.2.3 特征提取

在面部表情识别中最关键的步骤就是特征提取，在基于卷积神经网络的面部表情识别中，对面部表情特征进行提取的工作是网络自动完成的。首先确定一个合适的卷积神经网络结构对网络模型进行无监督学习，然后确定合适的超参数后就可以进行网络的训练，最后迭代至网络收敛。

2.2.4 特征分类

特征分类往往是和特征提取密不可分的，而在卷积神经网络结构的最后往往需要一个全连接层实现对网络中提取的特征进行分类输出，从而得到最终的分类结果，也就是本文 2.1.5 中介绍的卷积神经网络的全连接层和分类器，对于面部表情识别任务而言，最终的表情类别输出通常为七类或八类，所以一般选择 Softmax 作为最终的分类器。

2.3 本章小结

为了更好地学习基于深度学习的面部表情识别，本章对其研究方法的理论背景进行了介绍。2.1 节首先从卷积神经网络和全连接神经网络的对比引出对卷积神经网络的介绍，然后从卷积层、激励层、池化层、全连接层这四部分介绍了卷积神经网络的基本结构；2.2 节介绍了目前应用广泛的基于卷积神经网络的面部识别的流程，分别从人脸检测、图像预处理、特征提取和分类进行了详细介绍。

第三章 基于深度可分离卷积的 DCNN

在这一章，我们提出并构建了一种基于深度可分离卷积的深度卷积神经网络(DCNN)。本章首先介绍了一种经典的卷积神经网络：VGGNet，然后基于 VGGNet 的网络结构针对面部表情识别任务进行了优化改进，并提出了改进的经典卷积神经网络 VGG12，然后再介绍了基于深度可分离卷积的 DCNN 网络的结构。最后，本章通过仿真实验对改进的经典卷积神经网络和基于深度可分离卷积的 DCNN 进行性能对比，最终选择了 DCNN 作为面部表情识别系统的特征提取网络。

3.1 经典卷积神经网络结构设计

在传统的机器学习研究中，特征提取一直是一个难题。它们大多采用人工提取特征的方法，然后利用人工提取的特征训练分类器进行分类。然而，手工提取特征会存在人为因素的干扰，导致模型泛化能力不足从而无法实现较高的识别准确率。卷积神经网络(CNN)可以自动提取形成更抽象的高层特征或属性特征，具有更好的鲁棒性从而提高最终的分类或预测精度。因此通过 CNN 提取面部表情特征成为现阶段广大研究学者的首选，但现有的基于 CNN 的面部表情识别模型存在诸多问题：

1、深度卷积神经网络的本质是训练数据驱动，采用已有的模型进行训练，可能产生模型泛化能力弱等问题。

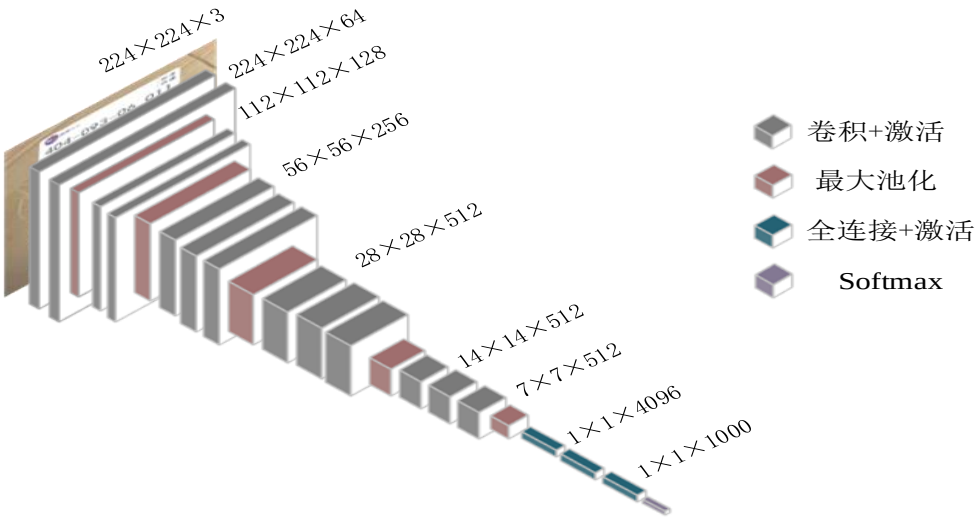
2、盲目加深或加宽深度卷积神经网络的模型结构，导致结构复杂，产生过拟合现象。

3、为了增大感受野，充分获取数据中的关键信息，使用大卷积核，如7×7，11×11等，造成网络的参数量巨大，训练时间过长。

针对以上问题我们设计了一个基于 VGGnet 的改进的经典卷积神经网络结构。VGGNet 的特点是简单，它在只使用小卷积核的卷积层的前提下，在深度上进行了拓展，将卷积层组成的卷积块层叠在一起。但是 VGGNet 是针对 ImageNet 数据集而提出的分类网络，而大部分面部表情数据集的要比 ImageNet 数据集小得多，为了避免网络过于复杂产生过拟合现象，使 VGGNet 能够在面部表情识别任务上表现的更加出色，我们将 VGGNet 简化，并添加了 Dropout 以避免模型过拟合。

3.1.1 VGGNet

VGGNet (Visual Geometry Group Network)网络[38]是 Simonyan 和 Zisserman 在 2014 年提出的网络，VGGNet 在当年的 ImageNet 竞赛的定位与分类任务上分别取得了冠军与亚军。得益于其简洁的网络结构与出色的性能，VGGNet 在计算机视觉领域获得广泛的应用 VGGNet16 是 VGGNet 网络中使用最为广泛的，其结构如图 3.1 所示，VGGNet16 网络结构中共有 13 个卷积核都为3×3的卷积层，这 13 个卷积层分为五个卷积块，前两个卷积块包含两个卷积层的堆叠，后三个卷积块包含三个卷积层的堆叠。块与块之间的使用最大池化层连接。最大池化层可以在加强图像特征不变性的前提下对卷积层输出的特征图做更一步降维，减少计算量。最后经由三个全连接层输出特征分类。



(a) VGGNet16 网络结构图

(b) 按块划分的 VGGNet16 网络结构图

图 3.1 VGGNet16 的网络结构

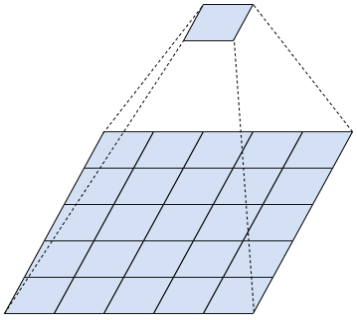
相比于之前的经典卷积网络结构,VGGNet 的主要创新点在于：

(1) VGGNet 中使用3×3的卷积层替代了 AlexNet 中5×5、7×7、11×11的大卷积核，使用小尺寸卷积核的卷积层可以使网络的参数量得到了大大缩减。假设网络中输入与输出的深度（通道）大小分别为C_{in}与C_{out}，则在 VGGNet 中一个3×3的卷积层的参数量为：

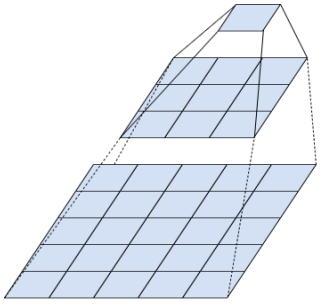
$C_{in} \times 3 \times 3 \times C_{out} = 9C_{in} \times C_{out}$，而 AlexNet 中一个$7 \times 7$的卷积层的参数量为：$C_{in} \times 7 \times 7 \times C_{out} = 49C_{in} \times C_{out}$。

(2) 在网络中使用小尺寸卷积核的卷积层可以通过加深网络来保证网络更加优秀的特征提取能力。如图 3.2，在 VGGNet 中经过两次卷积核为3×3的卷积操作得到的感受野与 AlexNet 中进行一次卷积核为5×5的卷积操作得到的感受野大小一致，同理在 AlexNet 中进行一次卷积核为7×7的卷积操作与 VGGNet 中经过三次卷积核为3×3的卷积操作得到的感受野一致。这说明两种卷积操作在图像特征提取的功能是类似的。

(3) 为了增强网络的非线性能力，避免网络产生梯度消失情况，VGGNet 在每个卷积层后增加了非线性激活函数。



(a) 一次5×5的卷积操作



(b) 两次3×3的卷积操作

图 3.2 不同尺寸的卷积操作的对比

3.1.2 Dropout

在深度学习的模型训练中，如果网络太过于复杂，导致模型的参数太多而训练集的样本数量又极其有限，训练得到的模型很容易过拟合。Dropout 是深度学习中经常采用的一种有效的防止模型过拟合，使其达到正则化效果的方法。Dropout 的做法可以简单的理解为在卷积神经网络的训练的过程中以指定的概率丢弃网络中的部分神经元，即让被丢弃的神经元的输入输出都为 0。Dropout 对神经网络的操作可以实例化表示为图 3.3。

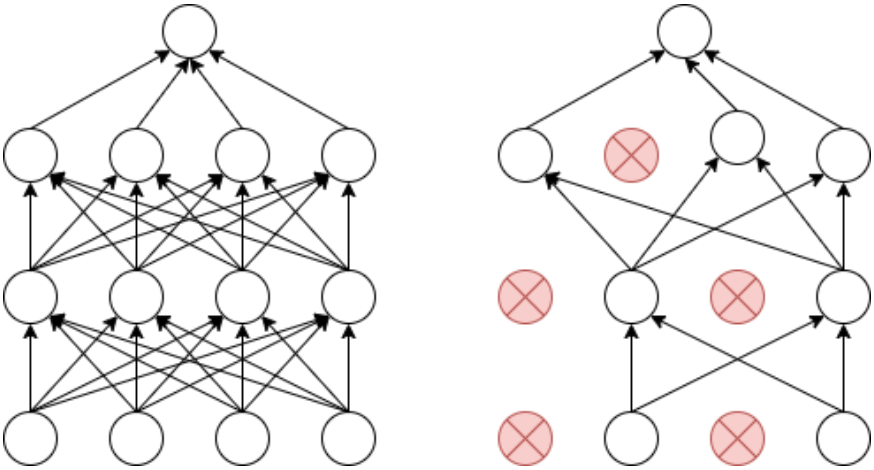


图 3.3 Dropout 对神经网络的操作实例化

可以从下面的两个角度去理解 Dropout 的正则化处理：

（1）Dropout 在每一轮训练中都会随机丢弃一定比例的神经元，这个操作相当于对多个卷积神经网络进行取平均值，因此最终得到的模型具有普适性。

（2）Dropout 可以有效的消除神经元之间存在的依赖性。对网络中的神经元进行随机筛选剔除操作可以增强网络的稀疏程度，即某些神经元存在的固有联系可能会产生一些负面特征干扰训练，而通过 Dropout 可以有效地过滤掉这些特征，从而有效的增强了网络的特征表达能力。

3.1.3 改进的经典卷积神经网络结构

VGGNet 是在 ImageNet 竞赛上针对 ImageNet 数据集而提出的分类网络，而大部分面部表情数据集的要比 ImageNet 数据集小得多，为了避免网络过于复杂产生过拟合现象，我们对其采取了一些改进措施，具体表现在：大大减少了所有卷积层和全连接层中的滤波器数量；将 VGGNet 原本的五个卷积块缩减为四个卷积块，去除最后一个全连接层，并添加了 Dropout。改进后的 VGGNet 网络仅有 12 层，我们将改进后的网络命名为 VGG12。

VGG12 的网络结构具体介绍如下。如图 3.4，VGG12 由四个卷积块组成。每个卷积块中的卷积层的滤波器大小都为3×3的，每个卷积块中的卷积层中的滤波器数量皆相同，四个卷积块的滤波器数量分别为 32，64，128，256。最后，通过两个全连接层至输出层，每个全连接层都添加了 Dropout 层，防止模型过拟合。输出层是一个完全连通的层，有 7 个神经元对应面部表情数据集的 7 个情感类别。

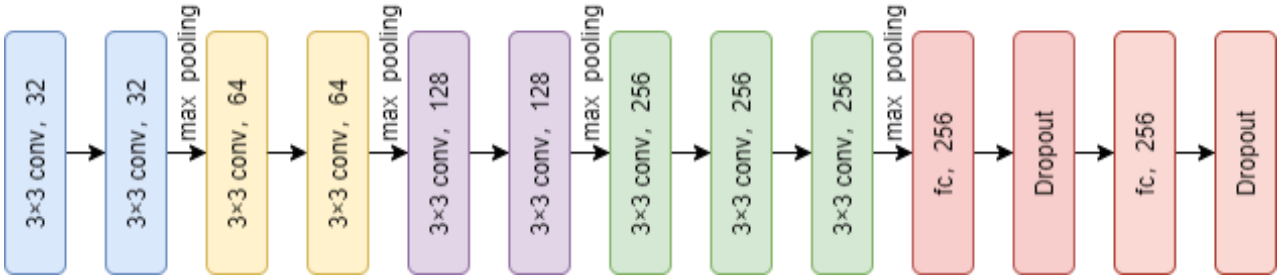


图 3.4 改进的经典卷积神经网络结构

3.2 基于深度可分离卷积的改进的神经网络结构设计

在面部表情识别任务中，经典的卷积神经网络的都是使用单一尺度卷积核的卷积层来对图片局部的视觉信息进行整合，而且在模型的结构设计上都是通过在网络深度上进行拓展，即一味地增加卷积层的数量来提高模型的特征提取能力。有研究表明[59]，不断地加深卷积神经网络的深度容易给网络带来两个极大的弊端：1、梯度爆炸和梯度消失：目前神经网络的优化方法都是依据损失函数计算得出的误差采取梯度反向传播算法，然后不断更新网络各隐藏层的权重值。伴随着网络深度的增加，梯度的更新程度将会以指数级别增加或减少，当梯度增加时易产生梯度爆炸，减少则易产生梯度消失。2、过拟合：当面部表情数据集不完备或者数据集的规模过小的时候，如果一味的增加卷积核的个数或者网络的层数，则会导致神经网络模型的参数量指数级地增加，其性能的提升仅仅体现在训练集样本识别精度的提高，模型在验证集上的表现并不优秀，最终造成模型的泛化能力不足的情况下，也会花费巨大的训练时间和浪费计算资源。

为了避免产生上述问题，在保证模型的较少参数量的前提下提高模型的识别精度，提出了基于深度卷积可分离卷积的卷积神经网络。我们提出的 DCNN 网络在不产生过多的训练参数前提下，结合卷积残差块的使用，让网络可以从不同的感受野进行特征提取工作，进一步提高模型的鲁棒性提高精度。此外为了尽可能的保持网络的轻量化，防止过拟合的发生，将经典卷积神经网络中的全连接层用全局平均池化层进行替换。

3.2.1 网络结构设计

卷积神经网络在进行特征提取工作时具有层次性，在接近输入层的卷积层所关注的特征一般是图片的边缘、颜色、纹理等这些易于表达的特征；在远离输入层的卷积层所关注的特征一般是复杂的高级语义特征。据此，此节构建的基于深度可分离卷积的卷积神经网络参考

了上文提到的改进的 VGGNet 的网络结构设计，核心结构是四个卷积块的堆叠,为了保证模型较少的参数量便于训练，在核心模块中使用深度可分离卷积层替代了原来的卷积层。深度可分离卷积层可以保证网络在拥有更少的参数量前提下，特征提取能力与使用同等大小的卷积层相差无几。为了防止网络出现梯度爆炸或梯度消失的情况，我们在卷积块中引入了残差块结构。此外为了进一步加快网络的收敛速度提高模型的泛化能力，我们将批归一化层添加至深度可分离卷积层后。卷积块的具体结构如图 3.5 所示。

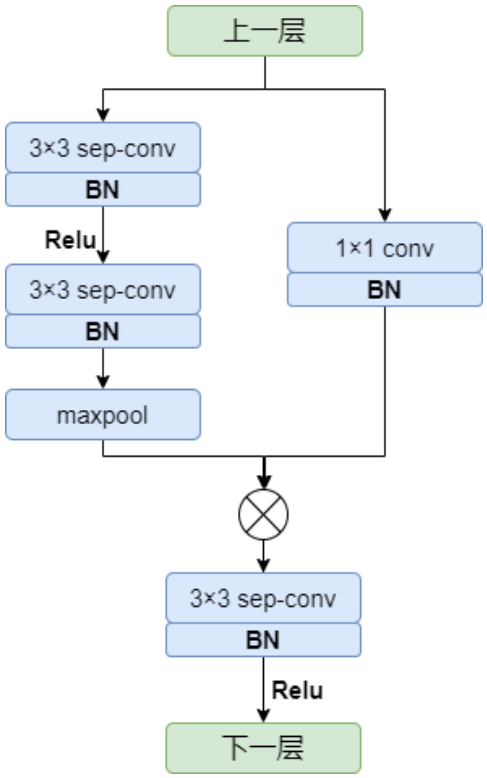


图 3.5 卷积块的结构

网络主要包含了三个模块：输入模块，中间模块和输出模块。如图所示。网络中的卷积层和深度可分离卷积层的滤波器即卷积核的尺寸都设置为3×3。输入模块将数据集通过两个卷积层输送到中间模块。中间模块则是由上文提到的卷积块堆叠四次而成，最后图片经由一个卷积层和全局平均池化层构成的输出模块进行输出。具体网络结构如图 3.6，我们将这个网络称为 DCNN。

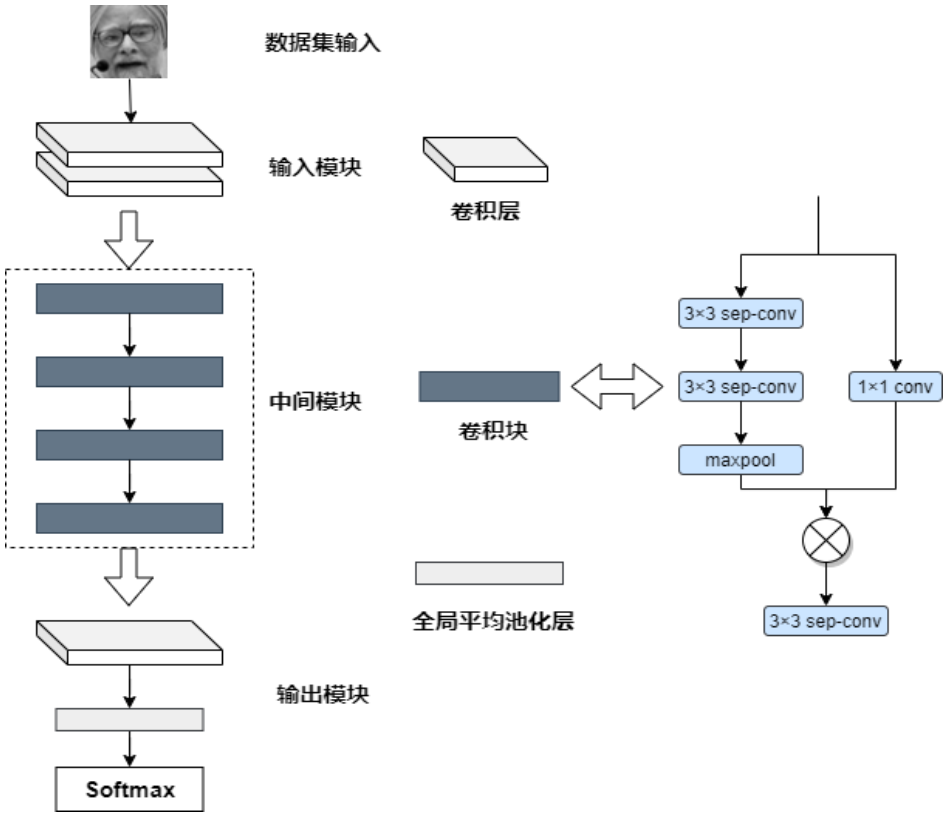
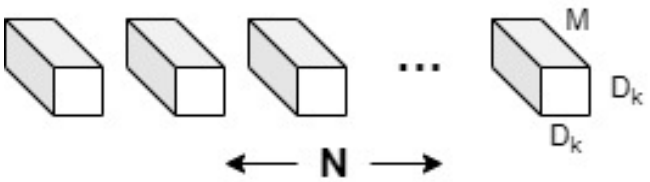


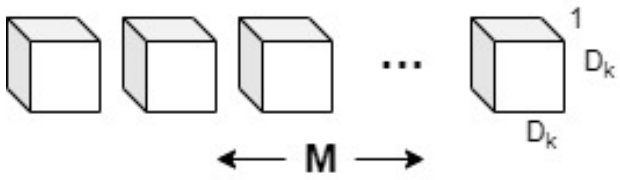
图 3.6 DCNN 网络的结构

3.2.2 深度可分离卷积

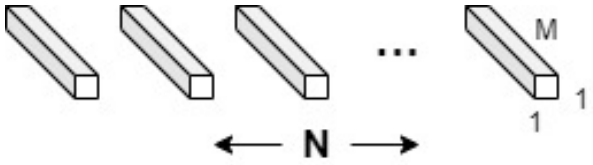
深度可分离卷积是一种特殊的卷积形式，被 Howard 等在 2017 年首次提出并应用于 MobileNet 网络。深度可分离卷积是深度卷积和点卷积（1×1卷积）的结合。深度卷积首先对每个输入通道分别执行深度卷积，然后通过点卷积（1×1卷积）将输出通道混合。即在一个标准的卷积中将输入同时进行滤波和组合操作，但深度可分离卷积将这个操作进行拆分，第一步将输入进行滤波操作，第二步再将其进行组合操作。这种将标准卷积进行分解的操作可以有效的减少运算代价和网络模型的尺寸。图 3.7 显示了一个标准的卷积操作是如何分解为深度卷积和点卷积（1×1卷积）的。



(a) 标准卷积的滤波器



(b) 深度卷积的滤波器



(c) 点卷积的滤波器

图 3.7 标准卷积和深度可分离卷积的对比

大小为$D_F \times D_F \times M$的特征图F作为一个标准卷积的输入，则输出的特征图G为$D_G \times D_G \times N$，其中D_F是输入特征图F的空间宽度和高度，M是输入的通道数，D_G是输出特征图G的空间宽度和高度，N是输出的通道数。

一个标准的卷积层由尺寸$D_K \times D_K \times M \times N$的卷积核参数化，其中卷积核的尺寸假设为$D_K$的平方，$M$和$N$分别是上文定义的标准卷积层的输入通道数和输出通道数。

则经过一个标准卷积层所需要的计算成本C_{conv}为式（3.1）。

$$C_{conv} = D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F \tag{3.1}$$

其中计算成本与输入通道数M输出通道数N，卷积核的尺寸$D_K \times D_K$和特征图的大小$D_F \times D_F$正相关，由式 3.1 可知卷积核的大小与输出通道数存在着相互影响，而使用深度可分卷积则可以消除这种影响。

标准卷积运算的原理就是通过卷积核将输入特征进行滤波和组合操作，然后产生新的特征输出。为了大大减少计算代价，可以将标准卷积中的滤波和组合的操作分解成独立的两个步骤，把这种分解操作即为深度可分离卷积。

深度可分离卷积是由深度卷积和点卷积顺序构成。首先深度卷积将一个滤波器作用于输入特征图的每个通道，然后再在输出后使用点卷积(简单的 1×1 卷积)进行线性叠加。

对每个输入通道(输入深度)应用一个滤波器的深度卷积可表示为式（3.2）。

$$\hat{G}_{k,l,m} = \sum_{i,j,} \hat{K}_{i,j,m} \cdot F_{k+i-1,l+j-1,m} \tag{3.2}$$

其中$\hat{K}$是深度卷积的卷积核的尺寸，大小为$D_K \times D_K \times M$，进行滤波后的输出特征图$\hat{G}$的第m个通道是$\hat{K}$中的第m个卷积核应用于F的第m个通道产生的。

深度卷积的计算成本C_{dconv}为式（3.3）。

$$C_{dconv} = D_K \cdot D_K \cdot M \cdot D_F \cdot D_F \tag{3.3}$$

深度卷积与标准卷积相比，其效果显著，但是也有局限性，深度卷积只对输入通道进行了滤波，并没有将滤波后的特征进行组合并输出新的特征，即仅仅实现了标准卷积中的滤波功能。所以为了对滤波后的特征计算其线性组合，需要在深度卷积的输出后添加一个1×1的点卷积。深度卷积的输出特征图大小为为$D_F \times D_F \times M$，则经过点卷积操作的计算成本C_{pconv}为式（3.4）。

$$C_{pconv} = M \cdot N \cdot D_F \cdot D_F \tag{3.4}$$

深度卷积和逐点卷积（1×1卷积）的线性组合称为深度可分离卷积。深度可分离卷积的计算成本C_{dsconv}为深度卷积和1×1逐点卷积的计算成本之和为式（3.5）。：

$$C_{dsconv} = D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F \tag{3.5}$$

通过将标准卷积拆分成滤波和组合两步，可以大幅度地减少参数计算量，深度可分离卷积的计算成本C_{dsconv}和标准卷积的计算成本C_{conv}具体的参数对比如式（3.6）。

$$\begin{aligned} \frac{C_{dsconv}}{C_{conv}} &= \frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F}{D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F} \\ &= \frac{1}{N} + \frac{1}{D_K^2} \end{aligned} \tag{3.6}$$

当我们使用卷积核为3×3的深度可分离卷积时，其计算量是使用同等尺寸卷积核的标准卷积的$\frac{1}{8}$至$\frac{1}{9}$，而精确度则基本没有差距。

3.2.3 批归一化层

在神经网络的训练中，伴随着隐藏层中的参数的不断变化，每一层的输入分布也会随之发生改变。针对这一现象，往往需要在训练开始前细致的进行参数初始化，以及在训练过程中设置较小的学习率，但这又会导致训练速度减慢，从而难以产生饱和的非线性的网络模型。一般的神经网络训练中仅仅对输入层的数据集进行统一的归一化处理，却忽视了中间隐藏层的归一化处理。即使对输入数据集进行了一次归一化处理，但是在隐藏层中经过一系列的数学运算和处理后其数据的分布规律极有可能产生变化，且伴随着神经网络深度的加深，数据分布的差异化就会变得愈加明显。所以这就需要在神经网络的隐藏层中也对数据进行归一化处理，即批归一化（Batch Normalization）。将批归一化应用于神经网络之中不仅可以加快网络模型的收敛速度从而加快训练速度也可以提高模型的泛化能力与训练精度。

批归一化层按照每个训练批次对上一层传入的数据进行归一化处理，将每个批次中的传入的样本数据归一化为标准正态分布（均值为 0，方差为 1）。假设批处理（mini-batch）的传入数据x为：$B=\{x_{i,\ldots,m}\}$，则经过批归一化后的响应为：$y_i=BN_{\gamma,\beta}(x_i)$

批归一化层对单位批次的传入数据的计算具体步骤如下：

（1）计算每一个批次传入的样本数据的均值。

$$x\mu_B=\frac{1}{m}\sum_{i=1}^{m}x_i \tag{3.7}$$

其中m指的是单位批次传入的样本数据的数量，x_i为单位批次传入的样本数据。

（2）计算每一个批次传入的样本数据的方差。

$$\sigma_B^2=\frac{1}{m}\Sigma_{\mathrm{i}=1}^{m}(x_i-\mu_B)^2 \tag{3.8}$$

其中μ_B为单位批次的样本数据均值。

（3）使用第一步和第二部计算求得的均值与方差对此批次的传入的样本数据进行归一化处理，获得$0-1$分布。

$$\hat{x}_i=\frac{x_i-\mu_B}{\sqrt{\sigma_B^2+\varepsilon}} \tag{3.9}$$

其中σ_B^2为单位批次的样本数据的方差，ε是为了避免除数为零而添加的一个极小的正数ε。

（4）尺度变换和偏移：将归一化的数据乘以一个参数γ用以尺度变换，在加上数值为β的偏移量，得到单位批次输入的样本数据的输出为式（3.10）。

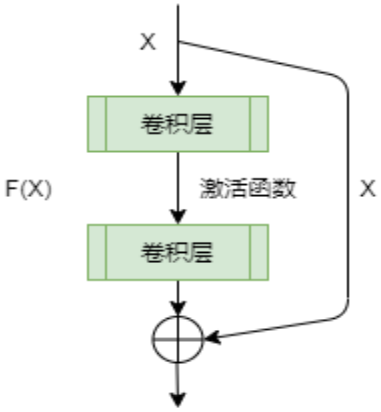
$$y_i=\gamma\hat{x}_i+\beta \tag{3.10}$$

其中参数γ是尺度因子，参数β是偏移因子，这两个参数都是网络在训练过程中自我学习得出的。

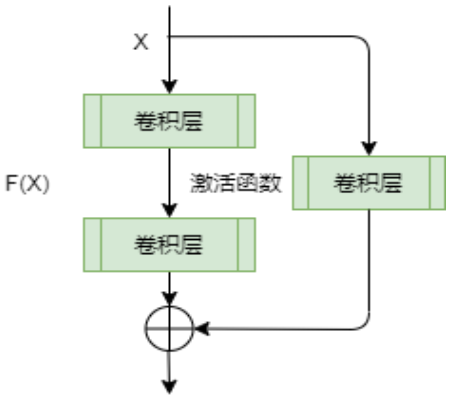
3.2.4 残差模块

在理论上，对神经网络的深度进行拓展即增加新的隐藏层，往往通过完备的训练后就可以有效地提高训练精度，因为在添加新层后的新模型的解的集合是原模型解的子集。一般来说，新模型要达到和原模型相同的训练精度，只需要将新的隐藏层训练成$f(x)=x$这样的恒等映射即可。此外，由于对神经网络拓展了深度，所以训练出的新模型有可能得到与原模型相比更优的解来拟合相同的训练数据集。但是在实际实验中，盲目添加过多的隐藏层，加深神经网络的深度反而容易使网络出现梯度消失和梯度爆炸问题，训练精度不升反降从而达不到预期目标。针对上述的问题，何恺明的团队提出了残差网络。残差网络中采取了跳跃连接结构。跳跃连接（Skip connection）可以从神经网络的任意的隐藏层中得到激活值，然后跨越

中间的网络层，传送给神经网络中较深的层。残差块的结构如图 3.8 所示。



(a) 恒等残差快



(b) 卷积残差块

图 3.8 残差块的结构

图（a）是恒等残差块，也是 ResNets 中使用的标准残差块，对应于输入与输出尺寸一致的情况。图（b）是另一种类型的残差块，叫做卷积残差块，主要应用于输入与输出尺寸不同的情况。卷积残差块与恒等残差块最大的不同就是在跳跃连接结构中加入了一个卷积层。跳跃连接结构中的卷积层主要作用是将输入数据X调整为不同尺寸，以便于跳跃连接中的数据返回到主路径时与最后相加的数据尺寸相匹配。跳跃连接结构中的卷积层后无需添加任何的激活函数，因为添加卷积层的作用仅仅是相当于一个线性函数来减小输入的大小，使得数据的大小可以与之后相加后的结果进行对照。

3.2.5 全局平均池化层

在经典的卷积神经网络的输出层中，一般都会设置及几层全连接层作为整个网络的“分类器”。但是全连接层存在着参数冗余的缺陷，全连接层的参数量一般占整个卷积神经网络参数量的 80%左右。参数量巨大往往会导致训练缓慢，模型发生过拟合现象，这也是在模型中

使用全连接层不可避免的弊端。为了避免网络参数冗余带来的负面影响，我们可以使用全局平均池化层来替换网络中的全连接层。既消除了全连接层所带来的不可知性，又有效地减少了全连接层所带来的过多参数量。

3.3 实验结果分析

3.3.1 面部表情数据集

在深度学习中，数据集的选取对神经网络模型的训练有着直接影响。因此本章实验选择了 Fer2013 数据集对上文提出的两个模型进行的性能进行评估。

Fer2013 数据集是 2013 年 Kaggle 面部表情识别大赛的官方数据集，大部分图片都是从网络爬虫上下载的，包括不同年龄、不同角度、部分遮挡等的图片。图 3.9 展示了一些表情的样本。Fer2013 共包含 35887 张 48x48 分辨率的灰度图像如图，在 35887 幅图像中，28709 幅用于训练，7178 幅用于测试。该数据集包含 7 种表情：愤怒、厌恶、恐惧、高兴、悲伤、惊讶和中性。每个表情分类的数据集的数量如表 3.1 所示：

图 3.9 Fer2013 数据集示例

表 3.1 每个表情分类的数据集的数量

表情	愤怒	平静	伤心	开心	厌恶	惊讶	恐惧
数量	4953	6198	6077	8989	547	4002	5121
比例	14%	17%	17%	25%	2%	11%	14%

3.3.2 实验结果与分析

本章提出了两个用于特征提取的基础网络，分别是基于 VGGNet 的改进的经典卷积神经网络 VGG12 和基于深度可分离卷积的 DCNN。为了选择使网络性能最优，本文分别对 VGG12 和 DCNN 网络进行训练和测试，使用网络的准确率和混淆矩阵对网络性能进行评价。

准确率是指分类正确的样本所占样本总数的比例，我们可以通过模型的准确率直观地评估网络模型的性能。

混淆矩阵在分类问题中是用来比较分类结果的预测值和实际测得值，本质上是一张表格。在一个 n 分类问题中，混淆矩阵则为一个$N \times N$的表格。一般在混淆矩阵中，行值代表了样本

的真实值，列代表了预测值。我们假设一个二分类问题的混淆矩阵如表 3.2 所示，其中将积极正确识别的样本数为 90，将积极错误识别为消极的样本数为 10；将消极正确识别的样本数为 70，将消极错位识别为积极的样本数为 30。将此混淆矩阵归一化后为表 3.3。代表了正确错误识别的比例。

表 3.2 混淆矩阵示例

	积极	消极
积极	90	10
消极	30	70

表 3.3 归一化后的混淆矩阵

	积极	消极
积极	0.9	0.1
消极	0.3	0.7

上文中提出的改进的经典卷积神经网络 VGG12 和基于深度可分离卷积的 DCNN 网络，两者在结构上的设计都是通过卷积块的堆叠来进行特征提取工作，所以我们首先需要通过实验来对比验证两个网络中不同的卷积块的数量对模型的特征提取能力和模型准确率的影响。

图 3.10 说明了卷积块层数分别为 2，3，4，5 时，两个基础网络的准确率。可以看出当网络只有两个卷积块时，由于网络结构过于简单，参数量过小，网络只能提取到输入图片中易于表达的特征，无法充分提取特征，导致两个网络的网络性能都较差；当我们设置为三层时，VGG12 的准确率略高于 DCNN 网络，而当我们将卷积块设置为五层时，由于数据集较少而模型参数量过多，过于复杂，产生了过拟合的现象，所以相较于设置为四层的卷积块，两个网络的准确率有所下降。卷积块设置为四层两个网络的准确率达到最高点，所以从识别准确率来看，我们将两个基础网络的卷积块设置为四层。

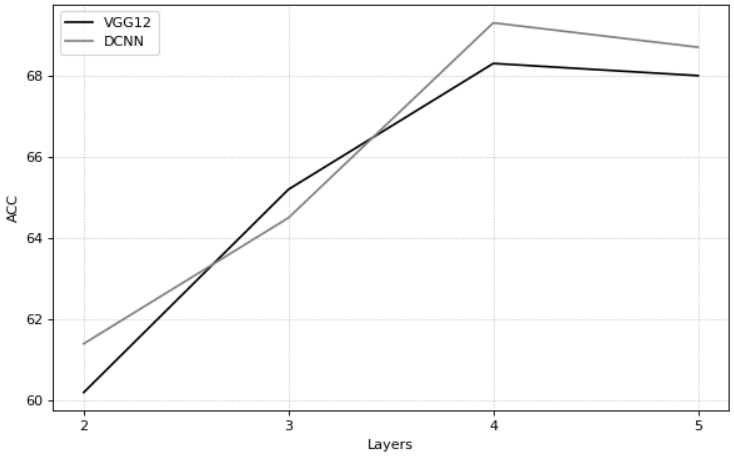


图 3.10 两个模型不同的卷积块的数量的准确率

在确定了基础网络的结构后，为了验证本文提出的网络的有效性，我们将提出的两个网络在 Fer2013 数据集上与其他网络进行对比。

表 3.4 不同模型的准确率

模型	准确率	参数量
VGG12	68.9%	2,375,783
DCNN	69.3%	298,311
VGGNET	67.2%	
DAM-CNN[60]	66.2%	
CNN-Ensemble[61]	65.1%	

由表 3.4 所示，其中准确率最高的是基于深度可分离卷积的 DCNN 网络。基于 VGGNet 的改进的经典卷积神经网络拥有比 VGGNet 更少的参数量，但是其准确率却比 VGGNet 提升了 1.7%;基于深度可分离卷积的 DCNN 网络由于深度可分离卷积和全局平均池化层的使用，参数量约为 VGG12 的八分之一，但是由于其结构的优越性，准确率比 VGG12 高 0.4%。因此本文选取了基于深度可分离卷积的 DCNN 网络作为基础网络。图 3.11 给出了针对每种情绪的 DCNN 模型和 VGG12 模型的面部表情估计对比，DCNN 与 VGG12 相比，能够以更高的准确率对情绪进行分类。

VGG12 和 DCNN 在测试集得出的混淆矩阵如表所示

表 3.5 DCNN 的混淆矩阵

真实值	预测值						
	愤怒	厌恶	恐惧	开心	伤心	惊讶	平静
愤怒	0.61		0.06	0.03	0.13	0.03	0.14
厌恶	0.11	0.75		0.03	0.09		
恐惧	0.11		0.50	0.03	0.17	0.06	0.12
开心	0.02			0.88	0.02		0.07
伤心	0.07		0.08	0.03	0.60	0.01	0.21
惊讶	0.03		0.09	0.06	0.02	0.77	0.03
平静	0.04		0.02	0.06	0.11	0.01	0.75

表 3.6 VGG12 的混淆矩阵

	预测值

真实值	愤怒	厌恶	恐惧	开心	伤心	惊讶	平静
愤怒	0.60	0.01	0.10	0.02	0.14	0.02	0.12
厌恶	0.15	0.68	0.05	0.03	0.08		
恐惧	0.12		0.48	0.04	0.18	0.07	0.11
开心	0.01		0.01	0.90	0.02		0.07
伤心	0.09		0.11	0.03	0.60	0.02	0.18
惊讶	0.02		0.09	0.04	0.02	0.79	0.02
平静	0.06		0.02	0.04	0.15	0.01	0.73

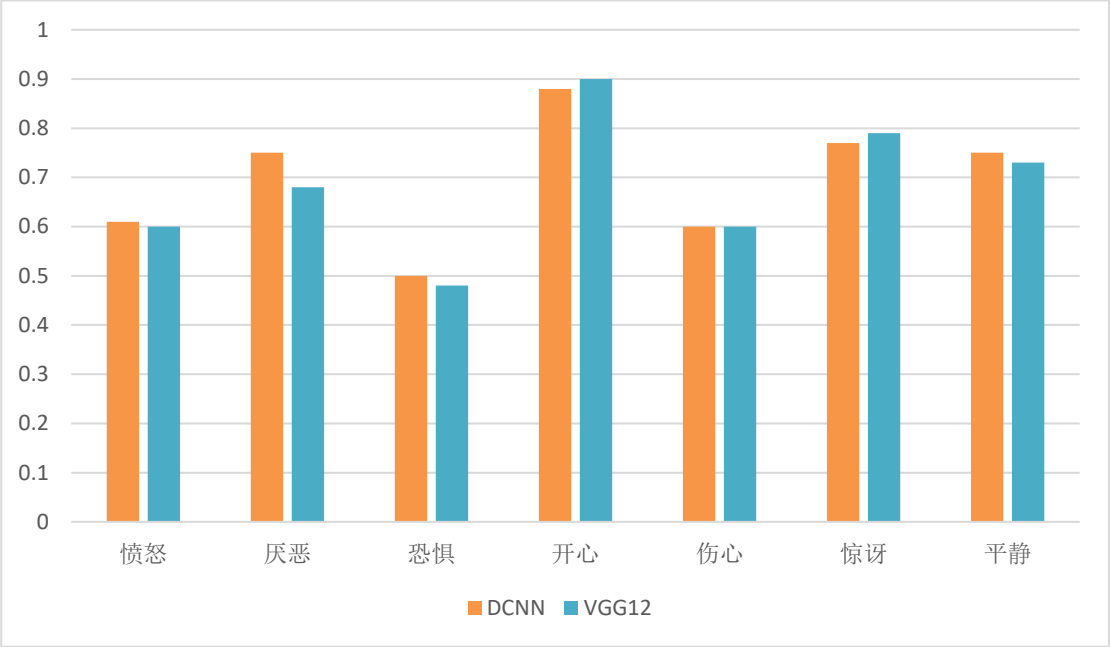


图 3.11 DCNN 模型和 VGG12 模型的面部表情估计对比

由表的数据可知，DCNN 和 VGG12 网络都对开心、惊讶和平静这三个表情识别精度都比较高，而恐惧、愤怒、伤心这三个表情的识别准确度则相对而言偏低。通过表 3.7 对各个表情分类的特征分析，原因可能如下：

识别准确率较高的表情都是特征较为明显的表情（开心具有嘴角上扬这唯一特征、惊讶则是在嘴部张开的同时瞪大眼睛、平静则是无任何面部变化特征）；而识别准确率较低的表情则具有容易与其他表情混淆的特征，其中恐惧和伤心最易被混淆。此外由于 Fer2013 数据集是从网络爬取的表情图片，如图 3.12 所示，数据集中存在许多有遮挡以及非正面甚至非表情的图像，导致网络无法提取到可识别特征，最终导致了总体的准确率偏低，最终影响分类结果。

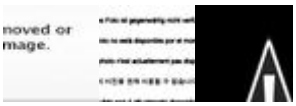
表 3.7 各种表情的面部特征

表情	眼部特征	嘴部特征

愤怒	眼睛瞪大且眼球微鼓	嘴唇紧闭；唇角拉直
厌恶	眉毛压低	嘴角下拉
恐惧	眉毛紧皱在一起	嘴部微张且嘴角后拉
开心	眼角有些许皱纹；眉毛略微下弯	嘴角上扬
伤心	眼睛微闭	嘴角下拉
惊讶	眼睛睁大；眉毛高且弯的抬起	嘴部张开；下颚下落；牙齿有些许露出
平静	无	无

(a) 面部有遮挡

(b) 非正脸图像



(c) 非人脸图像

图 3.12 负面图像示例

3.4 本章小结

本章针对目前面部表情识别任务提出了两个改进的卷积神经网络模型。首先在 3.1 节介绍了 VGGNet 的网络结构及其相对于其他网络的创新点，然后针对面部表情识别任务，对 VGGNet 进行了改进优化并提出了 VGG12 网络，具体操作为：在 VGGNet 的基础上减少了卷积层的层数和卷积层和全连接层中的滤波器数量，并添加了 Dropout 防止过拟合；然后在 3.2 节介绍了基于深度可分离卷积和卷积残差块的 DCNN 网络结构，深度可分离卷积和卷积残差块的使用可以有效地减少网络参数量；最后我们以准确率和混淆矩阵作为评价指标，通过仿真实验在 Fer2013 数据集上对这两个模型进行了性能对比实验，最终选择了基于深度可分离卷积的 DCNN 作为面部表情识别系统的特征提取网络。

第四章 基于注意力机制的 DCNN-CBAM

在第三章中我们提出了两个用于特征提取的网络结构，分别是基于 VGGNet 的改进的经典卷积神经网络 VGG12 和基于深度可分离卷积的 DCNN，通过在 Fer2013 数据集上的对比试验，结果表明，DCNN 在拥有更少的参数量的情况下准确率优于 VGG12，最终我们选择将 DCNN 作为我们面部表情识别任务中的基础特征提取网络。DCNN 经过实验验证是一个有效的特征提取网络，可以根据任务自动提取特征并对其进行分类。但是，图像的输入中缺乏多样性的特征极有可能会影响 DCNN 作为特征提取网络的学习效率，而计算机视觉中的注意力机制可以使神经网络忽略图片中无关特征而专注于有效信息。因此，本章将在 DCNN 模型的基础上添加注意力机制，从而提高模型的表情特征提取能力。首先，介绍了注意力机制的类型，并选择卷积块注意力模块（CBAM）作为 DCNN 中添加的注意力机制，接着介绍了网络训练情况，最后在确定了 DCNN-CBAM 网络的结构后，我们在 fer2013 数据集和 CK+数据集对模型性能进行了验证评估。

4.1 注意力机制

在日常生活中当我们关注到一个有趣的场景时，我们通常会聚焦于一个区别于其他类似场景的区域，并对这些区域进行快速处理识别。上述过程可以用公式表示为式（4.1）。

$$Attention = f(g(x), x) \tag{4.1}$$

式 4.1 中$g(x)$可以表示为关注特殊区域过程中产生的注意力。$f(g(x), x)$则表示为基于注意力$g(x)$来处理输入x，这与处理关键区域并获取信息的过程是一致的。根据上面对注意力机制公式化的定义，几乎所有现有的注意机制都可以用上述的公式进行表述。

注意力机制被引入到深度学习中来进行图片信息处理工作，在将注意力机制与深度学习中的神经网路结合的工作中，大部分都是使用掩码来形成用于特征提取的注意力机制。掩码的主要功能是通过使用一层全新的注意力权重，将特征进行区域划分并对每个区域的关键程度进行赋值，最后通过神经网络进行训练学习。掩码可以提取到每一张新的图片中需要关注的区域特征，最终形成注意力。这种思想形成了两种不同类型的注意力机制，分别是是软注意力(soft attention)和强注意力(hard attention)。

强注意力机制的判别过程是随机化，更加重视图像的动态变化，关注的重点是图像中的任意点也就是说注意力可以由图像中的随机一个点进行拓展延伸。此外，强注意力机制是不

可以进行微分的，这也就导致了它无法自动的提取图像特征的权重信息，通过训练得到权重的过程一般需要借助强化学习，无法集成于神经网络中。强注意力机制对提取到的特征的权重进行赋值时一般是把局部区域作为一个整体，当此区域的相关性较高时赋值为 1，而相关性较低时则赋值为 0，最终还需要神经网络对提取到的局部特征之间的关系进行更进一步的学习。

而软注意力机制则不存在上述硬注意力机制中存在的问题，所以广大研究人员也是将目光和重点都放在了软注意力机制中的研究工作中，目前软注意力机制被广泛应用于计算机视觉中的图像分类领域之中。软注意力机制学习到的权重大小主要依赖于图像特征之间的关系，且经过 Softmax 对特征采样后的权重进行处理后，所有的权重值分布于 0 至 1 之间，大部分以小数形式显示，各个特征之间的关系是权重和特征值的累加。软注意力机制与硬注意力机制最大的区别是软注意力机制是可以进行微分的，所以它可以集成到神经网络中通过神经网络的前向和反向传播来进行权重学习。从注意力关注的域来划分软注意力机制，一共有三种实现方式，它们分别是：空间域，通道域和混合域，其中混合域是空间域与通道域的结合。

4.1.1 空间注意力机制

在卷积神经网络中，一般使用池化层对输入的图片进行压缩，以达到减少参数量和计算量的目的，但是这种对图像进行的特征降维和信息合并操作在一定程度上可能会导致图片的一些重要特征无法提取并识别，空间注意力就是为了避免这种情况而提出的。空间注意力的主要原理就是将图像中的空间域信息映射到另一个空间域并且保留并提取图像中的关键信息。空间域的转换器最早由 Jaderberg 提出[50]。文中的空间域转换器主要被分为三个部分：定位网络、网格生成器以及最终的采样器。如图 4.1 所示，输入宽度、宽度和通道数分别是W、H和C的特征图$U \in R^{H \times W \times C}$，通过定位网络得到转换的参数$\theta$，接着网格生成器利用定位网络得到的参数$\theta$来形成采样网格，采样网络标志着输入特征图的采样形式，最后采样器将输入特征图和网格生成器生成的采样网络作为输入，输出特征图V。定位网络、网格生成器和采样器的三者有机结合共同构成一个空间域转换器。空间域转换器是一个即插即用的模块，可以插入到卷积神经网络的任意隐藏层之中，从而构成各种空间域转换网络。

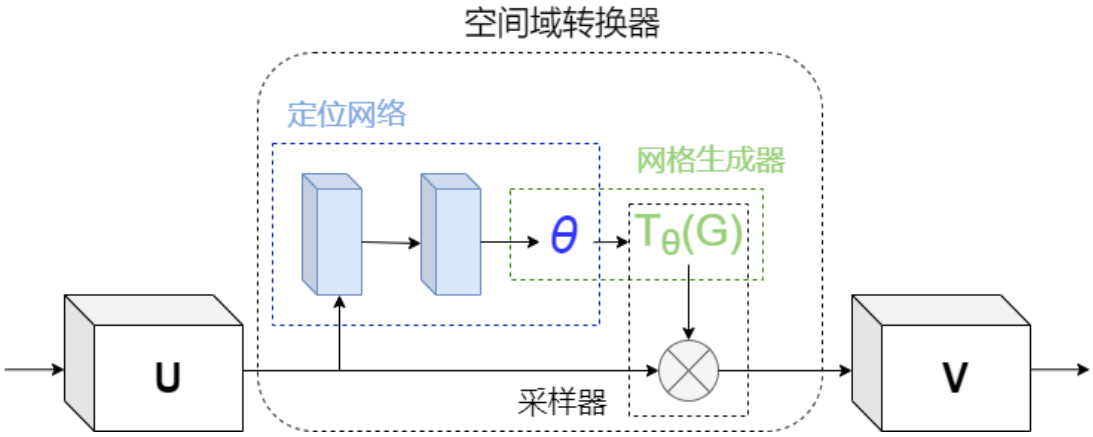


图 4.1 空间域转换网络的结构

图像可以通过空间注意力机制进行空间域转换保留并提取重要信息，也可以减弱在网络中由于扭曲、旋转、尺度变换等带来的影响特征提取的负面影响。

4.1.2 通道注意力机制

通道注意力机制和信号与系统中的信号处理的原理类似。在信号与系统中，所有的信号都可以用正弦波的线性叠加组合进行表示，信号在进行时频变换后，就可以用一个频率信号数值表示原本在时域上连续的正弦波信号。

而在卷积神经网络中，输入的图像一般都会由 RGB 三个通道进行初始化，在网络中经过不同的卷积层后都会产生新的信号。假设输入的 RGB 图像通过一个卷积核为 16 的卷积操作，会生成一个新的矩阵，新矩阵的通道数为 16，这 16 个通道的特征其实就是输入图像在不同卷积核上的占比。用信号中时频的概念去理解的话，卷积层使用卷积核对图像进行卷积操作类似于信号与系统中对信号进行傅里叶变换操作，从而将图像的特征分解为 16 个卷积核中的信号分量。既然特征被分解成不同的信号分量，那么在这新产生的 16 个信号分量中即通道中，对于关键信息的贡献必然存在大小差异，对于这种不同通道对图像关键信息贡献存在差异情况，就需要给每个通道赋予不同的权重值，因此，当通道的贡献值越大赋予的权重值就相应的变大，贡献值较小则权重较低，这样可以使包含关键信息的通道更容易被网络注意。

在将通道注意力机制的研究工作中，最为突出的就是 Hu 等人[51]提出的 SENet（Squeeze and excitation networks），SENet 将关注重点从空间转移到通道之间的关系，并由此提出了一个新的模块化结构：SE 块，主要原理是通过建立通道之间的依赖关系来自动调整并重新校准不同通道的特征响应。SE 块的本质为通道注意力机制，即依据具体任务确定各个通道特征的重要性，再根据重要性程度对各个通道赋予不同的权重来决定抑制或加强不同的通道信息。

SE 块的结构如图 4.2 所示，主要包括了三个部分，分别是：压缩、激励和缩放。首先通道数为c_1，宽度和高度为w_1和h_1的输入特征图X通过一列卷积操作成为大小为$w_2 \times h_2 \times c_2$的新的特征图$Y$，其中卷积操作的公式如式（4.2）所示。接着沿着空间维度对特征图Y进行压缩操作，即使用F_{sq}函数对特征图的每一个通道特征值累加再取均值，具体操作如式（4.3）所示。压缩操作可以使二维的通道被压缩成一个实数，并且输出的维度数目也对应了输入的通道数，最终使得全局感受野在靠近输入层也能被获取。然后，经过压缩操作的特征值经历一个激励操作，具体操作如式（4.4）所示。激励操作通过一个参数W使每个特征通道具有权重值，其中的参数W是被用来标识特征通道之间的联系性。最后经过激励操作的矩阵与特征图Y相乘如式（4.5），为每个特征通道进行特征选择由激励操作的矩阵上的权重值所决定，权重与特征的重要性程度正相关。通过乘法加权至特征图Y的特征上，最终完成了对原特征图的特征在通道上的标注重点区域，从而达到关注重点区域，抑制无关区域的通道注意力机制。

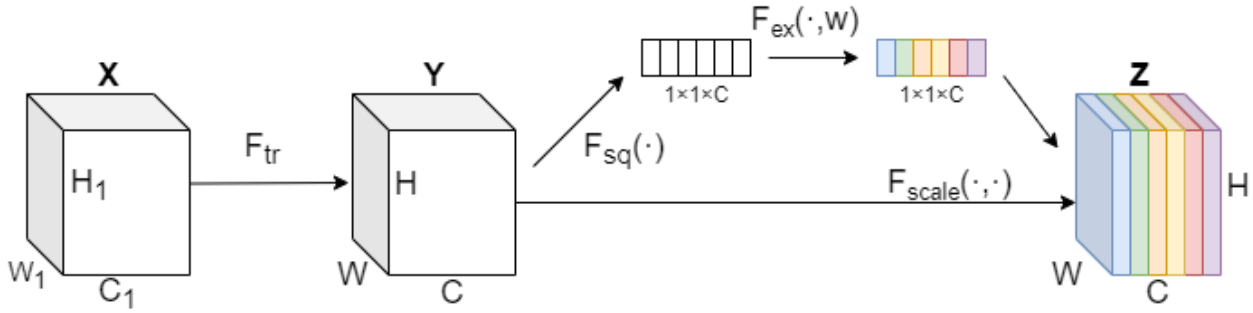


图 4.2 SE 块的结构

$$y_c = v_c \times X = \sum_{s=1}^{c_1} v_c^s \times x^s \tag{4.2}$$

$$z_c = F_{sq}(y_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} y_c(i,j) \tag{4.3}$$

$$s = F_{ex}(z, W) = \sigma\big(g(z,w)\big) = \sigma\big(W_2 \delta(W_1 z)\big) \tag{4.4}$$

$$z_c = F_{scale}(u_c, s_c) = s_c u_c \tag{4.5}$$

上式中，X表示为输入的特征图，v_c为第c个卷积核，v_c^s表示为作用于输入特征图X的相应通道的v_c的单个通道，x^s表示为第s个输入通道，最终得到的y_c表示第c个通道的特征图Y，δ表示 ReLU 激活函数，σ是 sigmoid 函数，W_1和W_2都是学习到的权重，s_c表示第c个通道的权重。

4.1.3 空间和通道混合注意力机制

在上面两小节分别介绍了软注意力机制中的空间和通道注意力，注意力机制和特征图利用的重要性已通过空间域转换网络和 SENet 的验证。但 SE-Network 仅考虑通道在特征图中

的作用，忽略了特征图的空间重要性，空间域转换网络也存在着相似的问题，它忽视了特征图中通道的影响，而本文在 DCNN 中引入的卷积块注意力模块（CBAM）与只关注通道的 SE 块和只关注空间的空间域转换器不同，它是一种结合了空间（spatial）和通道（channel）的注意力机制模块。CBAM 是一种轻量、通用的注意力模块，CBAM 可以使网络在训练过程中对训练目标更加关注。图 4.3 展示了 CBAM 的具体结构。输入的特征图为$F \in R^{C\times H\times W}$，$C$代表通道数，$H$和$W$分别代表特征图的高度和宽度，首先，如式（4.6）将通过 CBAM 中的通道注意模块处理后的特征图与输入特征图相乘得到新的特征图F'，然后，如式（4.7）再将F'和通过空间注意模块处理后的F'再次进行相乘操作得到最终的输出特征图F''。对F进行如下处理：

$$F' = M_c(F) \otimes F \tag{4.6}$$

$$F'' = M_s(F') \otimes F \tag{4.7}$$

上式中⊗表示逐元素乘法，M_C和M_S分别代表一维通道注意力图和二维空间注意力图，F''是最终的精炼输出。卷积块注意模块(CBAM)使得注意模块可以同时应用于通道维度和空间维度，所以我们将从通道注意模块和空间注意模块两个方面来介绍 CBAM 的工作原理。

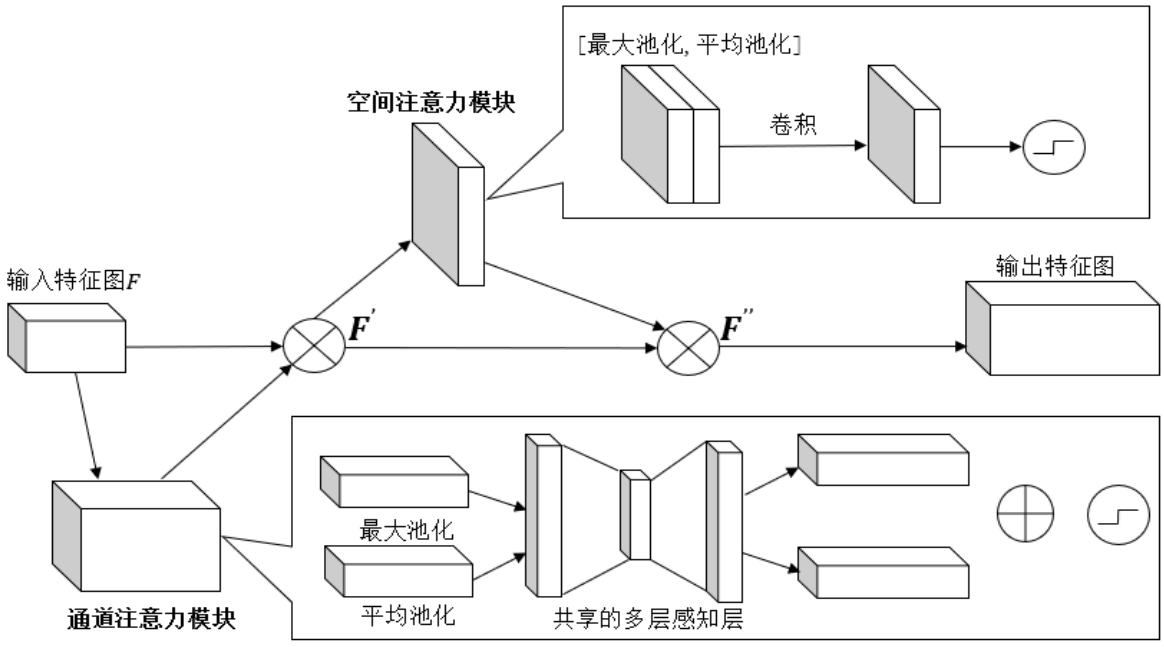


图 4.3 CBAM 的具体结构

首先介绍 CBAM 中的通道注意力模块，由于特征图中的任一通道都可以作为一个特征检测器，因此可以利用特征通道之间存在的联系来计算通道注意力图最终形成通道注意力模块。为了有效地计算通道注意力，将输入的特征图$F \in R^{C\times H\times W}$分别通过平均池化操作和最大池化操作对其在空间上进行压缩，生成两个大小$1\times 1\times C$的特征图，分别为F^c_{avg}和F^c_{max}。接着这两个被压缩后的特征图被传到由多层感知器（MLP）与一个隐藏层组成的共享网络中，然后将

共享网络输出的特征图进行对应元素逐个相乘并加和的操作，再经过 sigmoid 激活函数，最终产生我们的通道注意力图，通道注意力的计算公式为式（4.8）和（4.9）。

$$M_c(F) = \sigma(MLP\left(AvgPool(F) + MLP\big(MaxPool(F)\big)\right) \tag{4.8}$$

$$M_c(F) = \sigma(MLP\left(AvgPool(F) + MLP\big(MaxPool(F)\big)\right) \tag{4.9}$$

$$M_c(F) = \sigma(W_1\left(W_0\big(F_{avg}^c\big) + W_0\big(W_0(F_{max}^c)\big)\right)$$

上式中σ表示 Sigmoid 函数，MLP为一个多层感知器，F为输入的特征图，$AvgPool()$和$MaxPool()$分别表示平均池化操作和最大池化操作，F_{avg}^c为输入的特征图F经过平均池化操作得到的特征图，F_{max}^c为输入的特征图F经过最大池化操作得到的特征图，W_0和W_1为两个特征图经过MLP所得到的权重，$M_c(F)$为最终得到的通道注意力图。

空间注意力模块通过对特征的空间进行处理来提取注意力，它与通道注意力模块有所区别却又互为补充。空间注意力模块的输入特征图F是通道注意力图和通道注意模块的特征输入逐元素相乘的结果，需要对特征图通过平均池化操作和最大池化操作对其在通道上进行压缩，得到两个$H \times W \times 1$的特征图并将它们进行通道上的拼接。然后通过一个7×7的卷积进行降维操作，在经过 Sigmoid 形成一个空间注意力图。空间注意力的计算公式为式（4.10）。

$$M_s(F) = \sigma(f^{7\times7}([AvgPool(F); MaxPool(F)]) \tag{4.10}$$

$$M_s(F) = \sigma(f^{7\times7}(\left[F_{avg}^s; F_{max}^s\right])$$

上式中σ代表 sigmoid 函数，$f^{7\times7}$代表一个卷积核大小为7×7的卷积操作,F为输入的特征图，$AvgPool()$和$MaxPool()$分别表示平均池化和最大池化操作，F_{avg}^s和F_{max}^s分别为输入的特征图F经过平均池化和最大池化得到的特征图。$M_s(F)$为最终得到的通道注意力图。

4.2 网络训练情况

4.2.1 开发环境

随着对深度学习领域的不断探索，各种深度学习的基础框架和环境也不断涌现。目前较为常用的框架有 Tensorflow[62]、PyTorch[63]、Keras[64]、Caffe[65]和 Theano[66]等。其中，Keras 深度学习框架是由 python 编程语言进行编写，其底层框架基于 Tensorflow。相比于其他的深度学习框架，Keras 具有以下特点:

（1）高度的模块自由化：在 Keras 中，所有的网络层、损失函数、激活函数、优化器等模块都可以单独进行编写编译，可根据具体需求对所有的模块进行自由组合。

（2）易拓展性：添加新的模块时仅需模仿现有模块编写新的函数或类和方法。

（3）向上兼容：Keras 封装于 Tensorflow 框架，可以和 Tensorflow 框架联合使用

（4）与 Python 协作：与需要添加额外的模型配置文件的 caffe 不同，Keras 框架中，模型仅由 Python 编写，使其更容易找出 bug 和进行扩展。

因此，本文选取 Keras 框架构建 DCNN 模型。训练模型的硬件配置和软件配置如表 4.1 所示。

表 4.1 本文模型训练时的硬件配置和软件配置

操作系统	Win10(64 位)
处理器	Intel(R) Core(TM) i5-9300HCPU@2.40GHz
显卡	NVIDIA GeForcce GTX 1650
深度学习框架	TensorFlow1.14-gpu
编程语言	Python3.6

4.2.2 数据集介绍

为了能完全评估基于注意力机制的 DCNN 的模型性能和泛化性，除了使用上一章中用于对比确定基础网络的 Fer2013 数据集以外，还引入了 CK+[67]数据集。

CK+数据集是目前面部表情分类领域中的一个重要的数据集，它是由 CK 数据集扩展得来。该数据集使用统一的硬件总共采集了 123 位实验对象的 593 个视频序列，实验对象 69%为女性，年龄分布从 18 岁至 50 岁不等。标注每个视频序列的最后一帧或几帧为情图像的面部动作单元标签。CK+数据集一共将表情分为七类，分别是愤怒、蔑视、厌恶、恐惧、开心、伤心、惊讶。与 Fer2013 数据集的分类不同的是，CK+数据集中用蔑视替代了平静。图 4.4 展示了 CK+数据集的一些表情数据，每个表情分类的数据集的数量如表 4.2 所示。

图 4.4 CK+数据集的表情示例

表 4.2 CK+数据集数据分布

表情	愤怒	蔑视	伤心	开心	厌恶	惊讶	恐惧
数量	135	54	84	207	177	249	75
比例	14%	6%	9%	21%	18%	25%	8%

4.2.3 数据增强

在深度学习中如果神经网络设计层数过深，而数据集数量偏少，神经网络便会产生过拟合现象，虽然适当减少层数可以减轻过拟合的现象，但是如果当数据中包含的信息过于丰富而数据量很少，例如 CK+数据集，仅仅调整神经网络的层数就无法避免过拟合现象，所以这就需要对数据集采取数据增强(Data Augmentation)操作来扩充已有的数据集，增强网络模型的泛化能力。此外，卷积神经网络对偏移位置、转换视角、改变尺寸等操作及这些操作的组合具有不变性，保障了数据增强操作的有效性。数据增强一般以增强阶段来进行划分，可以分为在线增强和离线增强。在线增强就是在网络训练的过程中，同时进行数据增强工作；离线增强就是在网络训练之前对数据集进行所有的变换操作，然后再将增强过的数据集传入网络进行训练。Keras 框架支持在线增强，本文将采取在线增强方式对数据集进行扩充，具体操作如下：

（1）将数据集中的图片以随机角度进行旋转。

（2）将数据集中的图片以水平或垂直方向进行小范围的平移。

（3）将数据集中的图片作水平或垂直投影变换。

（4）对数据集中的图片进行随机缩放。

图 4.5 为 CK+数据集数据增强操作示例图。

(a) 原图

(b) 数据增强后的图片

图 4.5 CK+数据集的表情示例

4.2.4 优化方法

深度学习中一般使用梯度下降来求解网络的参数，一个网络的训练完成往往需要花费大量时间和计算资源，但理想的优化方法可以加快网络的训练速度。目前大部分神经网络中都使用随机梯度下降法（Stochastic Gradient Descent, SGD）[68]作为网络训练的优化方法，但是随机梯度下降法在网络训练过程中学习率一直保持不变，需要额外手动设置学习率的变化。为此，Kingma[69]提出了适应性矩估计(Adaptive Moment Estimation, Adam) 优化算法。与 SGD 算法有所区别的是，Adam 算法结合了梯度中的一阶矩和二阶矩，即当前以及过往梯度的均值和平方的均值。可以在网络训练过程中根据训练损失自动更新学习率。Adam 主要有以下优势：

（1）在 Kreas 中实现简单，超参数具有良好的可解释性，基本无需调参。

（2）加快网络收敛速度，计算高效，占用内存较少。

（3）记录了梯度的均值，这个操作可以在每一次进行梯度更新时，更新的梯度与前一个梯度保持较小的差值，有利于保持梯度的更新趋势的平稳，针对不稳定的目标函数的情况也同样适用。

（4）记录了梯度的方差，可以在不同的参数的情况下产生自适应的学习率。

因此，本文在网络中使用的优化方法是 Adam 算法，Adam 算法的迭代过程为：

（1）更新迭代步数：

$$t = t + 1 \tag{4.11}$$

（2）计算t步骤时损失函数$f(\theta)$关于参数θ的梯度g_t：

$$g_t = \nabla_\theta f_t(\theta_{t-1}) \tag{4.12}$$

（3）计算梯度一阶矩估计：

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{4.13}$$

其中，m_t表示t步骤的梯度一阶矩估计，β_1为一阶矩的衰减系数

（4）计算梯度二阶矩估计：

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{4.14}$$

其中，v_t 表示t步骤的梯度二阶矩估计，β_2为二阶矩的衰减系数。

（5）计算修正后的一阶矩估计：

$$\hat{m}_t = m_t / (1-\beta_1^t) \tag{4.15}$$

（6）计算修正后的二阶矩估计：

$$\hat{v}_t = v_t / (1-\beta_2^t) \tag{4.16}$$

由于m_t 和v_t 的初始值均设为 0，在训练初期一阶矩和二阶矩估计都易偏向于 0。因此进行修正计算十分有必要。

（7）更新目标参数θ_t：

$$\theta_t = \theta_{t-1} - \alpha * \hat{m}_t / \left(\sqrt{\hat{v}_t} + \varepsilon\right) \tag{4.17}$$

其中，$\alpha/\left(\sqrt{\hat{v}_t} + \varepsilon\right)$可以表示更新参数$\theta_t$的学习率，而$\hat{m}_t$则可以表示$\theta_t$的梯度。

4.2.5 网络超参数设置

学习率的设置对神经网络的训练至关重要，它决定着网络通过训练其参数能否达到最优解以及何时达到最优解。太大的学习率可能会导致训练的不稳定导致网络参数在最优解附近反复波动无法收敛，太小的学习率又可能会导致训练缓慢或参数陷入局部最优值，所以我们需要把学习率设定为合适的值。本文将初始学习率设为$\alpha_0 = 10^{-4}$，；Adam 优化器的一阶矩的衰减系数β_1设为 0.9，二阶矩的衰减系数β_2设为 0.999, 每一轮参数更新后需要设置一个学习率的衰减值 decay，本文将 decay 设置为 0。在训练过程中，我们随机初始化权重和偏差，并在批大小设置为 32 的情况下执行 300 个历元训练。

4.3 实验结果与分析

4.3.1 DCNN-CBAM 的结构

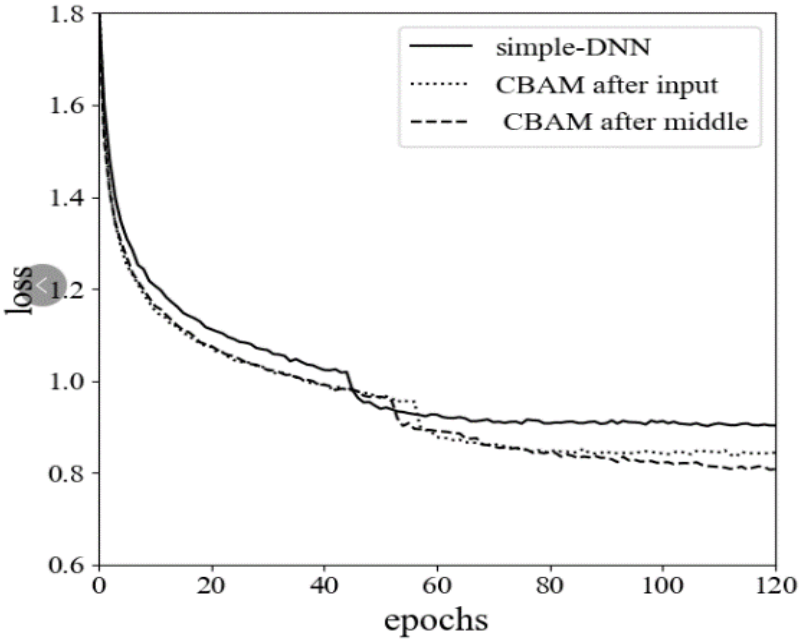


图 4.6 CBAM 位于模型不同位置的 Loss 曲线

在第三章的研究中，我们选择了基于深度可分离卷积的 DCNN 作为我们面部表情识别任务的基础网络。然而，DCNN 的输入数据中缺乏多样性的信息可能会影响网络的最终识别性能。因此，为了强调重要的细节特征并过滤掉一些与表情识别不相关的细节，我们选择在模型中引入了注意力机制。一般的注意力机制仅从单一的空间或通道维度计算特征图的注意力，它们都忽视了另一个维度对特征图的影响，所以我们在 DCNN 中引入了一种融合了空间（spatial）和通道（channel）的注意力机制：卷积块注意力模块（CBAM）。将 CBAM 插入到网络中，它将根据上一层输入的特征映射，按顺序从通道和空间计算输入其通道注意力图和空间注意力图，然后得到最终的注意力图，最后再将初始的特征映射与计算得出的注意力图相乘得到精细化后的输出特征。

在 DCNN 中的不同深度的可分离卷积层后都插入 CBAM 块将不可避免地增加网络架构和训练过程的复杂性。虽然 CBAM 是一个轻量级模块，但过度使用将破坏其轻量级特性。所以我们尝试在网络的不同位置插入 CBAM，为了不破坏网络的完整性，我们尝试将 CBAM 插入到输入模块或中间模块后。通过在 Fer2013 数据集上进行迭代 120 轮，对比原始网络模型的收敛损失曲线，可以看出在中间模块和输出模块之间插入 CBAM 的效果最好，且模型还有继续收敛的趋势，如图 4.6 所示。因此，决定将插入 CBAM 至 DCNN 的中间模块和输出模块之间，以提高网络的特征表达能力。插入 CBAM 后的网络结构如图 4.7，我们将引入注意力机制后的 DCNN 网络称为 DCNN-CBAM。

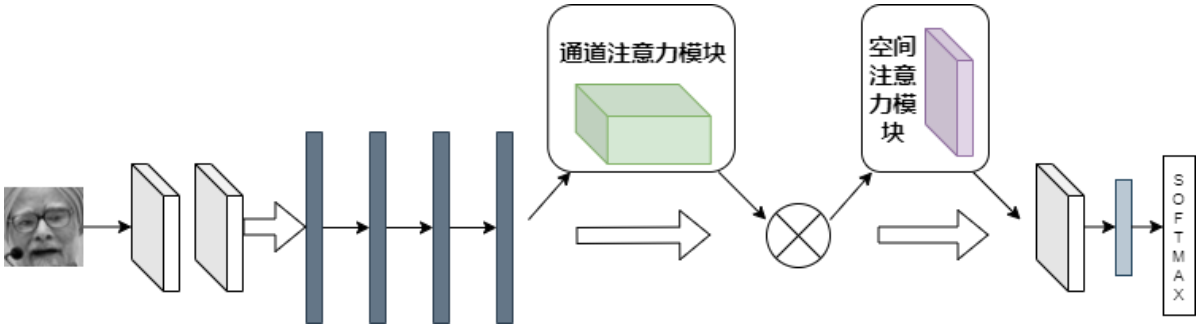


图 4.7 DCNN-CBAM 的网络结构

4.3.2 实验结果

为了验证引入注意力机制后的 DCNN-CBAM 网络性能，我们分别在 Fer2013 数据集和 CK+数据集上进行了实验验证，并与一些现有的模型进行了比较，准确率对比图如表 4.3。

表 4.3 不同模型的准确率对比

数据集	模型	准确率

Fer2013	**DCNN-CBAM**	**70.5%**
	DCNN	69.3%
	VGGNET	67.2%
	DAM-CNN[60]	66.2%
	CNN-Ensemble[61]	65.1%
CK+	**DCNN-CBAM**	**95.8%**
	PHOG+LBP[70]	94.6%
	SFPL[71]	94.7%
	DTAGN[72]	96.7%
	Ali-Net[73]	93.2%

首先在 Fer2013 数据集上，基于注意力机制的 DCNN-CBAM 模型的识别准确率较初始的 DCNN 模型提高了 1.2%，也高于比其它四个基于深度学习的面部表情识别模型。表中的其余模型分别为：自动定位表情图像中与表情相关的区域，并为 FER 提供稳健的图像表示的 DAM-CNN，由几个不同的单独训练的结构化子网组成的 CNN-Ensemble 模型。为了验证 DCNN-CBAM 模型的优越性和泛化能力，我们在 CK+数据集上进行实验验证，除了 DTAGN 和 Ali-Net 模型是基于深度学习的以外，其余的模型都是传统的面部表情识别方法。从表中，我们可以清楚地看到，所提出的 DCNN-CBAM 模型的性能优于许多其他方法。与传统方法相比，DCNN-CBAM 在不需要手动进行特征提取的情况下，通过基于深度学习的网络进行自动的特征提取，使其适用于更复杂的任务。与基于深度学习的方法相比，DCNN-CBAM 由于在模型中引入了注意力机制，能够区分面部表情的相关重要区域，这使得 DCNN-CBAM 在面部表情识别任务中更加有效。在所有比较的模型中，DTGAN 模型在识别准确率上优于 DCNN-CBAM。这是由于 DTGAN 模型在特征提取时，提取了一种以上的特征用来表示表情，而 DCNN-CBAM 只通过提取的 CNN 特征来表示图像。在图像分类相关任务中，多个不同特征融合往往比使用单一类型特征更加有效。然而，在 DTGAN 文献中，当图像仅由单一类型的特征表示时，例如仅仅由外观特征表示的 DTAN 模型和仅有几何特征表示的 DTGN 模型，DCNN-CBAM 模型仍然具有更好的性能。此外在 DTAGN 的工作中，输入的都是视频图像序列，这可以使模型能够学习到关于表情变化的时间或运动信息，这对面部表情识别任务是有效果的。实验结果表明在 DCNN 网络中引入 CBAM 能够有效地提高网络的特征表达能力和识别率。

表 4.4 DCNN-CBAM 在 Fer2013 上的混淆矩阵

	预测值

真实值	愤怒	厌恶	恐惧	开心	伤心	惊讶	平静
愤怒	0.64	0.01	0.07	0.03	0.11	0.02	0.12
厌恶	0.18	0.72	0.03		0.03		0.04
恐惧	0.10		0.52	0.03	0.17	0.07	0.10
开心	0.02		0.01	0.89	0.01	0.02	0.04
伤心	0.07		0.10	0.03	0.62		0.17
惊讶	0.03		0.07	0.05	0.02	0.80	0.02
平静	0.04		0.04	0.07	0.11	0.01	0.73

表 4.5 DCNN-CBAM 在 CK+上的混淆矩阵

真实值	预测值						
	愤怒	厌恶	恐惧	开心	伤心	惊讶	蔑视
愤怒	0.90	0.04	0.02		0.02	0.03	
厌恶	0.01	0.98				0.01	
恐惧	0.04		0.94	0.03			
开心				0.99			0.01
伤心	0.02				0.89	0.01	0.08
惊讶						1.00	
蔑视	0.04		0.06		0.03	0.02	0.85

表 4.4 和表 4.5 分别显示了 DCNN-CBAM 模型在 Fer2013 数据集和 CK+数据集上的混淆矩阵。可以看到在 CK+数据集上各个表情分类的准确率都远远高于 Fer2013 数据集上的准确率，这是由于 Fer2013 数据集是从网络上爬取的表情图片，数据集中存在许多有遮挡以及非正面甚至非表情的图像，这些干扰图片导致网络无法提取到可识别特征；而 CK+数据集则是在实验室中对实验对象进行视频录制，每个视频序列都是由中性表情向其他表情的转变，且每个序列都是基于面部编码系统进行的分类，其中的静态表情图片都是提取的每个视频序列的最后一至三帧，表情具有代表性和普适性，网络可以充分提取不同的表情分类的特征。

4.4 本章小结

本章主要在 DCNN 模型中添加了注意力机制。由于 DCNN 的输入数据集中缺乏多样性和易于分辨的信息，这样可能会影响 DCNN 作为特征提取网络的性能，而注意力机制可以使神经网络忽略图片中无关特征而专注于有效信息。因此，在 DCNN 模型的基础上添加注意力

机制，可以提高模型的特征表达能力，从而提高模型的性能。首先，介绍了目前在计算机视觉中常用的注意力机制的类型，并选择了将空间和通道注意力结合的卷积块注意力模块（CBAM）作为 DCNN 中添加的注意力机制，接着介绍了网络的具体训练情况及超参数设置，最后在确定了 DCNN-CBAM 网络的结构后，我们在 fer2013 数据集和 CK+数据集对模型性能进行了验证评估。从结果上看，相比于以往算法以及上文提出的 DCNN 模型，基于注意力机制的 DCNN-CBAM 算法性能得到有效地提升。

第五章 引入 Mish 和 AM-Softmax 函数的 DCNN-CBAM

第四章介绍了基于注意力机制的 DCNN-CBAM 模型，并通过实验对比证明了该模型优越的鲁棒性。但是有证据表明[74]，在不改变网络结构的情况下，网络中的激活函数和损失函数的选择也会对模型的特征表达能力产生影响。鉴于此，本章决定将研究重点放在了 DCNN-CBAM 模型中的激活函数和损失函数的选择上。首先通过对不同激活函数的数学原理进行阐述和分析，并最终选择 Mish 激活函数替代 DCNN-CBAM 中的原函数，从而有效解决训练过程中产生神经元坏死的情况；接着为了解决传统的 Softmax 损失函数无法解决在面部表情中训练数据存在不同类别的表情差异较小的情况，选择了一种改进的损失函数--相加边际 Softmax（AM-Softmax）来最大化类间的差异；然后，在 Fer2013 数据集和 CK+数据集上验证引入 Mish 激活函数和 AM-Softmax 损失函数的 DCNN-CBAM 网络性能；最后将训练完成的模型应用于面部表情识别系统。

5.1 激活函数

在神经网络的模型搭建中，如果不在隐藏层中引入激活函数，那么某一隐藏层中的输出与上一隐藏层的输入都是线性关系，所以无论如何加深网络的深度，最终得到的输出都与初始输入的正比例相关，在这种情况下会导致网络的特征提取能力极其有限。因此，就需要在神经网络的隐藏层中引入一个非线性函数作为其激励函数，以此来提高神经网络的特征表达能力[75]。

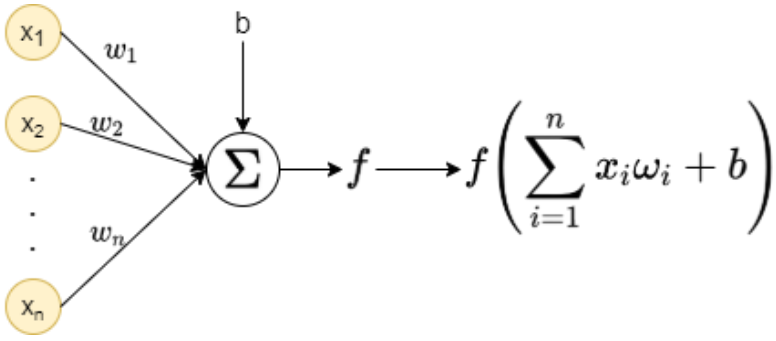


图 5.1 激活函数作用图

激励函数作用如图 5.1，其中输入神经元为x_1至x_n，对应权重为w_1至w_n，偏置为b。激活函数应用于进行线性组合的输入信号，最终输出$f(\sum_{i=1}^{n} x_i w_i + b)$。

在神经网络研究的初始阶段，普遍使用 Sigmoid 或者 Tanh 作为网络的激活函数。随着神经网络的发展，一般网络中都会使用 ReLU 激活函数以及对其进行改进的 PReLU、Leaky-

ReLU（LReLU）和 ELU 函数替代 Sigmoid 和 Tanh。目前在神经网络中使用何种激活函数效果最优仍没有定论，下面我们通过对目前常用的激活函数的数学原理进行分析，以此选择最为合适的激活函数。

5.2 不同激活函数的对比

5.2.1 Sigmoid 激活函数和 Tanh 激活函数

作为在神经网络中最初始的激活函数，Sigmoid 激活函数的功能就是将连续的输入压缩至零至一之间，最后再进行输出。Sigmoid 激活函数的表达式如式（5.1）。

$$Sigmoid(x) = \frac{1}{1+e^{-x}} \in (0,1) \tag{5.1}$$

Sigmoid 函数的图像如图 5.2 所示，由于其固有的缺点，Sigmoid 函数逐渐被弃用。例如，当神经网络的初始权值为上限为 1 下限为 0 的随机值，当反向传播时，梯度值在传过多层时很有可能接近于 0，发生梯度消失的情况，而当初始权值大于 1 时，则可能发生梯度爆炸。经过 Sigmoid 激活后的输出的值是非零均值，这会导致神经网络通过反向传播更新权重参数时只会单向进行更新，再加上表达式中幂函数的存在，最终大大减缓了网络的收敛速度。

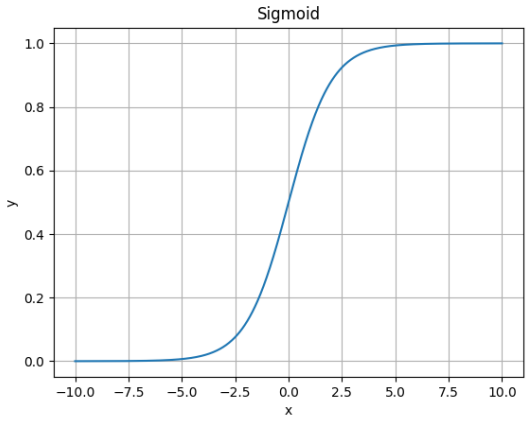


图 5.2 Sigmoid 函数图像

Tanh 激活函数如式（5.2），Sigmoid 函数的输出不是零均值的问题在 Tanh 激活函数中得到了有效解决，但是梯度消失这个问题仍然不可避免，且由于其表达式中使用了更加复杂的幂运算，导致其计算量十分巨大。Tanh 函数的图像如图 5.3 所示。

$$Tanh(x) = \frac{1-e^{-2x}}{1+e^{-2x}} = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \in (-1,1) \tag{5.2}$$

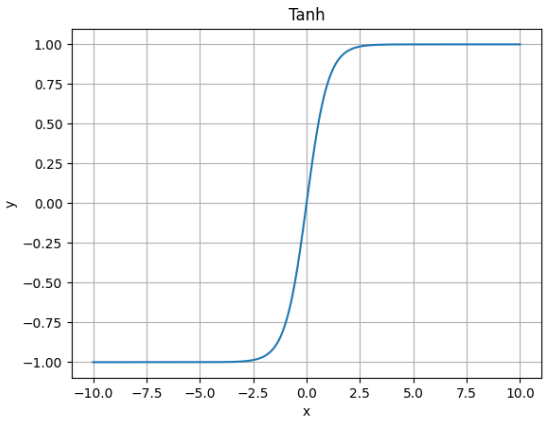


图 5.3 Tanh 函数图像

5.2.2 ReLU 激活函数

非饱和线性修正单元（Rectified Linear Unit, ReLU）激活函数目前普遍应用于卷积神经网络之中。ReLU 函数的表达式为式（5.3）。

$$ReLU(x) = \begin{cases} x & if\ x > 0 \\ 0 & if\ x \leq 0 \end{cases} \tag{5.3}$$

从表达式可以看出 ReLU 函数的本质就是常数与线性函数结合，对小于零的输出值进行归零操作使其为一个常值，输出正值则保持不变进行线性输出。其函数图像如图 5.4 所示。由图可知 ReLU 激活函数的梯度始终在正区间保持为一，这就可以完全避免 Sigmoid 和 Tanh 激活函数存在的梯度消失的问题；此外，在 ReLU 激活函数中不存在幂级运算，仅仅需要判断输入与零的大小关系，计算速度大大加快。但是，ReLU 激活函数也存在一些问题。首先，ReLU 函数的输出值也是非零均值，结果会产生偏移现象。其次，在训练过程中，负值区域的一些神经元可能会坏死，永远不会被激活，导致对应的参数无法进行更新。

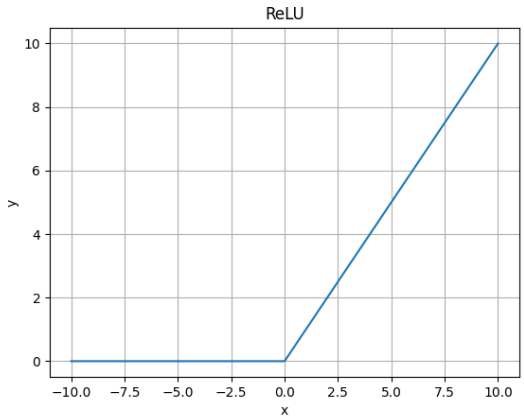


图 5.4 ReLU 函数图像

5.2.3 改进的 ReLU 激活函数

LReLU 和 PReLU 函数十分类似，它们在 ReLU 函数的基础上，为了防止 ReLU 在负值的情况下梯度为 0，为负值赋予一个非零的系数。如式（5.4），LReLU 和 PReLU 函数的区别在于α的取值，其中 LReLU 将α赋值为 0.01，而 PReLU 中的α不是预先设置的，是在训练过程中由反向传播算法自适应学习而得。在 PReLU 中，当α为 0 时 PReLU 等同于 ReLU 函数，为 0.01 时则等效于 LReLU 函数，LReLU/PReLU 函数图像如图 5.5，图中α的值为 0.01。

$$LReLU(x) = \begin{cases} x & if\ x > 0 \\ \alpha x & if\ x \le 0 \end{cases} \tag{5.4}$$

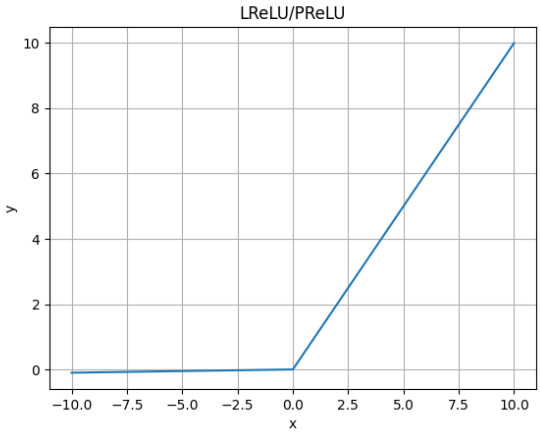


图 5.5 LReLU/PReLU 函数图像

ELU 激活函数是 ReLU 和 Sigmoid 函数的结合，表达式如式（5.5）。式中，α是一个表示 ELU 负值部分梯度的可变参数。ELU 函数图像如图 5.6 所示，我们将α设置为 1，ELU 函数解决了 ReLU 存在的问题，首先，由于负值不再为 0，所以 ELU 消除了负值时神经元可能坏死的情况，其次，ELU 激活函数输出的平均值近似接近于 0，结果不会产生偏移。类似于 LReLU，虽然在理论上 ELU 损失函数的性能表现是优于 ReLU 激活函数的，但在神经网络的应用中这两者的性能表现并没有较为明显的差异。

$$ELU(x) = \begin{cases} x & if\ x > 0 \\ \alpha(exp(x) - 1) & if\ x \le 0 \end{cases} \tag{5.5}$$

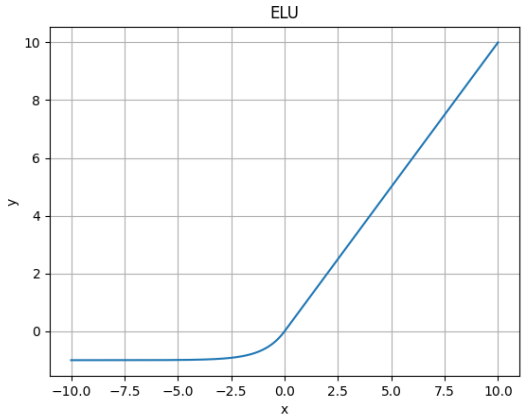


图 5.6 ELU 函数图像

5.2.4 引入的 Mish 激活函数

深度学习中对激活函数的进一步研究从未止步，ReLU 函数还是在被广泛使用，但是这种情况可能会被 Mish 激活函数[76]所改变，Mish 是 Diganta Misra 提出的新的深度学习激活函数，该函数在 70 多个深度学习任务上进行了测试，包括但不限于图像分类、图像分割和图像生成，并与其他 15 个激活函数进行了对比，无论是训练稳定性还是准确率都优于其他激活函数。Mish 激活函数的公式如式（5.6）。

$$Mish(s) = x * tanh\ (\ln\ (1 + e^{x}) \tag{5.6}$$

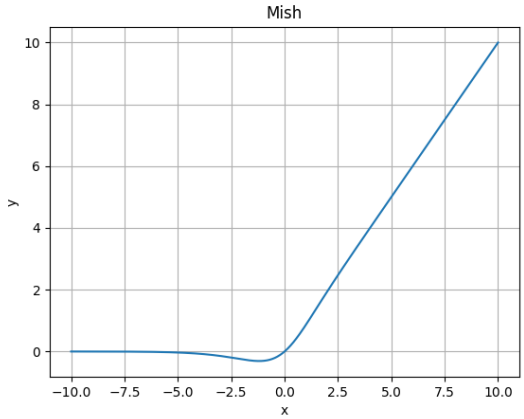


图 5.7 Mish 函数图像

Mish 激活函数无上边界(即正值可以达到任意极大值)，这样可以避免容易引起训练速度迅速下降的梯度饱和，负值不是像 ReLU 中那样的硬零边界，可以增强正则化效果。Mish 激活函数并不是在定义域上单调递增或单调递减的，这种性质允许输出保持小的负值，避免了 ReLU 无负值，神经元坏死的情况。如图 5.7 所示，Mish 损失函数是一条连续的光滑曲线，具有良好的泛化能力，并能对最终结果进行合理优化。

Mish 函数相较于 ReLU 函数，在表达式上增加了些许的复杂性，但是为了保证网络训练

的稳定和保持较高的准确率来说，Mish 激活函数的选择优先级是高于 ReLU 激活函数，所以我们选择使用 Mish 激活函数替代原来 DCNN-CBAM 中的 ReLU 激活函数。

5.3 引入的 AM-Softmax 损失函数

目前在图像分类中使用的损失函数大多是 Softmax 损失函数，但是传统的 Softmax 损失函数无法解决在面部表情中训练数据存在同类表情差异较大，不同类别的表情差异较小的情况。因此，本文引入了一种改进的损失函数--相加边际 Softmax（Additive Margin Softmax, AM-Softmax）来最小化类内差异的同时，对类间差异进行最大化处理。AM-softmax 是在 A-Softmax[77]基础上提出并加以改进。为了更好地理解的 AM-Softmax，需要简要回顾一下 Softmax 和 A-Softmax。Softmax 损失函数的由式（5.7）给出。

$$L_S = -\frac{1}{n}\sum_{i=1}^{n} log\frac{e^{\|W_{y_i}\|\|f_i\|\cos(\theta_{y_i})}}{\sum_{j=1}^{c} e^{\|W_j\|\|f_i\|\cos(\theta_j)}} \tag{5.7}$$

其中f是最后一个完全连接层的输入(f_i表示第i个样本)，W_j是最后一个完全连接层的第j列。

而在 A-Softmax 损失中，为了简化计算，将权重向量W_i进行了归一化，并将目标逻辑回归从$\|f_i\|\cos(\theta_{y_i})$推广到$\|f_i\|\psi cos(\theta_{y_i})$，A-Softmax 的损失函数为式（5.8）。

$$L_{AS} = -\frac{1}{n}\sum_{i=1}^{n} log\frac{e^{\|f_i\|\psi(\theta_{y_i})}}{e^{\|f_i\|\psi(\theta_{y_i})}+\sum_{j=1,j\neq y_i}^{c} e^{\|f_i\|\cos(\theta_j)}} \tag{5.8}$$

其中$\psi(\theta)$通常是定义为式（5.9）。

$$\psi(\theta) = \frac{(-1)^k\cos(m\theta)-2k+\lambda\cos(\theta)}{1+\lambda}\ \theta\in\left[\frac{k\pi}{m},\frac{(k-1)\pi}{m}\right] \tag{5.9}$$

其中，m通常是大于 1 的整数，而λ是一个超参数，用于控制分类边界。

AM-softmax 与 A-Softmax 类似，都引入了参数m来调整特征间的距离，并将 A-Softmax 中的$\psi(\theta)$改进为$\psi(\theta)=\cos\theta-m$，并且将 A-Softmax 中的倍乘法改为加法，用于增加类间距离并缩小类内距离。在实现过程中，对特征f和权重W_i进行归一化后的输入实际上是$x=\cos\theta_{y_i}$，因此在前向传播中只需要计算$\psi(x)=x-m$。最终的损失函数为式（5.10）。

$$L_{AMS} = -\frac{1}{n}\sum_{i=1}^{n} log\frac{e^{s\cdot\cos(\theta_{y_i}-m)}}{e^{s\cdot\cos(\theta_{y_i}-m)}+\sum_{j=1,j\neq y_i}^{c} e^{s\cos(\theta_j)}} \tag{5.10}$$

式中的s和m是两个超参数，通常将s的值设置为 30，m的值设置为 0.35。图 5.8 展示了当类别 1 和类别 2 出现类间间距过小的情况时，Softmax 损失函数和 AM-Softmax 损失函数分类可能的结果。当使用 Softmax 损失函数时，类别 1 与类别 2 映射到特征空间有一些区域

发生了重叠的现象，而 AM-Softmax 由于附加的幅度存在最大化了类间间距，可以很好的区分类别 1 和类别 2。

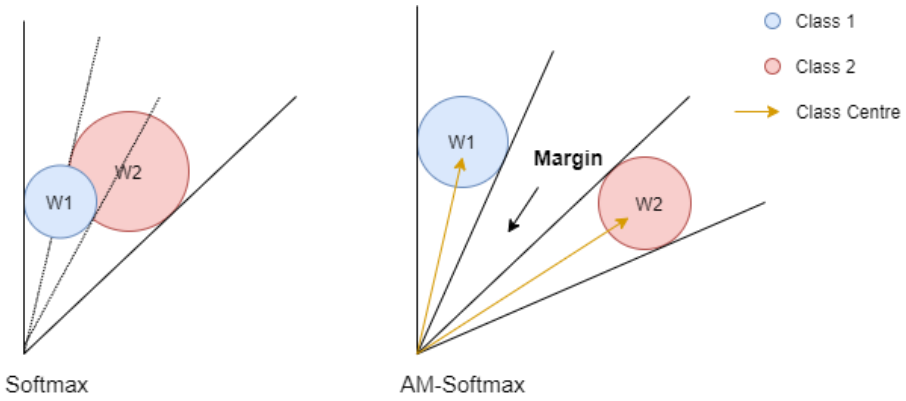


图 5.8 Softmax 和 AM-Softmax 函数的分类结果

5.4 实验结果与分析

最终我们将引入 Mish 和 AM-Softmax 函数的 DCNN-CBAM 模型命名为 MADCNN-CBAM，MADCNN-CBAM 在 Fer2013 数据集和 CK+数据集上的和其他模型的准确率对比如表 5.2 所示。

表 5.1 本文模型与现有常用算法的准确率对比

数据集	模型	准确率
Fer2013	**MADCNN-CBAM**	**71.5%**
	DCNN-CBAM	70.5%
	VGGNET	67.2%
	DAM-CNN[60]	66.2%
	CNN-Ensemble[61]	65.1%
CK+	**MADCNN-CBAM**	**99.2%**
	DCNN-CBAM	95.8%
	PHOG+LBP[70]	94.6%
	SFPL[71]	94.7%
	DTAGN[72]	96.7%

MADCNN-CBAM 模型在 Fer2013 数据集和 CK+数据集上的混淆矩阵如表 5.3，5.4。

表 5.2 MADCNN-CBAM 在 Fer2013 数据集上的混淆矩阵

	预测值

真实值	愤怒	厌恶	恐惧	开心	伤心	惊讶	平静
愤怒	0.65	0.01	0.07	0.03	0.12	0.02	0.10
厌恶	0.12	0.82			0.02		0.03
恐惧	0.11		0.54	0.02	0.16	0.08	0.09
开心	0.02		0.02	0.90	0.02	0.02	0.03
伤心	0.09		0.10	0.03	0.61	0.02	0.16
惊讶	0.02		0.07	0.04	0.01	0.83	0.02
平静	0.04		0.05	0.04	0.12	0.01	0.74

表 5.3 MADCNN-CBAM 在 CK+数据集上的混淆矩阵

真实值	预测值						
	愤怒	厌恶	恐惧	开心	伤心	惊讶	蔑视
愤怒	0.99	0.01					
厌恶	0.01	0.98				0.01	
恐惧			0.98	0.02			
开心				1.00			
伤心					1.00		
惊讶						1.00	
蔑视			0.01				0.99

图 5.9，5.10 出了针对每种情绪的 DCNN-CBAM 模型和 MADCNN-CBAM 模型的面部表情估计。直观的对比了引入了 Mish 激活函数和 AM-Softmax 损失函数后的 DCNN-CBAM 模型在两个数据集上较初始模型各个表情分类的准确率的。

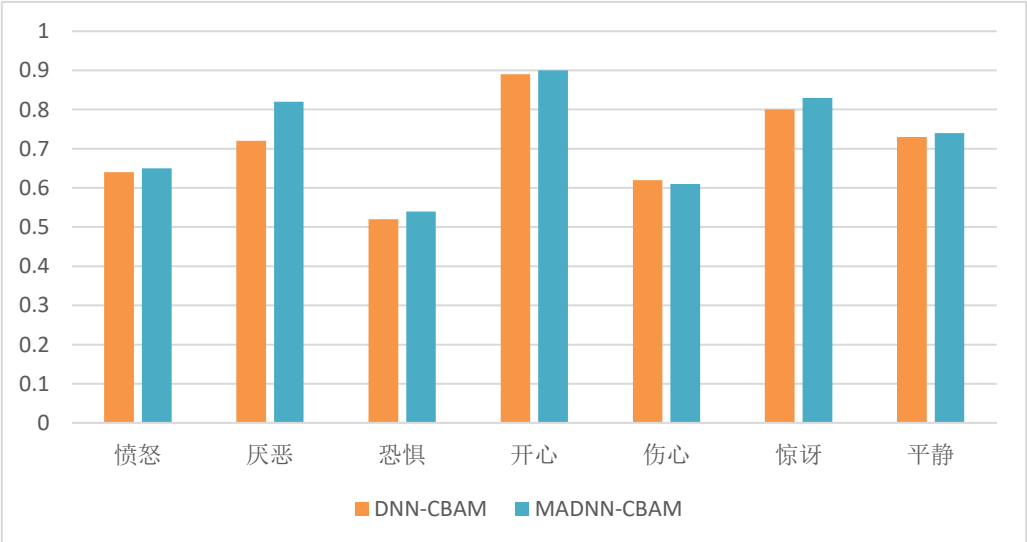


图 5.9 两个模型在 Fer2013 上的情绪估计

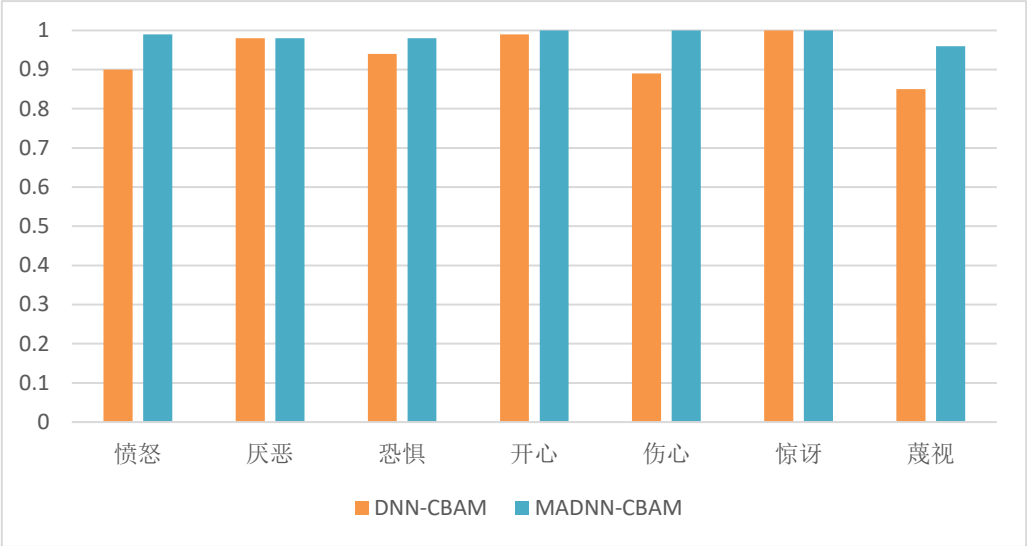


图 5.10 两个模型在 CK+上的情绪估计

由以上实验结果可知，引入 Mish 激活函数和 AM-Softmax 损失函数的 DCNN-CBAM 在面部表情识别任务上比第四章所提出的 DCNN-CBAM 模型的准确率更高，特征表达能力更强。

5.5 面部表情识别应用

5.5.1 系统框图

本文的目的是实现面部表情的识别，这个任务主要包含两部分，首先是网络模型的设计和训练，然后是应用训练好的模型进行面部表情识别。上文已经完成了网络模型的改进与训练，最终我们将训练完成的模型应用于面部表情识别系统，系统的流程图如图 5.11 所示。

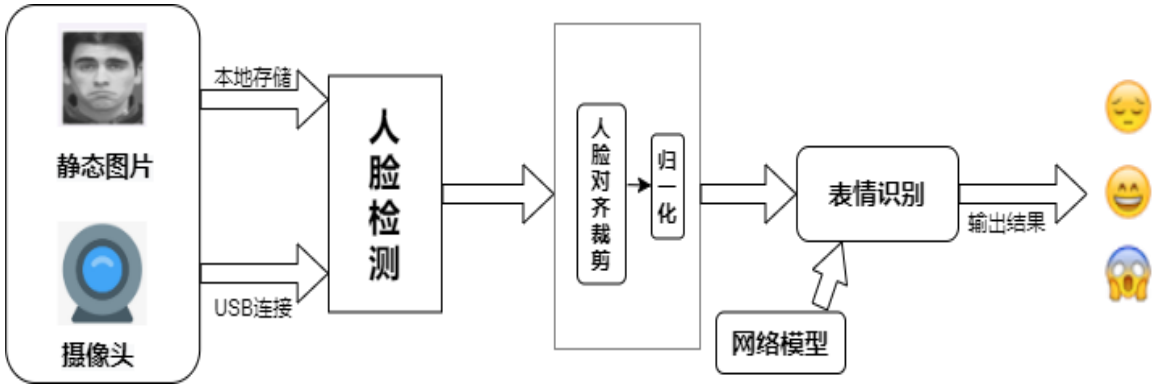


图 5.11 面部表情识别系统的流程框图

系统中的人脸检测器是使用 OpenCV 自带的基于 Haar 特征的级联分类器。Haar 级联算法是 OpenCV 中最流行的目标检测算法，尽管许多算法（SSDs、更快的 R-CNN、YOLO 等等）比 Haar 级联算法精确度上更加精确，但是识别速度远远不及 Haar 级联算法快速。

5.5.2 面部表情识别实验效果图

首先为了对真实人脸表情进行判别测试，我们在教室环境下进行实时面部表情识别，程序会调用电脑外接或自带的摄像头首先对拍摄到的画面进行人脸检测，如果检测到人脸就会框选出人脸并实时的展示表情识别结果，如图 5.12 所示，此时的面部表情以人的认知判断毫无疑问为平静，实时面部表情检测出的结果也为平静，图像展示窗口显示框选出人脸并给出表情预测为 calm，右侧的表情预测概率直方图窗口则显示了当前面部表情图像在 7 个基本表情上的预测概率图，平静的预测概率高达了 94.92%。图 5.13 展示了开心、愤怒、惊讶、伤心、厌恶和恐惧这些表情的实时识别结果。

图 5.12 平静的实时识别效果图

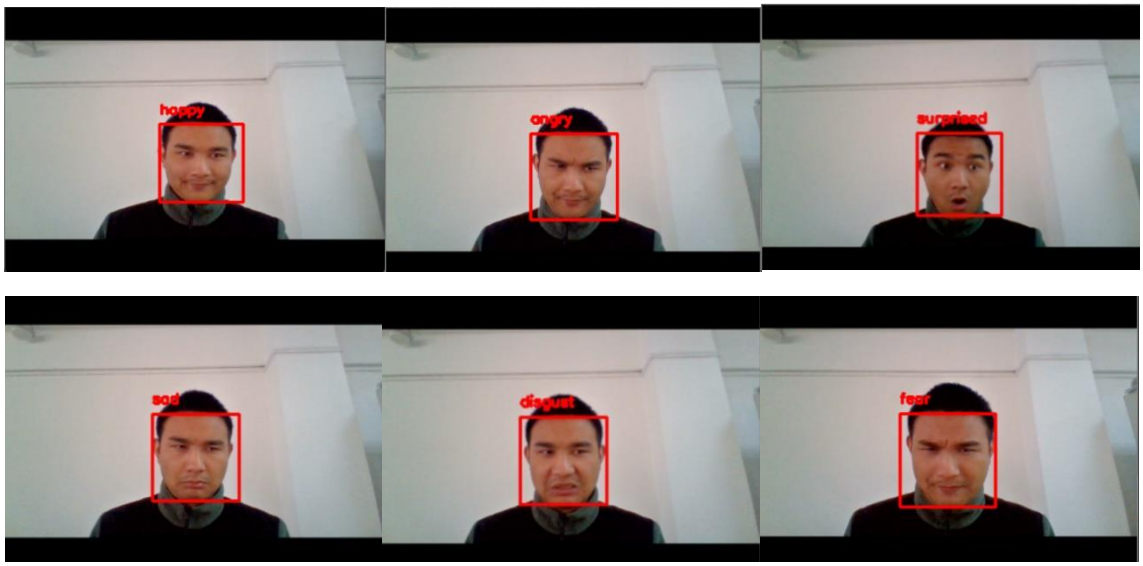


图 5.13 在教室环境下的实时识别效果图

此外为了研究在不同的场景下对实际测试产生的影响，我们还额外选取了实验室以及宿舍进行了面部表情识别测试，具体的实验结果如图 5.14 和图 5.15 所示。由图 5.14 可知，即使是在有他人干扰的复杂环境下对面部进行不同程度的遮挡，系统仍然可以较为准确的检测出人脸并对其进行准确的面部表情识别；由图 5.15 可知，即使在光照情况较为复杂的宿舍环境下，系统仍然可以较为准确的检测识别出人脸并对其进行准确的面部表情识别。此外在宿舍环境下我们对佩戴眼镜是否会对识别效果产生影响进行了研究，由结果可知，在佩戴眼镜情况下识别出的表情都是准确的，符合人类的认知。

最终为了研究人的肤色是否会对识别结果产生影响，我们从网上选取了一些白人的面部图像进行面部表情识别，图 5.16 和 5.17 展示了识别结果。因为有几种面部表情有许多共同的特征，比如惊讶和恐惧，愤怒和厌恶，所以结果可能会有一些偏差。然而，它们中的大多数都是准确的。

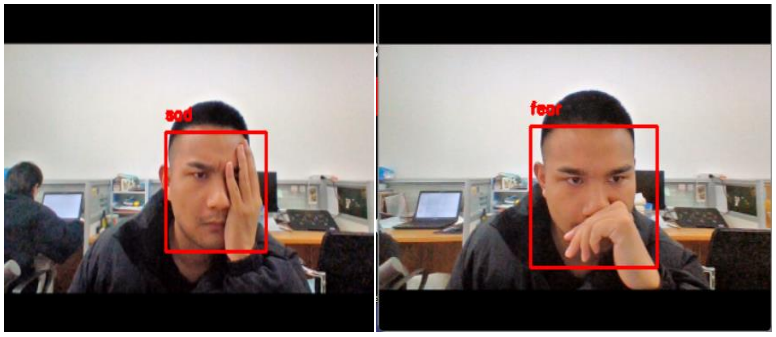


图 5.14 在实验室环境下的实时识别效果图

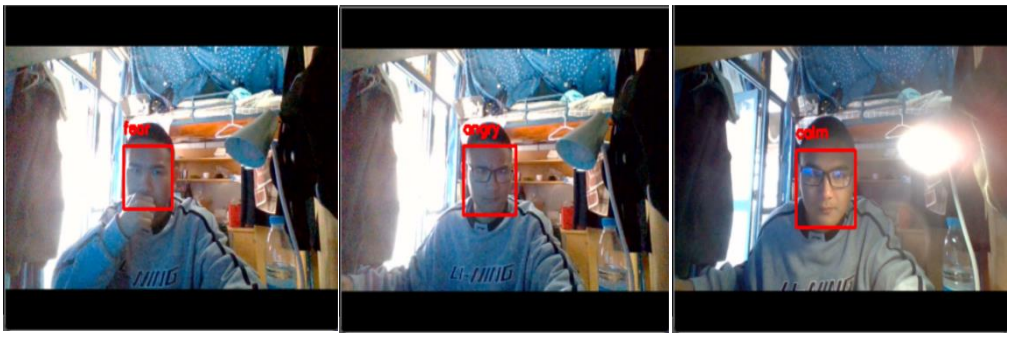


图 5.15 在宿舍环境下的实时识别效果图

图 5.16 对开心、恐惧图片的识别结果图

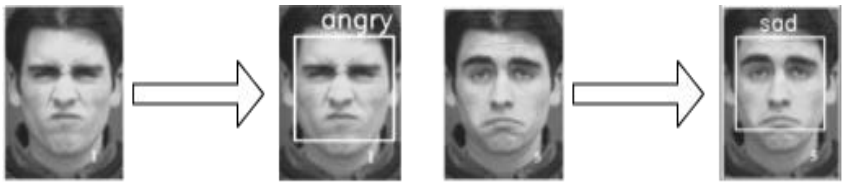


图 5.17 对愤怒、悲伤图片的识别结果图

5.6 本章小结

本章对网络模型中的激活函数和损失函数进行了研究，重点介绍了在 DCNN-CBAM 中引入的 Mish 激活函数和 AM-Softmax 损失函数。Mish 损失函数可以解决网络在训练中神经元坏死的情况从而提高网络的稳定性和准确率，而 AM-Softmax 损失函数则可以解决传统的 Softmax 损失函数无法解决的在面部表情中训练数据存在不同类别的表情类间差异较小的情况。通过实验验证了改进后的模型在面部表情识别任务上的性能得到一定程度的提升，最终我们将训练完成的模型应用于表情识别系统，取得了较好的表情识别效果。

第六章 总结与展望

6.1 本文总结

在众多的非语言成分中，面部表情承载着情感的意义，是人际交流的主要信息渠道之一。所以面部表情识别是当前的研究热点，可应用于人机交互、情感计算等计算机视觉领域。

本文以当前基于深度学习算法进行面部表情识别的论文为基础，针对传统的面部表情识别方法使用人工提取特征存在人为因素的干扰，以至于训练完成的分类器不能有效地解释表情信息，最终导致模型泛化能力不足等问题，在卷积神经网络的基础上进行改进。本文具体的工作如下：

（1）针对研究课题，阐述了面部表情识别的研究现状，对基于传统的机器学习和基于深度学习的面部表情识别算法进行了详细的对比分析，并详细的介绍了卷积神经网络的结构以及面部表情识别的各个步骤。

（2）本文针对面部表情识别任务，提出两个基础模型。第一个是改进的经典卷积神经网络，我们针对 VGGNet 的网络结构采取了一些改进措施，第二个是基于深度可分离卷积和残差块的 DCNN 网络。在 Fer2013 数据集上通过准确率和混淆矩阵指标对这两个模型进行了对比实验，最终选择基于深度可分离卷积的 DCNN 作为我们的特征提取网络。

（3）本文在 DCNN 模型的中引入了注意力机制，提出了一种基于注意力机制的 DCNN-CBAM 模型。在 DCNN 中引入注意力机制可以使神经网络忽略图片中无关特征细节而专注于有效信息，CBAM 依次沿着通道维度和空间维度对隐藏层输入的特征映射计算注意力图，在不会过多的增加模型的参数和计算量的前提下，可以显著的提高模型的特征表达能力。实验证明，基于注意力机制的 DCNN-CBAM 模型的性能显著。

（4）本文在基于注意力机制的 DCNN-CBAM 模型基础上，对模型的激活函数和损失函数进行了进一步研究，将 Mish 激活函数和 AM-Softmax 损失函数引入到 DCNN-CBAM 模型。使用 Mish 函数可以大幅降低网络在使用 ReLU 函数时发生的神经元坏死的概率，而使用 AM-Softmax 损失函数则可以最大化面部表情数据的类间间距。在数据集上的实验表明，引入 Mish 和 AM-Softmax 函数的 DCNN-CBAM 模型具有更好的特征提取能力，可以进一步提高表情识别率。

6.2 未来展望

如今，面部表情识别技术发展迅速，由于时间和技术的限制，本文仍然存在许多的局限性，未来的工作还可以对此研究方法做进一步的改进和优化，使其具有普适性和有效性。具体如下：

（1）在深度学习中，模型并不是对实验结果产生影响的唯一因素，数据集的质量和规模也很重要，目前针对面部表情识别任务的数据集都有其局限性，而且国内关于此类的公开数据集较少，因此在后面的研究中可以自己构建并设计一个公开的数据集用于面部表情识别任务。

（2）面部表情识别应用场景十分广泛，在后续的研究中，我们可以将其应用于一个更加具体的场景之中，比如课堂的专注度识别，因为在高校中采集学生在课堂中不同的表情比较简便，后续再对其进行标注分类制作成数据集进行训练分类识别，使其更加具有研究的价值。

（3）本文第四章引入的注意力机制虽然对网络的准确率有所提升，但是其结构还有改进的空间，接下来的工作中，可以进一步研究空间和通道注意力机制的关系从而更好地将二者结合，形成更加高效的混合注意力模块，让网络更好地提取特征的空间和通道信息。

参考文献

[1] Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10781-10790.

[2] 田萱, 王亮, 丁琪. 基于深度学习的图像语义分割方法综述[J]. 软件学报, 2019, 30(2): 440-468.

[3] Masi I, Wu Y, Hassner T, et al. Deep face recognition: A survey[C]//2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI). IEEE, 2018: 471-478.

[4] Xiao B, Wu H, Wei Y. Simple baselines for human pose estimation and tracking[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 466-481.

[5] Sariyanidi E, Gunes H, Cavallaro A. Automatic analysis of facial affect: A survey of registration, representation, and recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2014, 37(6): 1113-1133.

[6] Anagnostopoulos C N, Iliou T, Giannoukos I. Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011[J]. Artificial Intelligence Review, 2015, 43(2): 155-177.

[7] Tian Y L, Kanade T, Cohn J F. Facial expression analysis[M]//Handbook of face recognition. Springer, New York, NY, 2005: 247-275.

[8] Mehrabian A, Ferris S R. Inference of attitudes from nonverbal communication in two channels[J]. Journal of consulting psychology, 1967, 31(3): 248.

[9] Bosch N, D'Mello S K, Baker R S, et al. Detecting student emotions in computer-enabled classrooms[C]//IJCAI. 2016: 4125-4129.

[10] Samara A, Galway L, Bond R, et al. Affective state detection via facial expression analysis within a human–computer interaction context[J]. Journal of Ambient Intelligence and Humanized Computing, 2019, 10(6): 2175-2184.

[11] 张殿业, 程静, 张艺. 情绪对驾驶行为影响研究[J]. 中国安全科学学报, 2018, 28(10): 19-24.

[12] 吴媛媛, 宋玉祥. 中国人口老龄化空间格局演变及其驱动因素[J]. 地理科学, 2020, 40(5): 768-775.

[13] 周萌,程明凤,汪倩倩,陈滢. 人脸识别技术在医疗领域的应用前景及影响因素[J].中国校医,2020,34(10):786-788.

[14] 文贵华, 李辉辉, 李丹扬,等. 基于社区安全的人群甄别视频预警研究[J]. 华南理工大学学报(社会科学版), 2016, 18(4):79-84.

[15] Darwin, Charles. The Expression of the Emotions in Man and Animals[J]. Journal of Nervous & Mental Disease, 1978, 123(1):90.

[16] Ekman P, Friesen W V. Constants across cultures in the face and emotion[J]. Journal of personality and social psychology, 1971, 17(2): 124.

[17] Ekman P , Friesen W V . Facial Action Coding System (FACS): a Technique for the Measurement of Facial Actions[J]. Rivista Di Psichiatria, 1978, 47(2):126-38.

[18] Mase K. Recognition of facial expression from optical flow[J]. IEICE TRANSACTIONS on Information and Systems, 1991, 74(10): 3474-3483.

[19] Zong Y, Zheng W, Huang X, et al. Transductive transfer lda with riesz-based volume lbp for emotion recognition in the wild[C]//Proceedings of the 2015 ACM on International Conference on Multimodal Interaction. 2015: 491-496.

[20] Uddin M Z, Lee J J, Kim T S. An enhanced independent component-based human facial expression recognition from video[J]. IEEE Transactions on Consumer Electronics, 2009, 55(4): 2216-2224.

[21] Abdulrahman M, Gwadabe T R, Abdu F J, et al. Gabor wavelet transform based facial expression recognition using PCA and LBP[C]//2014 22nd Signal Processing and Communications Applications Conference (SIU). IEEE, 2014: 2265-2268.
[22] Tiwari P, Rathod H, Thakkar S, et al. Multimodal emotion recognition using SDA-LDA algorithm in video clips[J]. Journal of Ambient Intelligence and Humanized Computing, 2021: 1-18.
[23] Arora M, Kumar M. AutoFER: PCA and PSO based automatic facial emotion recognition[J]. Multimedia Tools and Applications, 2021, 80(2): 3039-3049.
[24] Zheng W, Zhou X, Zou C, et al. Facial expression recognition using kernel canonical correlation analysis (KCCA)[J]. IEEE transactions on neural networks, 2006, 17(1): 233-238.
[25] 朱晓明,姚明海.基于局部二元模式的人脸表情识别[J].计算机系统应用,2011,20(06):151-154.
[26] Mahmood M, Jalal A, Evans H A. Facial expression recognition in image sequences using 1D transform and gabor wavelet transform[C]//2018 international conference on Applied and Engineering Mathematics (ICAEM). IEEE, 2018: 1-6.
[27] 王思明,梁运华.基于改进 LBP 的人脸面部表情特征提取方法（英文）[J].Journal of Measurement Science and Instrumentation,2019,10(04):342-347.
[28] 李梦然. 贝叶斯网络在人脸表情识别中的应用研究[D]. 西安：陕西科技大学,2021.DOI:10.27290/d.cnki.gxbqc.2021.000095.
[29] Rahul M, Shukla R, Goyal P K, et al. Gabor Filter and ICA-Based Facial Expression Recognition Using Two-Layered Hidden Markov Model[M]//Advances in Computational Intelligence and Communication Technology. Springer, Singapore, 2021: 511-518.
[30] Saeed S, Baber J, Bakhtyar M, et al. Empirical evaluation of svm for facial expression recognition[J]. International Journal of Advanced Computer Science and Applications, 2018, 9(11): 670-673.
[31] Liu X, Cheng X, Lee K. Ga-svm-based facial emotion recognition using facial geometric features[J]. IEEE Sensors Journal, 2020, 21(10): 11532-11542.
[32] Xin M, Wang Y. Research on image classification model based on deep convolution neural network[J]. EURASIP Journal on Image and Video Processing, 2019, 2019(1): 1-11.
[33] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. science, 2006, 313(5786): 504-507.
[34] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25.
[35] Hinton G, LeCun Y, Bengio Y. Deep learning[J]. Nature, 2015, 521(7553): 436-444.
[36] Li W, Li M, Su Z, et al. A deep-learning approach to facial expression recognition with candid images[C]//2015 14th IAPR International Conference on Machine Vision Applications (MVA). IEEE, 2015: 279-282.
[37] Akhand M A H, Roy S, Siddique N, et al. Facial Emotion Recognition Using Transfer Learning in the Deep CNN[J]. Electronics, 2021, 10(9): 1036.
[38] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
[39] 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.
[40] 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.
[41] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708.
[42] Jain D K, Shamsolmoali P, Sehdev P. Extended deep neural network for facial emotion recognition[J]. Pattern Recognition Letters, 2019, 120: 69-74.

[43] Saurav S, Saini R, Singh S. EmNet: a deep integrated convolutional neural network for facial emotion recognition in the wild[J]. Applied Intelligence, 2021: 1-28.
[44] Vanholder H. Efficient inference with tensorrt[C]//GPU Technology Conference. 2016, 1: 2.
[45] Tong X, Sun S, Fu M. Data augmentation and second-order pooling for facial expression recognition[J]. IEEE Access, 2019, 7: 86821-86828.
[46] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.
[47] Sadik R, Anwar S, Reza M L. Autismnet: Recognition of autism spectrum disorder from facial expressions using mobilenet architecture[J]. International Journal, 2021, 10(1): 327-334.
[48] Agrawal A, Mittal N. Using CNN for facial expression recognition: a study of the effects of kernel size and number of filters on accuracy[J]. The Visual Computer, 2020, 36(2): 405-412.
[49] Wang F, Jiang M, Qian C, et al. Residual attention network for image classification[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 3156-3164.
[50] Jaderberg M, Simonyan K, Zisserman A. Spatial transformer networks[J]. Advances in neural information processing systems, 2015, 28: 2017-2025.
[51] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.
[52] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.
[53] Cao W, Feng Z, Zhang D, et al. Facial expression recognition via a CBAM embedded network[J]. Procedia Computer Science, 2020, 174: 463-477.
[54] Wang Y, Li Y, Song Y, et al. The influence of the activation function in a convolution neural network model of facial expression recognition[J]. Applied Sciences, 2020, 10(5): 1897.
[55] Wang F, Cheng J, Liu W, et al. Additive margin softmax for face verification[J]. IEEE Signal Processing Letters, 2018, 25(7): 926-930.
[56] Joulin A, Cissé M, Grangier D, et al. Efficient softmax approximation for gpus[C]//International Conference on Machine Learning. PMLR,2017: 1302-1310.
[57] Rowley H A, Baluja S, Kanade T. Neural network-based face detection[J]. IEEE Transactions on pattern analysis and machine intelligence, 1998, 20(1): 23-38.
[58] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[C]//Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. Ieee, 2001, 1: I-I.
[59] 李政浩. 基于深度注意力网络的人脸表情识别[D]. 重庆：西南大学, 2019.
[60] Xie S, Hu H, Wu Y. Deep multi-path convolutional neural network joint with salient region attention for facial expression recognition[J]. Pattern recognition, 2019, 92: 177-191.
[61] Liu K, Zhang M, Pan Z. Facial expression recognition with CNN ensemble[C]//2016 international conference on cyberworlds (CW). IEEE, 2016: 163-166.
[62] Abadi M, Barham P, Chen J, et al. Tensorflow: A system for large-scale machine learning[C]//12th USENIX symposium on operating systems design and implementation (OSDI 16). 2016: 265-283.
[63] Paszke A, Gross S, Massa F, et al. Pytorch: An imperative style, high-performance deep learning library[J]. arXiv preprint arXiv:1912.01703, 2019.
[64] Gulli A, Pal S. Deep learning with Keras[M]. Packt Publishing Ltd, 2017.
[65] Jia Y, Shelhamer E, Donahue J, et al. Caffe: Convolutional architecture for fast feature embedding[C]//Proceedings of the 22nd ACM international conference on Multimedia. 2014: 675-678.
[66] Al-Rfou R, Alain G, Almahairi A, et al. Theano: A Python framework for fast computation of mathematical expressions[J]. arXiv e-prints, 2016: arXiv: 1605.02688.

[67] Lucey P, Cohn J F, Kanade T, et al. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression[C]//2010 ieee computer society conference on computer vision and pattern recognition-workshops. IEEE, 2010: 94-101.
[68] Bottou L. Stochastic gradient descent tricks[M]//Neural networks: Tricks of the trade. Springer, Berlin, Heidelberg, 2012: 421-436
[69] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
[70] Happy S L, Routray A. Robust facial expression classification using shape and appearance features[C]//2015 Eighth International Conference on Advances in Pattern Recognition (ICAPR). IEEE, 2015: 1-5.
[71] Happy S L, Routray A. Automatic facial expression recognition using features of salient facial patches[J]. IEEE transactions on Affective Computing, 2014, 6(1): 1-12.
[72] Jung H, Lee S, Yim J, et al. Joint fine-tuning in deep neural networks for facial expression recognition[C]//Proceedings of the IEEE international conference on computer vision. 2015: 2983-2991.
[73] Mollahosseini A, Chan D, Mahoor M H. Going deeper in facial expression recognition using deep neural networks[C]//2016 IEEE Winter conference on applications of computer vision (WACV). IEEE, 2016: 1-10.
[74] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//International conference on machine learning. PMLR, 2015: 448-456.
[75] 周飞燕, 金林鹏, 董军. 卷积神经网络研究综述[J]. 计算机学报, 2017, 40(6): 1229-1251.
[76] Misra D. Mish: A self regularized non-monotonic neural activation function[J]. arXiv preprint arXiv:1908.08681, 2019, 4: 2.
[77] Liu W, Wen Y, Yu Z, et al. Sphereface: Deep hypersphere embedding for face recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 212-220.

附录 1 攻读硕士学位期间申请的专利

[1] 张昀，邹翔翔，毛新涛，等. 一种面部表情识别方法，2021111526698.7，2021.12。

附录 2 攻读硕士学位期间参加的科研项目

(1). 国家自然科学基金，基于深度学习的移位 MIMO“鬼”成像方法研究（61871234）

致谢

时光荏苒，与南邮初识于2015年秋季，在2022年夏季与其告别，在这七年的学生生涯中，多得是幸福的回忆。回顾自己的研究生生活，与入学相比，不仅褪去了几分稚气，更重要的是收获了许多知识与人生经验，我对此满怀感激，感谢我的母校。走过这三年的路程，更少不了他人的帮助与鼓励。

首先我要感谢我的家人，我的父亲母亲，每当我产生困惑亦或是遇到生活上的难关，他们一直是我坚强的后盾；我要感谢我的女朋友魏敏惠，与她相识相知五年，每当我遇到生活的不顺心之事，她总是能帮我排忧解难，她给我带来快乐，她是我努力前进的动力。

在这三年读研生活中，给予我最大帮助的就是我的导师们，张昀老师虽然有时候有些许严苛，但是我深知没有她的谆谆教诲与悉心指导，就没有我课题的顺利完成以及找到心仪的工作，同时我也真诚的感谢于舒娟老师对我生活和学习上的深切关心与帮助。在此，我衷心感谢她们对我学习和生活道路的谆谆指导，我会时刻铭记她们的教诲，在以后的工作生活中继续努力前进。

此外，我还要感谢我的同门们，姚成杰、张宇德和刘丹蕾。在生活中我们总是互帮互助，在学习中我们也会互相指导。正是因为你们的存在，让我拥有一个充实的研究生生涯，希望我们友谊天长地久。同时也感谢这三年中师门中的师弟师妹以及师兄师姐们。当然还有我的本科舍友刘庆胜，每当我们遇到学术或者生活上的难关之时，总是能够给予对方无私的帮助，这份友谊难得可贵。

最后，再次感谢这三年遇到的所有人们，你们的帮助为我指明了人生的道路以及努力的方向。希望在未来的生活中我可以继续以努力为标杆，朝着自己的目标前进。