

分类号 _____

密 级 _____

UDC _____

编 号 _____

成都理工大学

硕士学位论文

题名和副题名 基于卷积神经网络的音频场景分类方法研究

作 者 姓 名 _____

指导教师姓名及职称 _____

申请学位级别 学术硕士 专业名称 信息与通信工程

论文提交日期 _____ 论文答辩日期 _____

学位授予单位和日期 成 都 理 工 大 学 (年 月)

答辩委员会主席 _____

评阅人 _____

2019 年 月

分类号_____

学校代码: 10616

UDC _____

密级_____ 学号: _____

成都理工大学硕士学位论文

基于卷积神经网络的音频场景 分类方法研究

指导教师姓名及职称 _____

申请学位级别 _____ 学术硕士 _____ 专业名称 _____ 信息与通信工程 _____

论文提交日期 _____ 论文答辩日期 _____

学位授予单位和日期 _____ 成 都 理 工 大 学 (_____ 年 _____ 月) _____

答辩委员会主席 _____

评阅人 _____

2019 年 月

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得成都理工大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的人员对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：

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成都理工大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权成都理工大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名：

学位论文作者导师签名：

年 月 日

基于卷积神经网络的音频场景分类方法研究

摘 要

音频场景分类是一项通过音频分析使设备能够理解其所处环境的任务,属于计算机听觉场景领域的一个分支。目前该技术已广泛用于智能可穿戴设备、机器人传感、上下文感知服务等应用场景。近年来深度学习领域的发展更是加速了音频场景分类的研究进程。作为深度学习领域中一种重要的模型,卷积神经网络具有很强的学习能力。通过引入卷积神经网络模型作为音频场景分类器,可使分类准确率获得可观的提升,甚至能使机器超过人类水平。为了探究卷积神经网络在音频场景分类领域的适用性并寻找系统性能的提升方法,文中设计了三组系统并进行了实验及比较,主要工作如下:

本文从设计基于梅尔频率倒谱系数和高斯混合模型的基线系统开始,用传统机器学习的方法构造了一个典型的基线系统作为之后系统的对照组。接着研究基于卷积神经网络的音频场景分类系统的原理,探讨将卷积神经网络应用在音频场景分类中的适用性,并设计实现了一个有两层卷积模块的基本系统。训练系统时通过调整滤波器参数以发挥其分类潜力,同时还将训练时间考虑到系统性能评估的要素中去。评估阶段分析基本系统在各类别上的分类准确率并引入混淆矩阵,发现其学习能力相对基线系统更强,但应对不同数据时泛化能力不佳,且没有有效利用到音频文件中的空间信息。

根据基本系统体现出的问题,本文又设计了一个改进系统,从音频处理和网络结构两方面对基本系统进行改进。音频处理方面使用了双耳表示法及谐波-冲击源分离法对原始音频进行处理并提取相应特征,使系统得以利用场景的空间特征,进而使分类准确率得到了可观的提升。网络结构方面尝试借鉴图像识别领域中的 VGGNet 结构,在增加网络深度的同时提升系统灵活性,最终在不同的数据上取得了更好的泛化效果。此外改进系统还使用了集成学习中的 Stacking 方法将多个基于不同特征的独立子模型融合,融合后的系统相比其中的子模型分类性能又有了进一步的提升。

通过实验及比较,最终得出的结论是:在音频场景分类领域中,卷积神经网络相比于传统机器学习方法学习能力更强。在设计卷积神经网络时应注意网络的灵活性,将提升系统性能的重点放在网络结构优化而不是参数调整上,避免因参数过多而造成系统的泛化能力不佳。此外,通过引入集成学习的方法将多组模型进行融合通常可以显著的提升性能,但集成时应注意模型间的独立性。最后,在

音频特征提取阶段如果能利用到立体声信息，可以提升系统对空间的感知能力，进而提升分类准确率。

关键词：音频场景分类；卷积神经网络；梅尔频率倒谱系数；集成学习

Acoustic Scene Classification Method Based on Convolutional Neural Network

Abstract

Acoustic Scene Classification (ASC) is a task that enables devices to make sense of their environment through audio analysis, which belongs to a branch of the computational auditory scene analysis (CASA). At present, ASC has been widely used in intelligent wearable devices, robotics sensing, context-aware services and other application scenarios. In recent years, the development of deep learning has accelerated the research process of audio scene classification. As an important model in the field of deep learning, Convolutional Neural Networks (CNN) have strong learning ability. By introducing models CNN as acoustic scene classifier, the classification accuracy can be improved considerably, and even the machine can exceed the human level. In order to explore the applicability of CNN in ASC field and find the method to improve the system performance. In this paper, three groups of systems are designed and compared by experiments. The main work is as follows:

This paper begins with the design of a baseline system based on the Mel frequency cepstrum coefficient and the gaussian mixture model and constructs a typical baseline system as a control group for subsequent systems by means of traditional machine learning. Then it studies the principle of acoustic scene classification system based on CNN, probes into the applicability of convolution neural network in audio scene classification, and designs a basic system with two-layer convolution module. When training the system, filter parameters are adjusted to give full play to its classification potential, and the training time is also taken into account in system performance evaluation. In the evaluation stage, the classification accuracy of the basic system in each category is analyzed and the confusion matrix is introduced. It is found that the learning ability is better than the baseline system, but the generalization ability is poorer, and does not effectively use the spatial information in the audio file.

According to the problems of the basic system, this paper designs an improved system to make the basic system better from audio processing and network structure. In terms of audio processing, it uses binaural representation and harmonic percussive

source separation method to process the original audio and extract features, which has significantly improved the system classification accuracy in the scene with obvious spatial characteristics. As for network structure, the paper attempts to use the VGGNet structure in the field of image recognition for reference to improve the flexibility of the system while increasing the network depth, and finally achieve better generalization effect on different data. In addition, the improved system also uses Stacking method in ensemble learning to fuse multiple independent models based on different characteristics. Compared with the subsystem, the classification performance of the integrated system has further improved.

Through experimental and comparison, the final conclusion is that in the field of ASC, convolutional neural networks has better learning ability than traditional machine learning methods. In designing the convolutional neural network, we should pay attention to the flexibility of the network, and focus on improving the performance of the system on the network structure optimization rather than the parameter adjustment, to avoid the poor generalization ability caused by too many parameters. In addition, the integration of multiple sets of models by introducing ensemble learning methods can usually significantly improve the performance, but the independence between models should be paid attention to during the integration. Finally, if the stereo information can be utilized in the audio feature extraction stage, the system's perception of space can be improved, thereby the classification accuracy will be improved as well.

Keywords: Acoustic Scene Classification; Convolutional Neural Network; Mel frequency cepstral coefficient; ensemble learning

目 录

摘 要	I
Abstract	III
第 1 章 引言	1
1.1 课题背景与研究的目的及意义	1
1.2 国内外研究发展及现状	2
1.2.1 卷积神经网络	2
1.2.2 音频场景分类	3
1.3 主要研究内容与章节安排	5
1.3.1 主要研究内容	5
1.3.2 章节安排	6
第 2 章 音频场景分类基线系统	7
2.1 基线系统结构简介	7
2.2 音频特征的提取与处理	7
2.2.1 常用音频特征概述	8
2.2.2 MFCC 特征	9
2.3 GMM 模型原理	12
2.3.1 高斯混合模型	12
2.3.2 GMM 的参数估计	13
2.3.3 EM 算法	13
2.3.4 EM 算法在 GMM 模型中的应用	14
2.4 实验准备与结果	15
2.4.1 实验环境与系统细节	15
2.4.2 实验数据集	16
2.4.3 评价指标	17
2.4.4 实验结果与分析	18
2.5 本章小结	19
第 3 章 基于卷积神经网络的音频场景分类系统	20
3.1 引言	20
3.2 卷积神经网络原理	21
3.2.1 卷积层	21
3.2.2 激活函数	22
3.2.3 池化层	22
3.2.4 卷积神经网络的结构	23
3.2.5 卷积神经网络的特点	24
3.2.6 卷积神经网络的参数学习	25
3.3 卷积神经网络在音频场景分类中的适用性	26
3.3.1 卷积层的应用	26
3.3.2 池化层的应用	27
3.3.3 全连接层的应用	28

3.4 系统设计	28
3.4.1 特征图与预处理.....	28
3.4.2 卷积神经网络架构.....	29
3.4.3 批标准化	30
3.4.4 Dropout 机制	30
3.4.5 网络模型优化算法.....	31
3.4.6 模型训练	32
3.5 实验结果与分析.....	33
3.5.1 实验环境与数据集处理	33
3.5.2 评价指标	33
3.5.3 滤波器参数调整.....	33
3.5.4 结果分析	36
3.6 本章小结	38
第4章 音频场景分类系统的改进	39
4.1 引言	39
4.2 音频处理	39
4.2.1 双耳表示法	39
4.2.2 谐波冲击源分离法.....	41
4.3 改进系统的结构设计	42
4.4 集成学习	45
4.4.1 集成学习的概念.....	45
4.4.2 Stacking 方法	46
4.4.3 Stacking 在改进系统中的应用	47
4.4.4 随机森林算法	48
4.5 实验结果与分析.....	49
4.5.1 实验环境与数据集处理	49
4.5.2 结果分析	50
4.5.3 与其他系统的对比.....	51
4.6 本章小结	53
结 论	54
致 谢	55
参考文献	56
攻读学位期间取得学术成果.....	60

第1章 引言

1.1 课题背景与研究的目的及意义

声音作为一种信息载体，是我们感知外部环境的重要途径。随着信号处理技术与计算机科学的发展，借机器辅助人们从声音中提取出信息的音频处理任务得到了越来越多研究者的关注。相比于同为多媒体信息的图像，音频文件的采集不受光线环境、视觉障碍的限制，此外音频相对于图像占用的容量更少且处理速度更快。音频处理任务具体包括语音识别、音频指纹、音乐标记、音频场景分类等，本文主要研究音频场景分类。

音频场景分类属于计算听觉场景分析的子领域，其主要目标是通过分析声音使设备能够理解并分辨其环境。实现过程为：先对采集到的音频信号进行预处理，再从中提取用于区分环境的有用特征，最后根据这些特征进行分类。举例来说，当设备识别到谈笑声、餐具撞击声等音频信息的组合，可以很容易的与引擎声、鸣笛声的组合进行区分，将前者的音频场景判断为餐厅，后者判断为街道。其实现原理在于设备通过音频场景分类技术，将不同音频通过特征提取以得到对应的特征，再根据这些特征进行音频场景的建模，即构造一个分类器。分类器在学习到充分样本后，会根据提取到的音频特征判断所属的音频场景类别。

音频场景分类的应用包括上下文感知服务 (Schilit et al., 1994)，智能可穿戴设备 (Xu et al., 2008)，机器人传感 (Maxime et al., 2014) 和机器听觉系统 (Barchiesi et al., 2015) 等。此外，音频场景识别也与几个相关领域的研究有着相得益彰的关系，其中音频事件的检测与分类就常与音频场景分类联系在一起，原因在于音频场景可以看作是若干个音频事件叠加的产物。从另一方面讲，音频场景分类可以通过提供关于某些事件概率的先验信息来提高声音事件检测的性能 (Heittola et al., 2013)。与音频事件检测和分类技术密切相关的应用包括声源识别算法 (Defréville et al., 2006)、监视系统 (Radhakrishnan et al., 2005)，老年人援助 (Guyot et al., 2013) 和通过音频场景分割进行语音分析 (Hu and Wang, 2007)。

此前音频场景分类的实现常将通用分类器（如高斯混合模型、支持向量机、隐马尔可夫模型）应用于手工提取的特征，例如梅尔频率倒谱系数。近年来，得益于计算机速度的提升与深度学习的快速发展，人们逐渐意识到可以用深度学习自动特征提取的特性来代替以往低效的手工提取。与此同时，越来越多的录音设备如智能手机为音频数据集的扩充起到了重大贡献。有了海量的音频数据，使得以往难以实现的深度学习方法成为了可能。

有几种深度学习体系结构的变体，卷积神经网络其中的一种，由于其在学习独特的局部特征方面的优越性能，被广泛用于图像分类、语音识别、自然语言处理等领域。相比于其他深度学习结构，卷积神经网络在处理图像和音频方面的能力更强。与其结构类似的深度、前馈神经网络相比，卷积神经网络即使在有限的数据集和简单的数据增强下也可以有效应用于音频场景分类任务。更重要的是，可用数据集规模的显著增加很可能大大提高模型的性能。

1.2 国内外研究发展及现状

早在 1997 年，MIT 媒体实验室就已经展开了音频场景分类的研究工作。在研究起步时期，识别率不甚理想。随着如今智能设备大量涌现，优秀的计算机性能与深度学习技术的发展共同推动了该领域的研究进程。结合本文主要研究内容，下面将从卷积神经网络、音频场景分类两方面介绍国内外研究发展及现状。

1.2.1 卷积神经网络

卷积神经网络（Convolutional Neural Network, CNN）为深度学习中一类重要的神经网络模型。区别于其他神经网络模型诸如玻尔兹曼机、递归神经网络等，其重要的特点在于核心运算为卷积操作。得益于这个特性，卷积神经网络在许多领域如图像分类、检索以及物体检测等领域表现优异。随着卷积神经网络研究的深入，包括音频场景识别等越来越多的领域都采用了这一模型，并得到了相比传统方法更好的效果。下面将先回顾卷积神经网络的研究发展，再延伸到应用现状。

早在上个世纪六十年代，Wiesel and Hubel (1965) 通过对猫视觉皮层细胞的研究，提出了感受野（receptive field）的概念，这标志着神经网络结构首次在大脑视觉系统中被发现。而日本学者福岛邦彦 (Fukushima, 1980) 在 Wiesel 与 Hubel 的感受野概念的基础上提出了神经认知机（neocognitron），其目标在于处理手写字符识别等模式识别任务。在神经认知机模型中，有两类重要的神经元即“S 型细胞”和“C 型细胞”，这两类神经元交替堆叠共同构成了神经认知网络。其中，S 型细胞用于抽取局部特征，这类似于如今卷积神经网络中的卷积层结构。而 C 型细胞用于抽象与容错，这又与目前卷积神经网络中的池化层相对应。神经认知机的提出不仅是感受野概念在人工神经网络中的首次应用，也可看作是卷积神经网络的首次实现。

卷积神经网络结构的确立源自于 1998 年 Yann LeCun 的一篇论文 (Yann LeCun et al., 1998)，他们设计了一种名为 LeNet-5 的多层人工神经网络，可以对手

写识别数字做分类。卷积神经网络也像其他神经网络一样可以使用反向传播算法进行训练。在当时的技术条件下, LeNet-5 就可以实现低于 1%的错误率。由此该算法结构得到了当时整个美国绝大多数邮政系统的应用。LeNet-5 可以算是第一个产生实际商业价值的卷积神经网络应用。

直到 2006 年, 由 Geoffrey Hinton 等人提出的深度置信网络 (Hinton et al., 2006) 与受限玻尔兹曼机 (Salakhutdinov et al., 2007) 的学习算法才重新使人工智能领域对神经网络产生了足够的关注。卷积神经网络的热潮的掀起则是由于 2012 年开始举办的 ImageNet 图像分类比赛。赛中 Krizhevsky et al. (2012) 提出了一个经典的卷积神经网络结构, 并在图像识别任务上取得了重大突破。这一结构类似于 LeNet-5, 名为 AlexNet, 与前者不同的是其层次结构更深。同时 AlexNet 还使用了非线性激活函数 ReLu 与 Dropout 方法, 最终取得了卓越的效果。

1.2.2 音频场景分类

Sawhney and Maes (1997) 在 MIT 媒体实验室的技术报告中提出一种专门解决音频场景分类问题的方法。作者记录了一组包括“人”, “声音”, “地铁”, “交通”和“其他”的一组数据集。他们利用语音分析和听觉研究借鉴的工具从音频数据中提取了几个特征, 采用递归神经网络和 k 最近邻标准对特征和类别之间的映射进行建模, 并获得 68% 的整体分类准确率。一年后, 来自同一机构的研究人员 Clarkson et al. (1998) 等通过戴着麦克风录制连续的音频流, 同时进行一些超市自行车旅行, 然后自动将音频分割成不同的场景 (如“家”, “街道”和“超市”)。他们将从音频流中提取的特征的经验分布拟合成隐马尔可夫模型 (Hidden Markov Model, HMM)。

与此同时, 实验心理学的研究则着重于理解驱动人类对声音和场景进行分类和识别的能力的感知过程。Ballas (1993) 发现识别声音事件的速度和准确性与刺激的声学性质、它们发生的频率及是否它们可以与物理原因或声音刻板印象相关联有关。Peltonen et al. (2001) 观察到人类对音频场景的认识是通过识别典型声音事件诸如人声或汽车发动机噪声来实现的, 并且确定了人类识别 25 个声场中的能力的整体准确率为 70%。Dubois et al. (2006) 研究了在不是实验者先验的情况下, 个体如何定义他们自己的语义类别分类。最后, Tardieu et al. (2008) 测试了语义类的出现以及在火车站范围内对声场的识别。他们在报告中说, 声源、人类活动以及房间效应 (如混响) 是促成音频场景形成的因素, 也是类别为固定先验情况下的识别线索。

受心理声学/心理学文献的影响, 这些文献强调音频场景分类的局部特征和

全局特征，一些麻省理工学院研究人员则侧重于音频的时域特征。Eronen et al. (2003) 采用 Mel 频率倒谱系数 (Mel-Frequency Cepstrum, 简称 MFCC) 来描述音频信号的局部频谱包络, 用高斯混合模型 (Gaussian Mixture Model, GMM) 来描述其统计分布。然后, 他们通过利用训练信号种类的知识判别式算法来训练 HMM, 以解释 GMM 的时域演变。Eronen 及其合作者通过考虑更多的特征, 和在分类算法中增加一个特征变换步骤, 进一步推进了这项工作, 在 18 种不同的声场中获得了总体 58% 的准确性。

2015 年, Piczak et al. (2015) 对深度学习中的卷积神经网络是否可有效的应用于音频场景分类这一问题进行了探讨。为此他们依照此前将卷积神经网络成功用于图像分类的经验运用于音频场景分类上。实验表明, 使用卷积神经网络进行音频场景分类是一个切实可行的办法。卷积神经网络模型胜过基于手动设计特征的常用方法, 并达到与其他特征学习方法类似的水平。且卷积神经网络即使在有限的数据集和简单的数据增强下也可以有效应用于环境声音分类任务。其更重要的发现在于, 可用数据集规模的增加会大大提高卷积神经网络的分类性能。

尽管关于音频场景分类系统的文献丰富, 但研究界缺乏协调一致的标准来评估和测试解决这个问题的算法。2013 年, IEEE 音频和声学信号处理 (AASP) 技术委员会首次组织了 DCASE (音频场景和事件检测和分类) 挑战赛, 以测试和比较音频场景分类和事件检测算法。这一举措符合信号处理领域旨在促进可再生研究的目标。在这个挑战赛中, 主办方团队采集了大量高质量的音频文件用于比赛, 且每一年除了基本的音频场景分类任务以外, 还包括其他丰富的挑战项目, 诸如日常生活中的音频事件检测、稀有音频事件检测、鸟类音频检测等。过去几年来, 本挑战赛中已经提出了许多音频场景分类及音频事件检测技术, 对整个音频场景分类领域的发展做出了极大的贡献。其中音频场景分类从一开始就是一项受欢迎的任务, 在过去的几年中, 参与者数量最多。

DCASE 挑战赛开赛以来, 得益于参赛者的增多与主办方的重视, 开发数据集的大小逐渐增加。最初 DCASE 2013 的开发集中有 10 个场景类, 每类中仅 10 个样本。DCASE 2016 中场景类有 15 个, 每类 78 个样本。到 2017 年比赛中依然有 15 个场景类, 且每类更是多达 312 个样本。此外, 2017 年的比赛数据集中还引入了测试集, 与开发集中的数据相区别, 专门用于测试系统的分类性能。鉴于可获得的数据量增多, 2016 年的比赛标志着向深度学习方法的明确过渡。48 个提交的比赛模型中的 22 个都使用都了某种形式的深度学习 (Mesaros et al. 2018), 尤其是卷积神经网络得到了大量应用。

Schindler et al. (2016) 通过使用 CQT (常数 Q 变换) 特征作为卷积神经网络的输入来增强结果。使用 CQT 的动机来自于音乐感知领域的观点: 人类听觉

系统在大部分可听频率范围内可以近似为“常量 Q ”。该系统的关键是利用 CQT 以足够的分辨率捕获来自低频和高频的基本音频信息，并创建一个并行卷积神经网络架构，该架构能够及时捕获这两种频率。所呈现的深度学习神经网络架构已经比 DCASE 2016 声场景分类任务组织提供的基线系统超出 10.7% 的相对改进，在开发集合上达到 80.25%。此外，它在验证集中达到了 83.3%。

由于此前在音频场景分类领域缺乏大型标记的声音数据集，获得这些数据集通常既昂贵又不明确。来自 MIT 的 Aytar et al. (2016) 寄希望于通过利用视觉和声音之间的自然同步来学习来自未标记视频的音频特征来扩大规模，因此他们利用超过一年的野外采集的声音来学习语义丰富的音频特征。未标记的视频可以大规模、低成本的获得，且具有音频信号。计算机视觉方面的最新进展使机器能够高精度地识别图像和视频中的场景和对象。而如何将视频中的知识转化为标记音频的标签成为了研究的关键。在实验中，他们使用了可以直接在原始音频波形上学习的卷积神经网络，通过将知识从视觉传输到声音进行训练。尽管网络是通过视觉监督进行训练的，但网络在推理过程中不依赖视觉。结果表明，与简单的全连接的网络或较早的图像分类体系结构相比，最先进的图像网络在音频分类方面具有出色的结果。其对较大的标签集词汇进行训练可以提高性能，尽管在对较小的标签集进行评估时性能稍有提高。

1.3 主要研究内容与章节安排

1.3.1 主要研究内容

本文的核心目标是设计一个分类性能良好的基于卷积神经网络的音频场景分类系统。要设计一个性能良好的音频场景分类系统，首先应具备一个性能尚可的系统作为对照，即基线系统。之后，再根据卷积神经网络在音频场景分类中的应用特点设计本文中的核心系统，即基于卷积神经网络的音频场景分类系统。通过比较分析核心系统与基线系统在各场景下的分类性能差别，研究核心系统相对于基线系统的优缺点，再根据其不足之处进行改进。最后，改进后的系统即改进系统应在分类水准上优于基线系统，并具有更好的泛化能力。

尽管将多种深度学习方法应用于音频场景分类中加快了这一领域的研究过程，但总结之前科研工作者在这一方面的工作，还有以下四点问题值得探讨：

(1) 主流的音频特征提取方法都是基于梅尔频谱图进行的，那么是否存在一些对梅尔频谱图的特殊处理技巧，使其能更符合具体的应用场景，以提高音频分类的准确率。

(2) 通常在音频场景分类中使用的分类器主要是单一模型, 可否将机器学习领域中的集成学习方法应用于音频场景分类中, 通过集成多个不同模型以获取更好的性能。

(3) 由于音频特征提取过后的大多数结果也属于图像, 那么图像识别领域的优秀卷积神经网络架构能否应用到音频场景分类中来, 以提高系统性能。

(4) 在卷积神经网络的设计过程中, 很少有研究者会将注意力放在网络训练的单位迭代周期上。通常在训练网络时都会耗费大量的时间与计算资源, 那么能否将系统模型的效能也作为一个重要的衡量指标, 因为更快的网络运算速度意味着可以将更多精力放在对系统架构的研究上去。

针对以上四点问题, 本文将尝试在实现系统的过程中将其考虑进去。其中问题(4)会在第3章中训练卷积神经网络的调整参数阶段涉及到。而问题(1)、(2)、(3)将会放在第4章的内容中去研究, 作为改进系统的一部分。

1.3.2 章节安排

第1章为引言部分。其中包括了课题背景与研究的目的及意义, 并从卷积神经网络和音频场景分类两个方面介绍了对应领域的国内外研究现状及现状, 根据研究现状来探讨目前研究存在的问题。最后, 给出了文章的研究内容与章节安排。

第2章为音频场景分类基线系统的设计与实验。主要研究关键特征 MFCC 的原理及其提取过程。并且对分类器模型 GMM 的原理进行详细阐述, 主要包括 EM 算法的原理与其在 GMM 模型中的应用。最后, 将进行基线系统的测试, 并记录基线系统的分类准确率。

第3章为基于卷积神经网络的音频场景分类系统的设计与实现。首先介绍系统的总体架构, 然后详细介绍卷积神经网络的关键结构、原理及特点, 再根据这些基础知识探讨将卷积神经网络应用在音频场景分类领域的适用性。接着, 将探讨音频场景分类中卷积神经网络的训练方法。通过预训练调整网络参数, 并考虑到系统效能, 以得出最佳网络参数。最后, 根据之前的设计对模型进行训练并分析训练结果, 与基线系统作对比。

第4章为在前一章初步设计好的系统上进行一定的优化。本章中在原来系统的基础上, 通过引入新的特征, 来增强分类准确率。之后, 还对网络结构的改进做了进一步探索。此外, 本章中使用流行的集成学习算法 Stacking 来对多个不同模型进行集成。最后在实验中将第2章的基于 GMM 分类器和第3章基于卷积神经网络的模型进行对比。

结论部分对本文已经完成的工作做了总结, 并展望了之后工作的研究方向。

第 2 章 音频场景分类基线系统

2.1 基线系统结构简介

本章主要内容为设计基于 MFCC 特征及 GMM 分类器的音频场景分类基线系统。MFCC 作为语音识别领域常用的重要特征，有着良好的区分性能。而且研究也表明（Reynolds et al., 1995），在 GMM 作为分类器与 MFCC 等性能良好的特征配合时，有着出色的分类性能。此外，相比与目前广泛应用的深度学习算法，GMM 模型具有更快的训练速度，一旦系统设计完成，很快就能得到训练结果。因此选取以上经典方法作为音频场景分类的基线系统，作为之后设计的基于卷积神经网络的分类器的分类性能参考标准。

基线系统主要包括五个部分，分别为音频数据集、特征提取与处理、系统训练、系统测试、系统评估。整个系统结构如图 2-1 所示。



图 2-1 基线系统结构

本章首先研究音频特征的提取与处理部分，然后讨论 GMM 模型原理及其核心的 EM 算法，最后介绍实验的环境、数据集、评价指标与结果。

2.2 音频特征的提取与处理

提取音频信号的最佳参数表示是产生更好识别性能的重要任务之一。这一阶段的特征提取对下一阶段的分类器分类很重要，因为它会影响之后的分类效率。音频可以用多种方式表示，哪一种“最佳”取决于应用以及处理机器。多年来，特征设计和选择是许多音频分析任务的关键组成部分，经常使用的特征包括简单时频特征、频带能量特征、倒谱特征、发生特征、线性预测系数等。本节将首先概述常用音频场景分类中常用的音频特征，之后再对本文中使用的核心特征 MFCC 做详细的阐述并探究其提取方法。

2.2.1 常用音频特征概述

(1) 简单时频特征: Eronen et al. (2006) 及 Malkin et al. (2005) 设计的音频场景分类系统中, 采用了这类特征, 其特点为可以通过简单时域计算或通过傅里叶变换得到。其中包括过零率 (Zero Crossing Rate):

$$zcr = \frac{1}{T-1} \sum_{t=1}^{T-1} f(s_t s_{t-1} < 0) \quad (2-1)$$

其中 s 是长度为 T 的信号, $f(x)$ 在参数 x 为真时为 1, 假时为 0。过零率测量信号内的符号变化的平均速率, 并且与单声道音频的主频率相关, 是分类敲击声的关键特征。

简单时频特征还包括了谱质心 (Spectral Centroid) (郑继明 等, 2009), 其测量频率成分的中心, 与音色的明亮度有关。谱质心的特征参数计算方法如下:

$$sc = \frac{\sum_{n=1}^N f(n)E(n)}{\sum_{n=1}^N E(n)} = \sum_{n=1}^N f(n) \cdot P(E(n)) \quad (2-2)$$

其中, f 为信号的频率, $E(n)$ 为连续时域信号 $x(t)$ 经短时傅里叶变换后的对应频率的谱能量。

此外简单时频特征还包括频谱滚降, 它识别频率高于设定阈值的频率。

(2) 频带能量特征 (能量/频率): Eronen et al. (2006) 中使用的这类特征, 是通过在指定频带上对幅度谱或功率谱进行积分得到的。得到的系数用来衡量不同子带内存在的能量, 并且还可以表示为子带能量与总能量之间的比率, 以编码信号中最突出的频率区域。

(3) 听觉滤波器组: 能量/频率特征的进一步应用在于通过滤波器组分析音频帧, 以模仿人类听觉系统的响应。它通过了一组带通滤波器, 输出具有一定中心频率的子带信号。其中, Sawhney and Maes (1997) 使用 Gammatone 滤波器组, Jung et al. (2017) 采用了 Mel 滤波器组, 而 Patil and Elahili (2002) 则使用了听觉谱图。

(4) 倒谱特征: 倒谱特征是为了某些时候便于计算, 将原信号的频谱转化为类似分贝的单位, 再对其做逆傅里叶变换, 将其视为一种新信号处理。MFCC 则为音频场景分类中最常用的倒谱特征之一。其得以大量应用的主要原因就在于 MFCC 的频带划分是在梅尔刻度上等距划分的, 与常规对数倒谱中的线性间隔频带相比, MFCC 频带更接近人类的听觉系统。且通过 MFCC 的处理, 通常将许多复杂的特征用十几个简洁系数描述。在下一小节中将详细介绍该特征的原理以及提取过程, 并以此特征为基础进行系统的设计工作。

(5) 发声特征：每当认为信号中包含谐波分量时，可以估计其基频 f 或一组基频，并且可以定义特征组以测量这些估计的特性。在音频场景分类的领域中，谐波分量可能与音频场景内发生的特定事件相对应，且可以通过其识别确定不同的音频场景。Krijnders and Holt (2013) 提出的方法基于提取音调拟合特征，即从音频信号的感知动机表示导出的一系列发声特征。首先，计算耳蜗图以提供受人耳特性启发的音频场景的时频表示。然后，评估每个时频区域的音调以识别音频场景中的音调事件，从而产生音调拟合特征向量。

(6) 线性预测系数 (Linear Prediction Coefficient, LPC)：线性预测是根据当前信号采样点用线性函数计算未来离散时域信号的方法，其基本思想为一个音频样本可以通过过去若干时刻音频样本的组合来逼近。线性预测系数在语音分析中有着广泛的应用，原因不仅在于其模型参数简单、计算量较小，而且能较精确的表示音频信号的频谱幅度。假设 $x(n), n = 0, 1, \dots, N - 1$ 为音序列的第 n 帧，则其 P 阶预测值 $\hat{x}(n)$ 可以表示为：

$$\hat{x}(n) = - \sum_{i=1}^p a_i x(n-i) \quad (2-3)$$

其中 $x(n-i)$ 为之前采样的值， a_i 为预测系数，LPC 产生的误差为：

$$e(n) = x(n) - \hat{x}(n) \quad (2-4)$$

由于 LPC 的值与建模信号频谱包络之间存在映射，因此 a_i 编码包含关于声音的一般频谱特性的信息。Eronen et al. (2006) 在他们提出的方法中采用了 LPC 特征。

(7) 常数 Q 变换 (CQT)：常数 Q 变换是在 FFT 的基础上得来的。Zeinali et al. (2018) 在其音频场景分类系统中使用过此特征。在使用 FFT 进行频率分析时，由于频率是按线性分布，不同于人耳的感知频率，因此引入常数 Q 变换。常数 Q 变换为一滤波器组，其中心频率按指数规律分布，且滤波带宽也不同，中心频率与带宽的比值为常数 Q。区别于 FFT，其频率是按 \log_2 为底分布的，且可以根据频率谱线的不同选取不同的滤波器窗长。常数 Q 变换尤其适合处理音乐信号，原因在于其频率分布与音阶的频率分布相同。

2.2.2 MFCC 特征

在分类任务，尤其是音频分类任务中，描述光谱形状的梅尔频率倒谱系数 (MFCC) 具有悠久的历史。尽管 MFCC 提取过程中会造成数据的有损压缩，但其分类与识别效果在数据速率很低时也具备相当的可用性，且相对于其他分类特征，MFCC 由于更符合人耳听觉频率响应曲线，因此得到了广泛的应用。

人类之所以能在复杂的声音环境中判断出不同的环境,主要在于耳蜗的功劳。耳蜗可以看作为一个滤波器组,帮助人们过滤 20-20KHz 的音频。问题在于耳蜗对于听觉范围内频率的灵敏度并不是线性的,存在一种映射关系。因此为了模拟出人耳的频率响应,Davis et al. (1980) 提出了 MFCC 特征。

MFCC 特征提取由七个步骤组成,整个过程如图 2-2 所示,下面将给出提取过程的详细分析。

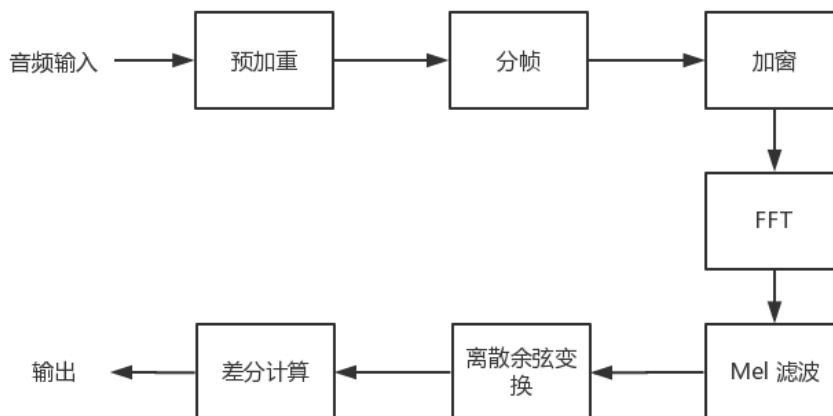


图 2-2 MFCC 特征提取步骤

(1) 预加重

常见的音频信号存在低频能量大而高频能量小的现象,如果直接传输会导致低频信噪比很高而高频信噪比不足。为了弥补音频信号在传输过程中这种损耗,引入预加重对输入信号进行补偿,使音频信号的高频特征得以凸显。通常借助高通滤波器来实现预加重。

设第 n 时刻的语音采样值为 $X[n]$,经过预加重处理后的结果是:

$$Y[n] = X[n] - \alpha X[n-1] \quad (2-5)$$

其中 α 为预加重系数,通常取 0.9~1.0 之间。

(2) 分帧

分帧将从模数转换(ADC)中得到的音频样本分割为长度在 20 至 40 毫秒范围内的小帧。音频信号被分 N 个样本的帧。相邻帧由 M ($M < N$) 分隔。通常取 $M=100$, $N=256$ 。

(3) 加窗(海明窗)

在预加重与分帧完成后,需要为每一帧加上海明窗(Hamming window)。加窗是为了控制数据处理量,每次仅处理窗中数据。由于之后 FFT 中的处理对象是有限长信号,对无限长信号强行进行有限个点的 FFT 会丢失频率信息导致频

谱泄露。海明窗的公式可以总结如下：

假设窗函数为： $W(n)$, $0 \leq n \leq N - 1$ 。且 N 为每帧中的样本数， $Y(n)$ 为输出信号， $X(n)$ 为输入信号，则加窗结果如式（2-6）。

$$Y(n) = X(n) \cdot W(n) \quad (2-6)$$

其中，

$$W(n) = 0.54 - 0.46 \cdot \cos\left\{\frac{2\pi n}{N-1}\right\}, 0 \leq n \leq N-1 \quad (2-7)$$

（4）快速傅立叶变换（FFT）

将 N 个样本的每个帧从时域转换到频域。具体过程如式（2-8）。

$$Y(w) = FFT[h(t) \otimes X(t)] = H(w) \cdot X(w) \quad (2-8)$$

其中 \otimes 表示卷积运算。

（5）Mel 滤波

FFT 频谱中的频率范围非常宽，语音信号不遵循线性标度。因此通过如图 2-3 所示的 Mel 标度滤波器组。

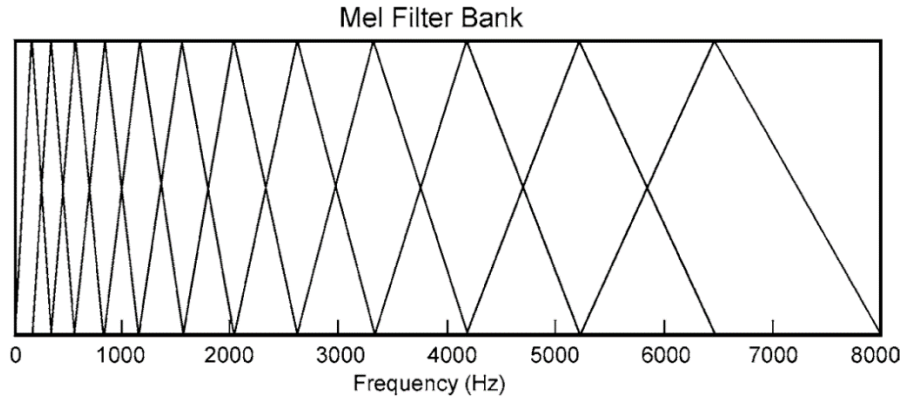


图 2-3 Mel 标度滤波器组

该图表示了一组三角形滤波器，用于计算滤波器频谱分量的加权和，使得处理后的输出近似于 Mel 标度。每个滤波器的幅度频率响应是三角形的，在中心频率处等于 1，在两个相邻滤波器的中心频率处线性减小到零，且每个滤波器输出是其滤波后的频谱分量的总和。最后，使用式（2-9）计算给定频率 f 的 Mel。

$$F(\text{Mel}) = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right) \quad (2-9)$$

（6）离散余弦变换

该过程为使用离散余弦变换（DCT）将 Mel 频谱转换为时域的过程。转换的结果称为 Mel 频率倒谱系数。系数集称为声矢量。因此，每个输入都被转换为音频矢量序列。

（7）差分计算

以上取得的倒谱参数只能反应音频信号的静态特性。为了提高信号的识别性能，应采用音频信号静态特性的差分谱来描述音频信号的动态特性。因此引入了 13 个一阶差分特征（12 个倒谱特征加能量），以及 39 个二阶差分特征。从时间 t_1 到 t_2 的窗口中信号 X 的帧能量如式（2-10）。

$$Energy = \sum_{t=t_1}^{t_2} X^2(t) \quad (2-10)$$

13 个一阶差分特征中的每一个特征表示 MFCC 特征中对应的倒谱或能量特征的帧之间的变化，而 39 个二阶差分特征中的每一个表示对应的一阶差分特征中的帧之间的变化。其中一阶差分的计算如式（2-11）。

$$d(n) = \frac{c(n+1) - c(n-1)}{2} \quad (2-11)$$

其中， $c(n+1)$ 表示 $n+1$ 时刻的倒谱系数。

2.3 GMM 模型原理

GMM 即高斯混合模型，是表示为高斯分量密度加权和的参数概率密度函数。目前 GMM 已经广泛应用于语音识别、音频场景分类、音频事件检测（金海，2016）、音频检索等领域，得到了研究人员的广泛的认可。GMM 使用迭代期望最大化（Expectation-maximization, EM）算法从训练数据估计中 GMM 参数。下面将详细介绍 GMM 模型。

2.3.1 高斯混合模型

M 阶 GMM 是由 M 个高斯概率密度函数加权求和而成，如式（2-12）所述。

$$P(x|\lambda) = \sum_{i=1}^M w_i g(x|\mu_i, \Sigma_i) \quad (2-12)$$

其中， x 是 D 维连续特征向量， w_i , $i = 1, \dots, M$ ，为加权系数， $g(x|\mu_i, \Sigma_i)$, $i = 1, \dots, M$ 为高斯密度分量。每一个密度分量可表示为：

$$g(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu_i)' \Sigma_i^{-1} (x - \mu_i) \right\} \quad (2-13)$$

其中 μ_i 是均值向量， Σ_i 是协方差矩阵，且 M 个权重 w_i 满足约束条件 $\sum_{i=1}^M w_i = 1$ 。完整的高斯混合模型由均值向量、协方差矩阵和混合权重组成，统一表示如式（2-14）。

$$\lambda = \{w_i, \mu_i, \Sigma_i\} i = 1, \dots, M \quad (2-14)$$

利用 GMM 进行音频场景分类的核心思想在于：在音频文件经过特征提取与处理后，我们可以得到特征序列。在忽略时序信息的条件下，通过以帧为单位，利用 GMM 对音频信息进行建模。由于音频事件中每帧信号对应的特征划分为若干类，类与类之间的音频特征又相互独立，且音频特征均服从相同的正态分布。所以可以将多个类的正态分布按照一定权重进行组合，代表某类音频特征的总体分布，以表现该音频场景的特征。

2.3.2 GMM 的参数估计

在给定了训练向量和 GMM 配置后，我们希望估计 GMM 的参数 λ ，该参数在某种程度上与训练特征向量的分布最匹配。有几种技术可用于估计 GMM 的参数，迄今为止最流行和最成熟的方法是最大似然估计。

最大似然估计的目的是在已知训练特征向量集的情况下，找到使 GMM 似然函数最大的模型参数 λ 。对一组长度为 T 的训练特征向量序列 $X = \{x_1, \dots, x_T\}$ ，假设向量间相互独立，则 GMM 似然度可表示为式（2-15）的形式。

$$P(X|\lambda) = \prod_{t=1}^T P(x_t|\lambda) \quad (2-15)$$

但上式为参数 λ 的非线性函数，无法直接求出极大值，故不能显式的求出参数 λ 。然而，可以使用期望最大化（EM）算法的特殊情况迭代地求出最大似然参数估计。

2.3.3 EM 算法

EM 算法（Dempster et al., 1977）是当数据不完整或有缺失值时从给定数据集中找到基础分布的参数的最大似然估计的一般方法。EM 算法的每次迭代由两部分组成，第一步为 E 步，即求期望（expectation）过程；第二步为 M 步，即求极大（maximization）过程，故该算法简称为 EM 算法。由上一小节总结可以得出，在概率模型中含有隐变量时，不能依赖于极大似然估计法，进而引入 EM 算法以解决含有隐变量时概率模型参数的极大似然估计。EM 算法保证了在每一次的迭代中都使模型的似然度 $P(X|\lambda)$ 增大，其证明如下：

我们假设 X 为一组长度为 T 的训练特征向量序列 $X = \{x_1, \dots, x_T\}$ ，模型参数的似然度为 λ ，则 GMM 的训练目的在于找出使 $p(X|\lambda)$ 最大的模型参数 λ ，也即：

$$\bar{\lambda} = \arg \max_{\lambda} P(X|\lambda) \quad (2-16)$$

由于 $p(X|\lambda)$ 为参数 λ 的非线性函数，无法直接求最大值，故引入 Q 函数

(Nuttall et al. 1972), 应用如下:

$$Q(\lambda, \bar{\lambda}) = \sum_{i=1}^M P(X, i|\lambda) \log P(X, i|\bar{\lambda}) \quad (2-17)$$

其中, i 为高斯分量序号, $\bar{\lambda} = \{\bar{w}_i, \bar{\mu}_i, \bar{\Sigma}_i\}$ 为估计的新模型参数, $P(X, i|\lambda)$ 为在模型参数为 λ 时, X 归于高斯分量序号为 i 的概率密度。

进一步的, 有:

$$\begin{aligned} Q(\lambda, \bar{\lambda}) - Q(\lambda, \lambda) &= \sum_{i=1}^M P(X, i|\lambda) \{ \log P(X, i|\bar{\lambda}) - \log P(X, i|\lambda) \} \\ &= \sum_{i=1}^M P(X, i|\lambda) \log \frac{P(X, i|\bar{\lambda})}{P(X, i|\lambda)} \end{aligned} \quad (2-18)$$

故:

$$Q(\lambda, \bar{\lambda}) - Q(\lambda, \lambda) \leq P(X, i|\bar{\lambda}) - P(X, i|\lambda) \quad (2-19)$$

当且仅当 $\bar{\lambda} = \lambda$ 时, 等号成立。因为 $Q(\lambda, \bar{\lambda})$ 与 $P(X, i|\lambda)$ 具有相同的单调性, 所以只需证明 $P(X, i|\lambda)$ 是关于 λ 对数函数的凹函数即可。对 $P(X, i|\lambda)$ 取关于 λ 的微分:

$$\begin{aligned} \nabla_{\lambda} P(X, \lambda) &= \nabla_{\lambda} \sum_{i=1}^M P(X, i|\lambda) = \sum_{i=1}^M \nabla_{\lambda} P(X, i|\lambda) \\ &= \sum_{i=1}^M P(X, i|\lambda) \nabla_{\lambda} \log P(X, i|\lambda) \end{aligned} \quad (2-20)$$

将 $Q(\lambda, \bar{\lambda})$ 带入上式, 可以得到:

$$\nabla_{\lambda} P(X, \lambda) = \nabla_{\lambda} Q(\lambda, \bar{\lambda}) |_{\bar{\lambda}=\lambda} \quad (2-21)$$

当 $\bar{\lambda} = \lambda$ 时, $P(X, \lambda)$ 与 $Q(\lambda, \bar{\lambda})$ 极值点相同。故 $P(X, \lambda)$ 与 $Q(\lambda, \bar{\lambda})$ 有相同的单调性和极值点。由此可以得出结论: 可以通过求 $Q(\lambda, \bar{\lambda})$ 的局部极大值来得到 $P(X, \lambda)$ 的新模型参数 $\bar{\lambda}$ 。

2.3.4 EM 算法在 GMM 模型中的应用

如前所述, GMM 算法的输入数据为一组长度为 T 的训练特征向量序列 $X = \{x_1, \dots, x_T\}$ 。要求的结果为 $\bar{\lambda} = \{\bar{w}_i, \bar{\mu}_i, \bar{\Sigma}_i\}$, 即新模型参数。在每次 EM 迭代中, 首先设定每个参数的初始值 $\lambda = \{w_i, \mu_i, \Sigma_i\}$ 。接着, 根据当前的模型参数, 计算分模

型 i 对 X 的响应度:

$$P(i|x_t, \lambda) = \frac{w_i g(x_t|\mu_i, \Sigma_i)}{\sum_{k=1}^M w_k g(x_t|\mu_k, \Sigma_k)} \quad (2-22)$$

此为EM算法中的E步。然后,使用以下重估公式计算新一轮模型的迭代参数:
混合权重:

$$\bar{w}_i = \frac{1}{T} \sum_{t=1}^T P(i|x_t, \lambda) \quad (2-23)$$

均值向量:

$$\bar{\mu}_i = \frac{\sum_{t=1}^T P(i|x_t, \lambda) x_t}{\sum_{t=1}^T P(i|x_t, \lambda)} \quad (2-24)$$

方差(对角协方差):

$$\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|x_t, \lambda) x_t^2}{\sum_{t=1}^T P(i|x_t, \lambda)} - \bar{\mu}_i^2 \quad (2-25)$$

此为EM算法中的M步。

最后,重复E步与M步,直至收敛。

2.4 实验准备与结果

2.4.1 实验环境与系统细节

本实验在以Linux为内核的Ubuntu系统下实现。系统使用处理器训练,配置为8核心的Intel E3-1270@3.8GHz处理器,系统内存为32GB。实验使用Python语言,引入了外部库Librosa进行特征提取。在训练部分引入了sklearn库,以进行应用GMM模型的非监督学习。音频特征部分包括60维MFCC特征向量,其中包括了20个MFCC静态系数(包括第0个)、20个一阶差分系数、20个二阶差分系数。分帧过程的帧长为40ms,帧移为帧长的50%即20ms,加窗过程采用汉明窗。对于每个音频场景,根据特征标签使用EM算法训练具有16个分量的GMM模型。测试阶段使用最大似然判定法判断音频场景的归类。使用准确度来衡量分类性能。

系统的实现细节如表2-1所示。

表 2-1 系统实现细节

音频输入	
通道	单通道（多通道平均至单通道）
标准化处理	无
音频特征	
类型	MFCC
窗口长度	40ms
帧移	20ms
特征向量	
构成	静态 MFCC+一阶+二阶
向量长度	60
高斯混合模型	
高斯分量数	16
协方差	对角线
参数数量	1936
模型	每个场景对应一个模型
决策方法	似然积累+最大值
积累窗口	信号长度

2.4.2 实验数据集

本实验使用的数据集为 TUT Acoustic Scenes 2017，负责采集的团队为坦佩雷理工大学音频研究小组。该数据集由 15 个不同标签的音频场景组成：沙滩、公交、咖啡馆/餐厅、汽车、市中心、森林小径、杂货店、家庭、图书馆、地铁站、办公室、公园、小区、火车和电车。所有音频文件都被切割成长度为 30 秒的片段，音频文件格式为 WAV。为了满足所有音频场景类别的高声学可变性的要求，每次录音均在不同的位置进行，录制的平均持续时间为 3-5 分钟。用于记录此特定数据集的设备包括双声道 Soundman OKM II Klassik / studio A3 入耳式麦克风和使用 44.1 kHz 采样率、24 位分辨率的 Roland Edirol R09 波形记录仪。

本文使用的数据集分为开发集和验证集两部分，其中，开发集包含 4680 个音频文件，每类场景的文件数皆为 312 个。其中大约 70% 的数据用于训练音频场景分类模型，剩下约 30% 用作测试。可以从数据集中提供的元数据文件或音频文件名中找到类别标识符。

而在验证集中，共有 1620 个音频文件，其中每类音频文件 108 个。每段音

频长度为 10 秒，每个场景音频总计 18 分钟。验证集中的文件皆为重新采集的音频，且全部数据都用于验证系统性能。验证集的所有文件在测试时都不带有标签，只会在测试结束后才会从独立的标签文件中提取音频的对应标签，因此验证集的分类准确率可能会显著低于开发集。

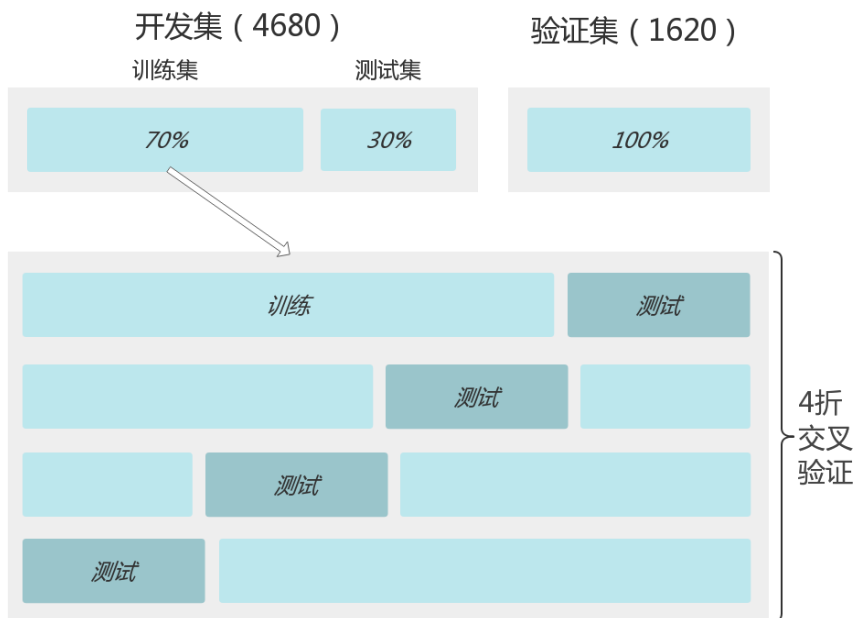


图 2-4 实验数据集详解

由于样本量较小，在训练阶段为了充分利用数据，将数据集等分为四份，其中每一份中各类型音频比例与总数据集一致。任意取其中一份作为测试集验证训练结果，其他三份用做训练数据，然后换其中另一份作测试集，其他份用于训练，如此操作共四次直到每份数据都被用于测试。此种方法称为 4 折交叉验证法（4-fold Cross Validation），保证了可以生成四组不同的训练与测试集。

2.4.3 评价指标

在使用开发集训练系统阶段，用训练集对 GMM 模型进行训练后，我们通过测试集来测试模型在开发集上的训练结果。在使用验证集测试系统性能阶段，每一个验证集中的数据都用来测试分类结果。两个数据集中训练结果的唯一评判标准为准准确率（ACC），即预测正确的样本数与总样本数间的比值。计算如式（2-26）所示。

$$ACC = \frac{N_{True}}{N_{Total}} \quad (2-26)$$

其中， N_{True} 为预测正确的样本数， N_{Total} 为总样本数。

2.4.4 实验结果与分析

如表 2-2 所示,通过 4 折交叉验证法的训练与测试,一共得出了四组准确率数据,且因为数据在切分时的总量与各类别比例都相同,因此总准确率为四组准确率的平均值。从数据中我们可以看出,不同子数据集之间因数据差异,训练准确度也各有差异,但总体准确度维持在了 74%左右,可以评价该基线系统对于不同数据能实现相对稳定的分类水平。

表 2-2 不同子集的总分类准确率 (单位: %)

场景	总体	1	2	3	4
沙滩	76.0	79.5	78.2	96.2	50.0
公交	83.0	89.7	75.6	87.2	79.5
咖啡/餐馆	78.5	61.5	93.6	84.6	74.4
汽车	92.0	88.5	89.7	89.7	100.0
市中心	86.5	78.2	94.9	76.9	96.2
森林小径	66.7	74.4	79.5	65.4	47.4
杂货铺	74.4	89.7	71.8	74.4	61.5
家	72.0	85.9	80.2	60.5	61.5
图书馆	60.6	16.7	73.1	75.6	76.9
地铁站	77.2	83.3	71.8	69.2	84.6
办公室	97.8	96.2	100.0	96.2	98.7
公园	48.4	75.6	44.9	21.8	51.3
小区	73.7	70.5	70.5	78.2	75.6
火车	36.2	38.5	29.5	20.5	56.4
电车	81.7	91.0	62.8	88.5	84.6
平均	73.6	74.6	74.4	72.3	73.2

如表 2-3 所示,在验证集上重新运行模型后,整体分类性能相比开发集有了明显下降。分析其原因,就在于使用验证集测试时不带有测试标签,而之前训练过程中存在对特定数据的记忆效应,在验证阶段新数据带来的随机性会导致分类性能的下降。由于验证集的测试数据量相比开发集更大,因此更能反映出系统的真实分类性能。

表 2-3 开发集与验证集分类准确率对比（单位：%）

场景	开发集	验证集
沙滩	76.0	30.6
公交	83.0	41.7
咖啡/餐馆	78.5	61.1
汽车	92.0	57.4
市中心	86.5	79.6
森林小径	66.7	81.5
杂货铺	74.4	61.1
家	72.0	96.3
图书馆	60.6	23.1
地铁站	77.2	94.4
办公室	97.8	69.4
公园	48.4	15.7
小区	73.7	80.6
火车	36.2	63.0
电车	81.7	55.6
平均	73.6	60.7

2.5 本章小结

本章首先介绍了音频场景分类基线系统的系统结构，然后介绍了音频场景分类中常用的音频特征，并详细介绍了 MFCC 特征，给出了 MFCC 特征的提取过程。再以 EM 算法为核心，介绍了 GMM 模型的原理及 EM 算法在 GMM 模型中的应用。最后，进行了基线系统的实验。实验包括了实验环境、实验数据集介绍，系统性能评价指标以及实验结果和分析。该系统在训练集上实现的分类准确率为 73.6%，验证集上分类准确率为 60.7%。

第3章 基于卷积神经网络的音频场景分类系统

3.1 引言

卷积神经网络目前已广泛用于语音识别，计算机视觉和自然语言处理应用。尽管之前的研究者将卷积神经网络主要用于视觉识别中，但卷积神经网络也已成功应用于语音和音乐分析。由于卷积神经网络能够从高维原始数据中学习分层特征，使得卷积模型优于基于手工设计特征的常用方法，并达到与其他特征学习方法类似的水平。虽然训练时间可能会更长且结果远非突破性，但卷积神经网络可以有效地利用现有的音频数据集，即使数据集的数量可能非常有限。

本章将从研究卷积神经网络的原理出发，然后讨论将卷积神经网络应用在音频场景分类中的适用性，最后将完整设计一个基于卷积神经网络的音频场景分类系统，并使用与第二章相同的数据集进行测试与比较。分类系统主要包括了训练和测试部分，其流程设计如图 3-1。

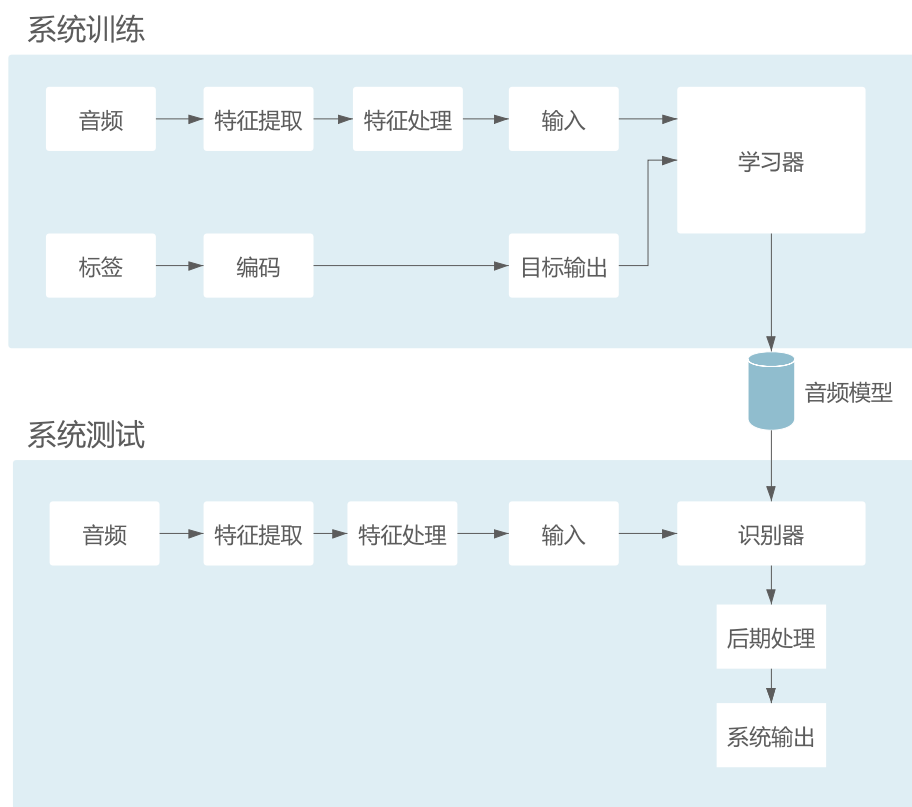


图 3-1 基于卷积神经网络的音频场景分类系统流程设计

3.2 卷积神经网络原理

一般的卷积神经网络主要由卷积层、激活函数、池化层与全连接层等组成。作为将卷积神经网络应用到音频场景分类领域的基础知识,本小节将分别研究卷积神经网络中的重要结构、原理及特点。

卷积神经网络作为一种特殊的深层前馈神经网络,具有局部连接、权值共享、子采样的结构特点。得益于这些特点,使卷积神经网络有了缩放不变性、平移不变性及旋转不变性,同时也使卷积神经网络在计算时相比一般的前馈神经网络需要更少的参数,大大提高了效率。

3.2.1 卷积层

卷积层是卷积神经网络的核心模块,可以完成大部分繁重的计算工作,其作用为提取一个区域的局部特征。卷积层执行的核心操作称作卷积(convolution),卷积为分析数学中的常用运算方式,是通过两个函数生成第三个函数的一种数学算子。在机器学习领域的应用中,卷积通常体现为在一幅图像或某种特征上滑动一个滤波器,借助这样的操作以得到一组新的特征。其中二维离散序列卷积的定义如下:

假设有图像 $X \in \mathbb{R}^{M \times N}$ 与滤波器 $W \in \mathbb{R}^{m \times n}$,且 $m \ll M$, $n \ll N$,则其卷积可以表示为:

$$y_{ij} = \sum_{u=1}^m \sum_{v=1}^n w_{uv} \cdot x_{i-u+1, j-v+1} \quad (3-1)$$

如图 3-2 为一个二维卷积的示例。

1	1	1	1	1
-1	0	-3	0	1
2	1	1	-1	0
0	-1	1	2	1
1	2	1	1	1

 \otimes

1	0	0
0	0	0
0	0	-1

 $=$

0	-2	-1
2	2	4
-1	0	0

图 3-2 二维卷积的示例

卷积运算的结果,如图 3-2 中等式右侧所示,称为特征图(feature map)。特

征图为一幅图像经过卷积运算后提取到的特征。通常为了提升卷积神经网络的特征提取能力，可以在神经网络的每层使用多个不同特征。

而应用于本文音频特征的图像处理中，为了更加充分的提取出图像的局部信息，使用三维结构的神经层，其尺寸为高度 $H_1 \times$ 宽度 $W_1 \times$ 深度 D_1 ，即 D 个 $H \times W$ 的特征图组成。此外，还需要指定的超参数有滤波器的个数 K ，滤波器的大小 F ，步长 S 以及边界填充 P 。

生成的特征图 $H_2 \times W_2 \times D_2$ 大小的计算方法如下：

$$H_2 = \frac{H - F + 2P}{S} + 1 \quad (3-2)$$

$$W_2 = \frac{W - F + 2P}{S} + 1 \quad (3-3)$$

$$D_2 = K \quad (3-4)$$

3.2.2 激活函数

一般在卷积层进行卷积运算之后，为了增强网络的表达与学习能力，会在卷积层之后加入连续非线性的激活函数（activation function）。原因在于连续非线性的激活函数可导，可以用最优化的方式来学习网络参数。

本文中使用的函数为 ReLU（Rectified Linear Unit，线性修正单元），为当前卷积神经网络中常用的函数。其定义为：

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (3-5)$$

ReLU 本质上为一种斜坡函数，优点在于计算时只用进行加法、乘法与比较操作，无梯度耗散问题，收敛快，计算上更加高效。不同于已经濒临淘汰的 sigmoid 函数，当 $x < 0$ 时，神经元的输出为 0，增加了网络的稀疏性，使 50%的神经元处于激活状态。

3.2.3 池化层

池化层（pooling layer），通常置于卷积层、激活函数之后，其对输入的特征图进行压缩，以减小图片尺寸进而简化网络复杂度；另一方面其进行特征选择，通过降低特征的数量以减少网络参数数量。

假设池化层输入的特征图组为 $X \in \mathbb{R}^{H \times W \times D}$ ，对于其中的每一个特征图 X^d ，可以将其划分为子区域 $R_{m,n}^d, 1 \leq m \leq H, 1 \leq n \leq W$ 。则池化的常见两种定义如下：

（1）最大池化（maximum pooling）

最大池化为在一个区域内寻找所有神经元的最大值，表述如式（3-6）。

$$Y_{m,n}^d = \max_{i \in R_{m,n}^d} x_i \quad (3-6)$$

其中， x_i 为子区域 $R_{m,n}^d$ 中的输入特征图组， y_i 为经过最大池化的输出特征图组。最大池化的例子如图 3-3 所示。

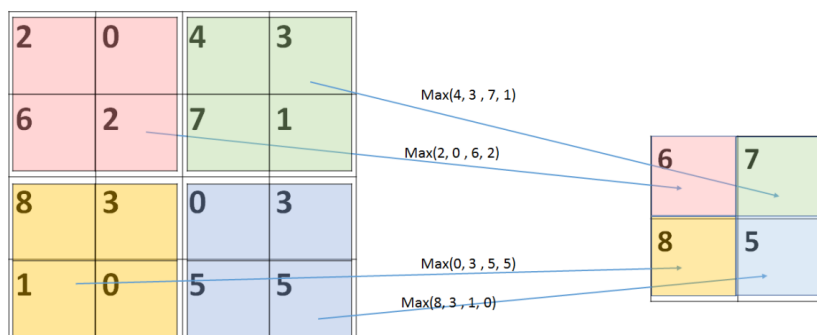


图 3-3 最大池化过程示例

（2）平均池化（mean pooling）

平均池化为在一个区域内取所有神经元的平均值，表述如式（3-7）

$$Y_{m,n}^d = \frac{1}{|R_{m,n}^d|} \sum_{i \in R_{m,n}^d} x_i \quad (3-7)$$

3.2.4 卷积神经网络的结构

典型的卷积神经网络通常由卷积层、池化层、激活函数与全连接层交叉堆叠而构成。结构如图 3-4 所示，通常卷积层与激活函数组合的个数 M 取 2~5，池化层个数 N 取 0 或 1。在经过 P 个（ P 取 1~100）连续的卷积模块后，再通过 Q 个（ Q 取 0~2）全连接层将所有特征连接，将输出值给 softmax 分类器。其中，由卷积层、池化层、激活函数与全连接层所构成的部分也成为隐藏层。

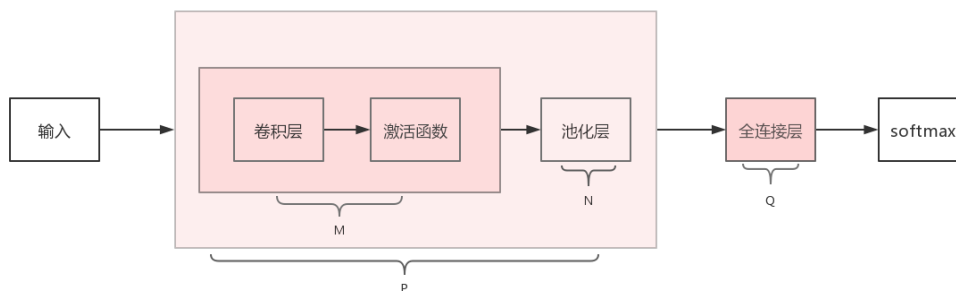


图 3-4 典型的卷积神经网络结构

3.2.5 卷积神经网络的特点

不同于全连接前馈神经网络，卷积神经网络的权重矩阵参数非常少，带来的结果是训练效率的大大提升。卷积神经网络高训练效率的核心就在于其局部连接和权重共享的特点。

(1) 局部连接

如图 3-5 (a) 所示，在全连接层中，第 l 层的每一个神经元都与下一层（即第 $l+1$ 层）中的每一个神经元相连。故总连接数为 $n^l \times n^{l+1}$ 。而在卷积层中，如图 3-5 (b) 所示，每一个神经元都只与下一层中滤波器窗口内的神经元相连，构成了局部连接网络。连接数变为了 $n^l \times m$ (m 为滤波器尺寸，且通常 $m \ll n$)，使连接数大大减小，进而提升了计算效率。

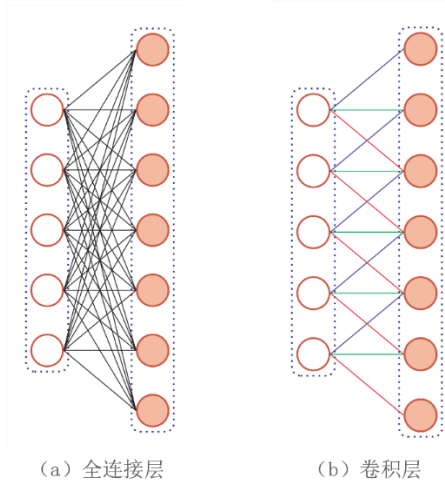


图 3-5 全连接层与卷积层

(2) 权重共享

在卷积层的卷积运算中，第 l 层的净输入 $z^{(l)}$ 可表示为：

$$z^{(l)} = w^{(l)} \otimes a^{(l-1)} + b^{(l)} \quad (3-8)$$

其中， $w^{(l)}$ 为权重向量， $a^{(l-1)}$ 为第 $l-1$ 层的激活值， $b^{(l)}$ 为偏置。

参考公式 3-8 我们可以看出，滤波器权重 $w^{(l)}$ 作为参数对于第 l 层的所有神经元都是相同的。由于滤波器上的神经元权值相同，所以网络可以并行学习。权重共享降低了网络的复杂性，特别是多维输入向量的图像可以直接输入网络这一特点避免了特征提取和分类过程中数据重建的复杂度。

由于局部连接与权重共享，导致了卷积层中的参数个数与层中神经元的数量没有关系，而只与滤波器尺寸 m 和 1 维偏置 $b^{(l)}$ 有关，共计 $m+1$ 个参数。因此神经元的增多并不会严重影响卷积层的计算效率。

3.2.6 卷积神经网络的参数学习

卷积神经网络中网络参数的训练与全连接前馈网络类似,采用误差反向传播算法。但区别于全连接前馈网络的通过计算每一层误差项 δ 进行反向传播来计算每层参数梯度,卷积神经网络中的参数只有卷积核与偏置,故只需计算卷积层中参数即可。

假设第 l 层为卷积层,其前一层 $l-1$ 层的输入特征图为 $X^{(l-1)} \in \mathbb{R}^{H \times W \times D}$ 。则经过 $l-1$ 层卷积运算后,第 l 层的特征图净输入为 $Z^{(l)} \in \mathbb{R}^{H' \times W' \times K}$ 。其中,第 l 层的第 k 个($1 \leq k \leq K$)特征图的净输入为:

$$Z^{(l,k)} = \sum_{d=1}^D W^{(l,k,d)} \otimes X^{(l-1,d)} + b^{(l,k)} \quad (3-9)$$

其中, $W^{(l,k,d)}$ 为滤波器权重, $b^{(l,k)}$ 为偏置。第 l 层共有卷积核 $K \times D$ 个与偏置 K 个,其梯度可以使用链式法则计算。

损失函数关于第 l 层的滤波器权重 $W^{(l,k,d)}$ 的偏导数为:

$$\begin{aligned} \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial W^{(l,k,d)}} &= \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial Z^{(l,k)}} \otimes X^{(l-1,d)} \\ &= \delta^{(l,k)} \otimes X^{(l-1,d)} \end{aligned} \quad (3-10)$$

其中, $\delta^{(l,k)}$ 为损失函数关于第 l 层的第 k 个特征图净输入 $Z^{(l,k)}$ 的偏导数。

同理,损失函数关于第 l 层第 k 个偏置 $b^{(l,k)}$ 的偏导数为:

$$\frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial b^{(l,k)}} = \sum_{i,j} [\delta^{(l,k)}]_{i,j} \quad (3-11)$$

由此可知,每层参数的梯度计算依赖于其所在层的误差项 $\delta^{(l,k)}$ 。而在卷积层和池化层中,误差项的计算又有区别,因此需要分别计算。

(1) 卷积层的误差项计算

假设第 $l+1$ 层为卷积层,特征图净输入 $Z^{(l+1)} \in \mathbb{R}^{H' \times W' \times K}$,类似于式(3-9),第 l 层的第 k 个($1 \leq k \leq K$)特征图的净输入 $Z^{(l+1,k)}$ 可表示为:

$$Z^{(l+1,k)} = \sum_{d=1}^D W^{(l+1,k,d)} \otimes X^{(l-1,d)} + b^{(l+1,k)} \quad (3-12)$$

则第 l 层的第 d 个特征的误差项 $\delta^{(l,d)}$ 可借助式(3-12)推导:

$$\begin{aligned} \delta^{(l,d)} &\triangleq \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial Z^{(l,d)}} \\ &= \frac{\partial X^{(l,d)}}{\partial Z^{(l,d)}} \cdot \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial X^{(l,d)}} \end{aligned}$$

$$\begin{aligned}
 &= f'_l(Z^l) \odot \sum_{k=1}^K \left(\text{rot180}(W^{(l+1,k,d)}) \widetilde{\otimes} \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial Z^{(l+1,k)}} \right) \\
 &= f'_l(Z^l) \odot \sum_{k=1}^K (\text{rot180}(W^{(l+1,k,d)}) \widetilde{\otimes} \delta^{(l+1,k)}) \quad (3-13)
 \end{aligned}$$

其中, $f'_l(\cdot)$ 为第 l 层激活函数的导数, $\widetilde{\otimes}$ 为宽卷积。

(2) 池化层的误差项计算

由于池化层进行的是下采样 (subsampling) 操作, 故当 $l+1$ 层为池化层时, 第 $l+1$ 层神经元的误差项 δ 对应于其上一层特征图的一个区域。根据连式法则, 要求得第 l 层的一个特征图对应的误差项 $\delta^{(l,k)}$, 只需将其下一层对应的特征图误差项 $\delta^{(l+1,k)}$ 进行上采样 (upsampling) 操作, 再与第 l 层特征图激活值的偏导数逐元素相乘即可。

第 l 层的特征图误差项 $\delta^{(l,k)}$ 的推导如下:

$$\begin{aligned}
 \delta^{(l,k)} &\triangleq \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial Z^{(l,k)}} \\
 &= \frac{\partial X^{(l,k)}}{\partial Z^{(l,k)}} \cdot \frac{\partial Z^{(l+1,k)}}{\partial X^{(l,k)}} \cdot \frac{\partial \mathcal{L}(Y, \hat{Y})}{\partial Z^{(l+1,k)}} \\
 &= f'_l(Z^l) \odot \text{up}(\delta^{(l+1,k)}) \quad (3-14)
 \end{aligned}$$

其中, up 为上采样函数。

3.3 卷积神经网络在音频场景分类中的适用性

此前卷积神经网络已成功用于各种音频处理任务, 如语音识别, 音乐分析、音频事件检测等。而在音频场景分类中使用卷积神经网络作为分类器有以下几点重要原因:

- (1) 卷积神经网络可以直接处理时频联合数据;
- (2) 卷积神经网络可以用自动学习的特征代替手动设计的特征进行分类, 使分类更为高效;
- (3) 卷积神经网络具有良好的捕捉周期性时频特征的能力。

本小节将从卷积神经网络的各层结构出发, 研究卷积神经网络在音频场景分类中应用的适用性及各层的具体作用。

3.3.1 卷积层的应用

复杂音频场景包含有容易辨别的时频重复特征, 如发动机噪音和电话铃声。

这些特征称为局部模式，表现为频率和时间上能量的反复集中。例如，发动机噪声的特征在于跨越时间轴的局部模式，而铃声可以呈现跨越频率轴的重复特征。这种局部模式可以通过输入 I 和一组滤波器权重 W 之间的卷积运算来表示，它产生输出 O ：

$$O[i, j] = I[i, j] \otimes W[i, j] = \sum_{u=-\infty}^{\infty} \sum_{v=-\infty}^{\infty} i[u, v] W[i - u, j - v] \quad (3-15)$$

其中 i 和 j 是 I 的行和列索引，而 u 和 v 是滤波器权重 W 的行和列索引。该卷积运算为离散二维卷积运算，在卷积层中进行。而卷积层对输入数据的应用如图3-6所示。

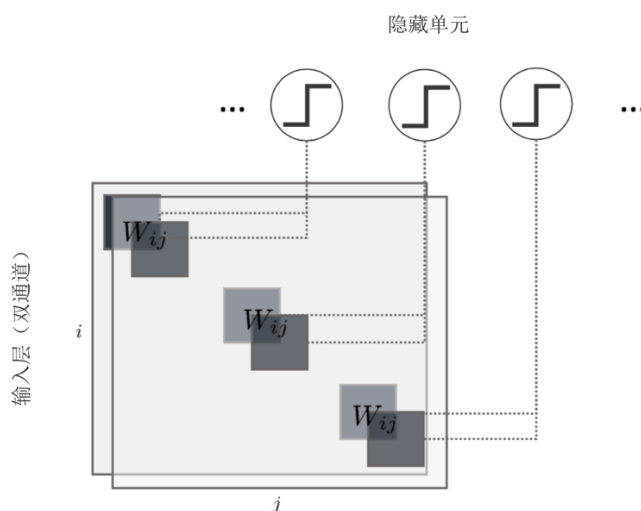


图 3-6 卷积层对输入数据的应用

其中：

- (1) 每个隐藏单元与输入层中坐标 i, j 处的感受野之间为局部连接；
- (2) 输入的多个通道在感受野和隐藏单元之间保持相同的关系；
- (3) 使用相同的滤波器权重 W ，即构成权重共享，以捕获输入数据中的类似的重复特征。

3.3.2 池化层的应用

池化层置于每个卷积层的输出之后以降低其分辨率。最简单的池化操作是最大池化，其中池化层输入中的值块被替换为其单个最大值。应用于音频场景分类任务时，池化操作可将时频特征中的微小变化过滤掉。

例如，以特定频率为中心的相同局部模式（如引擎噪声）可能只在一个记录与另一个记录间略微变化。池化过程允许降低频率或时间分辨率，将分类的重点

转移到局部模式上。

3.3.3 全连接层的应用

卷积层和池化层可以按顺序复制，以增加深度进而提取出更高级别的输入特征。在末端通过全连接层和 softmax 层实现分类。全连接层的输入为最后一层卷积层或池化层的输出。全连接层的作用为将输入分类为输出的音频场景之一。

每个输入片段（原始的全谱图）都是独立分类的，借助 softmax 函数识别最可能的音频场景 \hat{y} ，频谱图的分类根据多数投票进行。用于卷积神经网络训练的目标函数（即用于模型参数 W 和 b 的优化）为在 N 个输入样本上最小化目标 y 和预测 \hat{y} 之间的损失函数 l ：

$$l(\theta = \{W, b\}, N) = -\frac{1}{N} \sum_{n=1}^N y_n \log \hat{y}_n + (1 - y_n) \log(1 - \hat{y}_n) \quad (3-16)$$

3.4 系统设计

训练卷积神经网络需要做出很多关于架构（如输入数据的格式，卷积层的数量和大小，池化层的数量，滤波器维度）和学习超参数（学习率、动量、批次大小、丢失率）根据图 3-1 的系统设计流程图，本小节给出系统的设计细节。

由于培训完整模型所需的时间很长，详尽评估所有潜在组合是不可能的。因此，综合效能最佳的模型的选择必须基于对最重要因素（层数/滤波器，滤波器形状，学习速率，丢失率）执行的有限验证（通常设置为 10-20 个迭代周期）。故设计阶段应考虑不同的训练参数作为对比，在有限的迭代周期内判定参数的适用性。而考虑中核心的要素即为分类平均准确率和单位迭代周期长度。最终得出的系统要在分类准确率尽可能高的同时训练时间在可控制范围内。

3.4.1 特征图与预处理

本章系统选择的特征图为 log-mel 谱图。MFCC 的优势在第 2 章已经阐述过，不再赘述。为了计算它，在 40 毫秒音频的窗口上应用短时傅立叶变换（STFT），并加 50% 的重叠和 Hamming 窗口。然后，对每个窗的绝对值进行平方，并应用 60 波段的梅尔滤波器组。最后，进行梅尔能量的对数转换以获得 log-mel 谱图。使用 librosa 库实现了整个特征提取过程。

在提取过程之后，通过减去其平均值并除以其标准偏差来标准化每个窗，两

者都在每个折叠的整个训练集上计算。然后将归一化的谱图分成更短的谱图，下文中称之为序列。与用于 STFT 的帧不同，此处选择序列不重叠。在该过程结束时，卷积神经网络的输入是矩阵，可以将其视为单声道图像。

3.4.2 卷积神经网络架构

系统核心的卷积神经网络模块构成如图 3-7 所示。

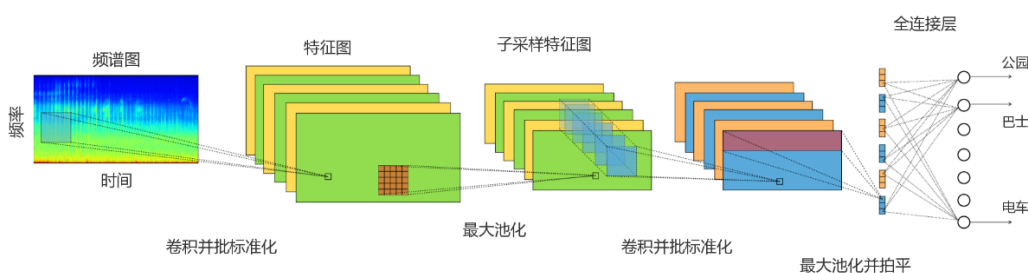


图 3-7 卷积神经网络模块构成

第一层在输入的频谱图上执行卷积。由于卷积滤波器的大小体现为局部感受野的大小，不同尺寸的感受野大小决定其提取特征的粒度，故实验中会安排不同尺寸的滤波器大小以进行对比。此外，滤波器的数量也会影响到特征的分析角度，因此实验过程中也会安排不同的滤波器数量进行对比。滤波器数量变多会增加特征分析的角度，但同时也会使计算量上升，而且过多的滤波器可能导致参数冗余。通常，滤波器的数量取 2^n 。在卷积过程执行完成后，使用最大池化层对所获得的特征图进行子采样。

第二卷积层与第一卷积层基本相同，区别在于第二层使用了更多数量的内核（取第一层的 2 倍），以便在更高级别上表示特征。然后，针对时间轴的“破坏”执行第二次和最后一次子采样。因此，依然使用最大池化层，其在整个序列长度上操作。卷积层中用于内核的激活函数是 ReLU。

最终，由于分类涉及 15 个不同的类，最后一层是由 15 个完全连接的神经元组成的 softmax 层，它对网络的输出结果进行归一化，使系统输出分类结果。假设 y_i 是上一层神经元 i 的输出，则 y_i 可以定义为：

$$y_i = \text{softmax}(x_i) = \frac{\exp(x_i)}{\sum_{j=1}^N (\exp(x_j))} \quad (3-17)$$

其中， N 为类别总数， x_i 是非线性输入， y_i 是输入序列属于第 i 类的预测得分。因此，总输出是包含与每个类相关联的所有类别预测分数的向量 \mathbf{y} 。如果 y_j 为这些分数中的最高分，则输入序列的预测类将是第 j 类。

3.4.3 批标准化

批标准化（Batch Normalization, BN）是一种解决内部协变量转换（Internal Covariate Shift）问题（Shimoidara et al., 2000）的技术。换句话说，在卷积神经网络训练过程中，由于网络中参数不断变化导致内部节点数据分布发生变化，这种变化过程称为内部协变量转换。由于输入数据发生了变化，上层神经网络需要不断调整参数以应对这种变化，导致的后果就是卷积神经网络学习的速度降低。此外，由于网络在训练时容易进入梯度饱和区，造成网络参数更新减慢，进而影响收敛速度。

为了解决内部协变量转换造成的网络学习速度与收敛速度下降的问题，通过引入批标准化来改变输入数据的分布，批标准化层将线性变换 $\text{BN}_{\beta,\gamma}$ 应用于其输入 x ，如式（3-18）所示。

$$\text{BN}_{\beta,\gamma} = \frac{\gamma}{\sqrt{\text{Var}(x) + \epsilon}} \cdot x + \left(\beta - \frac{\gamma \cdot E[x]}{\sqrt{\text{Var}(x) + \epsilon}} \right) \quad (3-18)$$

其中 $E[x]$ 和 $\text{Var}(x)$ 是批标准化层输入的平均值和方差。此外， γ 和 β 表示在训练期间将学习的变换参数。由于使用批标准化来规范化内核输出，因此每个内核都有一个 γ 和一个 β 。

通过使用批标准化，使得输入特征分布之间具有相同的均值和方差，去除了特征之间的相关性。虽然增加了模型的复杂性，但减缓了内部协变量转换过程，使训练收敛时间大大减少，学习进度加快。

3.4.4 Dropout 机制

在卷积神经网络模型中，如果参数过多且训练样本过少，经常会发生过拟合（overfitting）现象。过拟合具体体现为：模型在训练阶段损失函数较小，预测准确率较高，但在测试阶段损失函数较大，准确率偏低。

为了解决过拟合现象，通常使用模型集成方法，即多个模型组合训练。而模型集成带来的问题就是训练过程中成本过大，过于费时，且测试阶段同样要耗费大量时间。

Hinton et al.（2012）在其论文中首次提出了 Dropout 机制，Dropout 通过在训练过程中，通过忽略固定数量（通常为二分之一）的特征检测器，可以有效缓解过拟合的发生，使神经网络训练加速。

假设神经网络的输出为 x ，输出为 y 。则标准神经网络的流程为将 x 通过网络前向传播，之后将误差反向传播，更新参数让网络学习。而 Dropout 引入后，

会首先随机隐藏网络中一半的隐藏神经元，如图 3-8，而输入输出神经元保持不变。然后依然将输入 x 通过修改后的网络进行前向传播，把得到的损失结果通过网络进行反向传播。经过一批训练样本后，在没有删除的神经元上按梯度下降法更新参数 W, b 。最后，继续重复该过程。

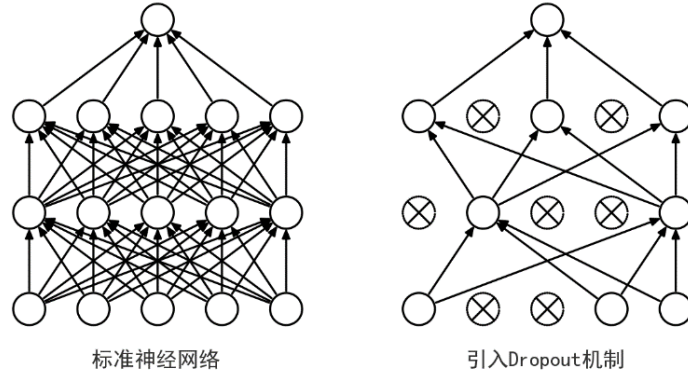


图 3-8 Dropout 机制

由于随机删掉一半隐藏神经元导致网络结构发生变化，整个 Dropout 过程相当于对多个不同的神经网络取平均。通过引入 Dropout 机制产生的平均作用，抵消了网络中部分相反的拟合，进而达到类似模型集成减少过拟合的效果。此外，由于 Dropout 过程可能会造成某两个神经元不会总是同时出现的情况，可以减少神经元之间复杂的共适应关系。这样一来，权重更新不会再依赖于网络中的某些固定结构，从而使整个网络的学习更具有健壮性。

3.4.5 网络模型优化算法

传统卷积神经网络的模型训练和参数求解常采用随机梯度下降类优化算法。而随着近年来深度学习的发展，诞生了新的一批网络优化算法。本章中采用的网络优化算法优化为 Adam optimizer (Adam et al., 2014)，此算法基于一阶梯度的随机目标函数优化算法。该方法实现简单，计算效率高，具有较小的存储器要求，对于梯度的对角重新缩放具有不变性，并且适用于数据或参数数量较大的情况。下面简要介绍该优化算法：

Adam 算法的输入参数包括：学习率 α 、指数衰减率 $\beta_1, \beta_2 \in [0, 1)$ 、为了维持数值稳定性的常数 ϵ 、带有参数 θ 的随机目标函数 $f(\theta)$ 以及初始参数向量 θ_0 。

首先需要进行的是一阶矩向量、二阶矩向量、时间步的初始化：

$$m_0 \leftarrow 0 \quad (3-19)$$

$$v_0 \leftarrow 0 \quad (3-20)$$

$$t \leftarrow 0 \quad (3-21)$$

然后，在参数 θ 没有收敛时，循环迭代地更新各个部分。先对时间步 t 加一：

$$t \leftarrow t + 1 \quad (3-22)$$

接下来，更新目标函数在该时间步上对参数 θ 的梯度：

$$g_t \leftarrow \nabla_{\theta} f_t(\theta_{t-1}) \quad (3-23)$$

接着更新偏差的一阶矩估计与二阶原始矩估计：

$$m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \quad (3-24)$$

$$v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \quad (3-25)$$

再计算偏差修正的一阶矩估计和二阶矩估计：

$$\hat{m} \leftarrow m_t / (1 - \beta_1^t) \quad (3-26)$$

$$\hat{v} \leftarrow v_t / (1 - \beta_2^t) \quad (3-27)$$

最后，用以上计算出的值更新模型的参数 θ ：

$$\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m} / (\sqrt{\hat{v}} + \epsilon) \quad (3-28)$$

本文卷积神经网络训练部分采用 Tensorflow 作为后端，且网络优化采用其默认参数： $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$ 。

3.4.6 模型训练

系统训练包括两个阶段。第一阶段称为非完全训练，首先将整个训练数据分成两个子集：一个用于训练，一个用于测试。每个迭代周期都会将训练频谱图收集到分类特征列表中，以便在序列分类之前随机改变它们并对其进行时移，这样做是为了增加输入可变性。然后，每隔固定迭代周期（本文取 5）检查训练和测试集上的分段性能。检查后，如果分段验证分数得到改善，将保存网络参数。最后，如果在足够长的迭代周期之后（本文取 15）没有记录到任何改进，会停止训练。通过这种方法，会使分段验证性能容易饱和。这意味着得分开始在固定的稳定值附近振荡。当发生这种情况时，说明系统已经收敛，因此可以进入第二阶段，即完全训练阶段。

在完全训练阶段，在所有训练数据上重新训练网络。通过观察非完全训练期间分段验证准确度的收敛时间来选择完全训练的迭代周期。以这种方式训练的模型将达到收敛状态且不会过度拟合，并充分利用所有可用的训练数据。

3.5 实验结果与分析

3.5.1 实验环境与数据集处理

本章实验使用的数据集与第二章相同，此处不再介绍。对于每个音频文件，使用 40ms 的分析帧和 50% 帧移的窗口在 40 个频带中提取对数梅尔能量。卷积神经网络结构如 3.4.2 节中介绍的，由两个 CNN 层和一个全连接层组成，并使用大小为 40×500 的输入，相当于要分类段的全长。此外，每一 CNN 层后置一个 Dropout 层，Dropout 比例为 30%。训练网络时使用 Adam optimizer 优化算法，学习率为 0.001，批大小为 16。完整的训练周期为 100 个迭代周期，系统参数调整阶段训练周期为 15。

数据集处理方面不同于第 2 章中的 GMM 模型，由于卷积神经网络的训练时间远远大于 GMM，且卷积神经网络模型对数据集的大小相对不敏感，因此本章训练阶段不再使用 4 折交叉验证法。替代方法使用由大约 30% 的开发集中数据组成的测试集来完成，使开发集中训练数据和测试数据互斥，并且两组数据都具有每个音频场景的样本，类别比例保持相等。在每个迭代周期之后在测试集上评估模型性能，选择性能最佳的模型。

3.5.2 评价指标

音频场景分类准确率依然为本系统评价指标之一，在此不再赘述。此外，为了准确分析音频场景分类系统在各场景下的差异，以便后期优化系统，引入了混淆矩阵（Confusion Matrix）作为评价指标之一。混淆矩阵是判定分类精度的一种方法。在混淆矩阵中，其列坐标代表预测的值，行坐标代表实际的值。通过引入混淆矩阵可以清晰的判断出系统在哪些场景类别上存在混淆及混淆程度。

3.5.3 滤波器参数调整

根据 3.4.2 节中讨论的参数调整方案，现进行初步试验以得出系统最佳参数。参考依据为分类准确度和单位迭代周期。系统均使用开发集数据进行训练与测试，以确保以尽可能小的代价体现系统性能，最后整体实验阶段可能需要比参数调整阶段更长的单位迭代周期。

（1）滤波器数量的调整

滤波器数量的调整策略见表 3-1。

表 3-1 滤波器数量调整策略

	CNN_1 层	CNN_2 层
方案 a1	16	32
方案 a2	32	64
方案 a3	64	128
方案 a4	128	256
方案 a5	256	512

我们分别设计了 5 组方案以对照,滤波器数量呈倍数增加,且整个网络中其他参数固定。鉴于之后要调整滤波器尺寸,从参数较少的设计出发,滤波器尺寸暂定取 3*3。

表 3-2 不同滤波器数量的分类结果

	准确率	单位迭代周期
方案 a1	45.7%	0:00:15
方案 a2	61.7%	0:00:29
方案 a3	72.8%	0:01:00
方案 a4	78.7%	0:02:13
方案 a5	77.9%	0:05:23

根据表 3-2 实验结果可以发现,方案 a4 可以达到准确度与训练时间的最佳平衡。而滤波器数量的无限制增加不仅没有提高准确率,且大大增加了单位迭代周期。因此,通过本轮实验决定最终训练模型 CNN_1 层滤波器数量为 128,CNN_2 层滤波器数量为 256。

(2) 滤波器尺寸的调整

滤波器尺寸的方案如表 3-3 所示为 4 种,不同的滤波器尺寸决定了局部感受野的大小,也即确定了要提取特征的层次。通过调整参数,找到与模型最匹配的特征粒度,从而使分类准确率最大化。

表 3-3 滤波器尺寸调整策略

	CNN_1 层	CNN_2 层
方案 a4	3*3	3*3
方案 b1	5*5	5*5
方案 b2	7*7	7*7
方案 b3	9*9	9*9

第二轮参数调整的分类结果如表 3-4 所示。

表 3-4 不同滤波器尺寸的分类结果

	准确率	单位迭代周期
方案 a4	78.7%	0:02:13
方案 b1	85.9%	0:03:01
方案 b2	89.1%	0:04:15
方案 b3	89.3%	0:05:50

观察表 3-4 我们可以发现,当滤波器尺寸设置为 7*7 与 9*9 时,系统都能达到相对高的分类准确率,且方案 b3 分类性能略微优于 b2。但考虑到两方案分类准确率差距甚微,且方案 b3 由于滤波器尺寸更大致使参数增加进而导致单位迭代周期有明显的增多,在最后的整体实验阶段会导致训练时间大幅增加。因此综合考虑分类准确率与单位迭代周期,本文决定采用效能比最佳的方案 b2,即 7*7 的滤波器尺寸。在确定了系统关键性因素后,最终的系统网络参数如表 3-5 所示。

表 3-5 卷积神经网络参数

层 (类型)	输出尺寸	参数 (个)
卷积层_1	(40,500,128)	6400
批标准化_1	(40,500,128)	160
激活函数_1	(40,500,128)	0
最大池化_1	(8,100, 128)	0
Dropout_1	(8,100, 128)	0
卷积层_2	(8,100,256)	1605888
批标准化_2	(8,100,256)	32
激活函数_2	(8,100,256)	0
最大池化_2	(2,1,256)	0
Dropout_2	(2,1,256)	0
拍平	512	0
Dense(全连接层)_1	100	51300
Dropout_3	100	0
Dense(全连接层)_2	15	1515
参数总计: 1,665,295		
可训练参数: 1,665,199		
不可训练参数: 96		

3.5.4 结果分析

根据 3.5.3 节的参数，在两个数据集上训练得到的结果分类准确率结果如表 3-6 所示。

表 3-6 音频场景分类准确率结果（单位：%）

音频场景	开发集	验证集
沙滩	92.0	41.7
公交	94.6	26.9
咖啡/餐馆	89.1	55.6
汽车	99.0	70.4
市中心	100.0	91.7
森林小径	99.7	92.6
杂货铺	88.8	49.1
家	95.2	64.8
图书馆	89.1	37.0
地铁站	96.5	91.7
办公室	100.0	86.1
公园	91.3	21.3
小区	86.9	56.5
火车	87.8	60.2
电车	90.4	42.6
平均	93.4	59.2

分析表 3-6 我们可以发现，系统最终在开发集上的分类准确率为 93.4%，且各类的识别准确率差异不大，皆处在很高的水平上。这说明卷积神经网络模型有效的利用了现有数据集。验证集的分类准确率为 59.2%，说明系统泛化能力一般。而造成这种结果的原因很大一部分在于卷积层中的参数达百万之多，对训练的数据产生了记忆效应，因此在开发集上表现的很好，但在验证集上大打折扣。因此，后期对网络优化阶段应从改变网络结构入手，即不需要单纯的依靠参数的设置来提升系统的分类准确率。

通过比较系统在验证集中各个场景的分类准确率，可以发现系统在图书馆、公交、公园场景中的分类准确率不佳，而在森林小径、市中心与地铁站的分类准确率皆超过了 90%。因此，需要进一步了解导致系统在部分场景中分类准确率不

高的原因。将系统在验证集上的分类结果进行归纳，整理成如表 3-7 所示的混淆矩阵。其中行坐标为音频文件的真实场景标签，列坐标为系统预测的场景。

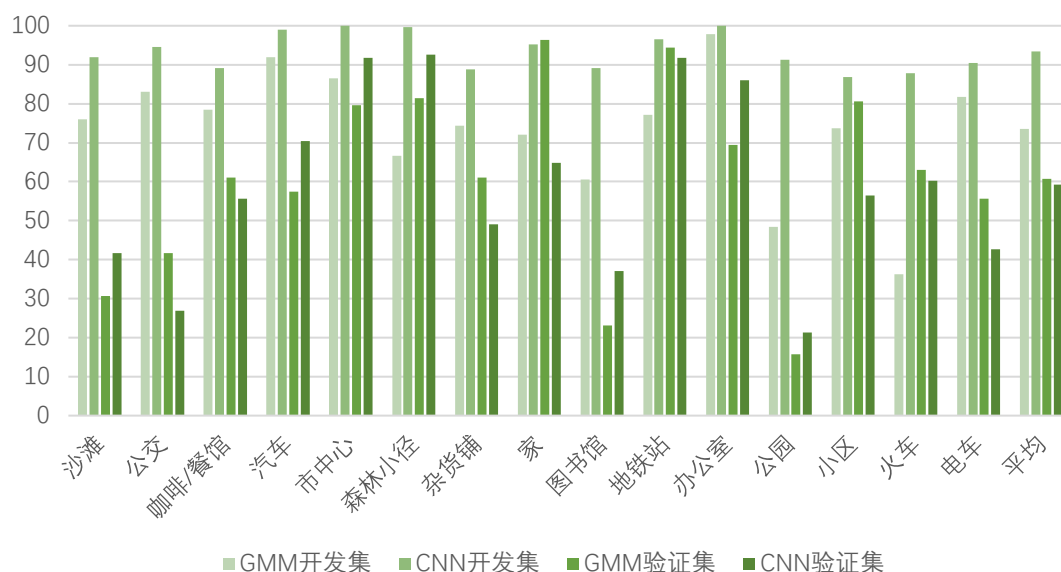
表 3-7 系统在验证集上分类结果的混淆矩阵

	沙滩	公交	咖啡	汽车	市中	森林	杂货	家	图书	地铁	办公	公园	小区	火车	电车
沙滩	45	0	27	0	1	1	4	0	4	0	0	5	21	0	0
公交	2	29	17	19	6	0	4	2	0	5	0	0	1	9	14
咖啡	1	0	60	0	0	1	6	3	3	26	0	0	0	0	8
汽车	0	0	0	76	0	0	0	0	0	0	0	0	3	11	18
市中	0	0	0	0	99	3	1	0	0	0	0	3	2	0	0
森林	1	0	0	0	0	100	0	0	0	0	1	5	1	0	0
杂货	7	0	23	0	0	15	53	1	1	7	0	0	0	0	1
家	4	0	0	0	0	5	0	70	11	0	18	0	0	0	0
图书	4	0	0	0	0	5	3	8	40	13	35	0	0	0	0
地铁	0	0	0	0	6	3	0	0	0	99	0	0	0	0	0
办公	1	0	0	0	0	8	0	4	2	0	93	0	0	0	0
公园	3	0	0	0	6	20	4	0	0	0	0	23	52	0	0
小区	1	0	0	0	15	27	0	0	1	0	0	3	61	0	0
火车	3	5	0	10	4	0	0	0	0	0	0	0	2	65	19
电车	3	5	25	5	2	0	0	0	0	12	0	3	0	11	46

通过观察表 3-7 的混淆矩阵可以更深入地了解哪些场景被错误分类。可以发现混淆度较高的情况存在于公园-小区、办公室-图书馆、公交-电车、森林-小区等场景对之间，这也与听觉常识相吻合。这表明本章设计的系统在分类过程中更多地依赖于背景噪声而不是基于具体的音频事件。此外，由于预处理过程中将双通道音频压缩成单通道，人为的减少了可分类因素导致系统不能有效利用空间信息，也可能是部分相似场景下混淆度较高的原因。

最后，第 2 章与第 3 章系统的对比如表 3-8 所示。

表 3-8 GMM 系统与 CNN 系统的分类性能比较



从表 3-8 中我们可以看出，在开发集上卷积神经网络模型的分类性能大幅优于 GMM 模型，但在验证集中两系统的分类性能各有优劣。在开发集中分类性能更好这一结果说明了卷积神经网络对有限数据集的学习能力高于一般的 GMM 模型。但是受制于对原始数据集的依赖，本章卷积神经网络模型的推广性能一般。

3.6 本章小结

本章首先给出了基于卷积神经网络的音频场景分类系统的整体结构，然后从卷积神经网络的结构出发，研究了其特点以及参数学习的方法。接着分别探讨了将卷积神经网络各层应用到音频场景分类中的可行性，再从系统设计的角度探讨了分类系统的实现细节。最后，先进行了参数调整找到了系统最优分类参数，然后进行了完整实验，验证了将卷积神经网络应用在音频场景分类中的可行性。其中系统在训练集上实现的平均分类准确率为 93.4%，验证集上平均分类准确率为 59.2%。在最后还与第二章的基线系统进行了对比，结果表明本章系统在开发集上学习能力更强，但存在泛化能力不佳的问题，且系统没有有效利用到音频文件中的空间信息。

第4章 音频场景分类系统的改进

4.1 引言

在上一章中通过实验验证了卷积神经网络在音频场景分类中应用的可行性，并在开发集中取得了大幅优于基线系统的效果。为了解决之前在验证集中出现的部分场景对混淆及泛化能力不强的问题，本章将尝试从音频处理和网络结构优化两方面出发，以进一步提升音频场景分类系统在各种数据集上的分类准确率。

在第2章与第3章的系统中，特征提取部分一直采用的是 MFCC 特征。尽管 MFCC 可以用十几个系数简洁地描述特征，但是，能否改进 MFCC 使其能在诸如图书馆这样的具有明显空间特征的场景中更好的胜任音频场景分类任务是本章要研究的一个方面。另外，上一章中的卷积神经网络结构在训练过程中也存在过于依赖训练参数的问题。目前得以广泛应用的性能良好的卷积神经网络框架有很多，能否将其结构稍加修改利用于音频场景分类之中以提高系统性能也是本章的一个研究重点。

基于以上的讨论，本章将先从音频处理、特征提取方面出发，再研究改进的卷积神经网络结构设计。此外，本章还将对多个模型的集成进行探讨。最后，将改进后的系统应用于实验中，以检测最终的分类准确率，并与前两章进行对比。

4.2 音频处理

本节介绍了两种方法处理输入的音频，即双耳表示法和谐波冲击源分离法。下面给出了每种方法的详细解释，以及提取对应 MFCC 频谱图的例子。

4.2.1 双耳表示法

尽管在录音时使用立体声设备录制很常见，但通常在音频处理时都会在处理之前对信号求平均来使其成为单声道。固然使用单声道便于处理且易于特征提取，然而这样会丢失不少空间信息。如果重要的音频信息仅在其中一个通道中捕获良好，则可能会出现这个问题。因为将双通道平均到一个通道后会降低信噪比，这样两个声道的差别不容易被体现出来，因此在分类过程中极易造成混淆。分别对两个通道进行分析可以缓解这个问题。鉴于使用双耳表示法在先前 DCASE 挑战中取得的优异结果 (Eghbal-Zadeh et al., 2016)，本文决定在特征处理过程中使用 LR

表示法（Left、Right，即左右）和 MS 表示法（Mid、Side，即中间一侧面）。

LR 表示法代表的是常规立体声录音中的左声道与右声道。例如，汽车在麦克风前经过，声音从 L 声道移动到 R 声道或从 R 声道移动到 L 声道，这种在单声道中只体现为幅度变化。通过引入 LR 表示法，可以体现出音源在空间中的移动。应用在本章中仅将源音频文件的左右声道分离即可。

而 MS 表示法则强调到达立体声麦克风的每一侧的声音之间的时间差。MS 表示法通过对立体声输入两个通道的波形分别进行求和与求差以获取最终结果。其具体操作方法如图 4-1 所示。

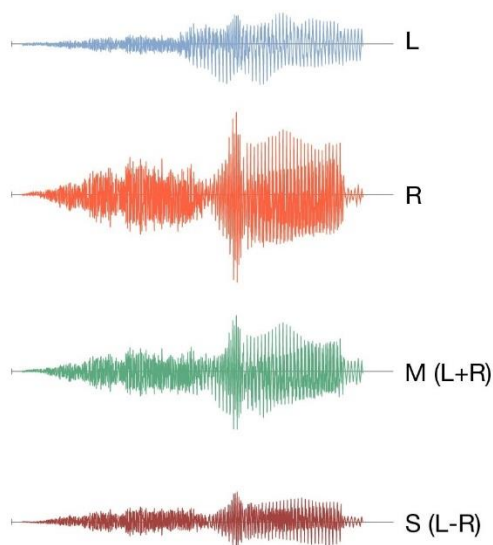


图 4-1 MS 表示法的具体操作方法

其中，Mid 通道定义为 $L + R$ ，Side 通道定义为 $L - R$ ，即两个通道之间的差异。

分别将对以上四种表示法提取 Mel 频谱图，如图 4-2 所示。

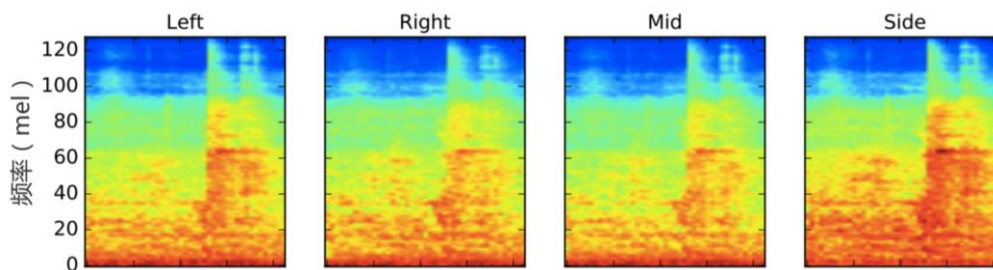


图 4-2 双耳表示法的 MFCC 频谱图

对于 LR 谱图和 MS 谱图，分别将其输入 4.3 节的卷积神经网络中，作为分类特征之一。

4.2.2 谐波冲击源分离法

声音一般可分为两种类型：谐波和冲击声。如 Ono et al. (2008) 的论文中所述, 在传统的研究工作中, 谐波冲击源分离 (Harmonic Percussive Source Separation, HPSS) 算法是在音乐信号处理的背景下提出的, 目标是将输入音频信号分解为由所有谐波和由所有冲击源组成的信号。为了解决音频场景分类中泛化性差且依赖于学习数据的问题, 本节从音乐信号处理中借鉴了该方法, 以尝试提高系统分类性能。下面给出 HPSS 算法的步骤。

假设输入离散输入音频信号 $x \in \mathbb{R}$ 。算法应计算出谐波分量信号 x_h 和冲击源分量信号 x_p , 使得 $x = x_h + x_p$ 。

首先, x 的 STFT (短时傅里叶变换) \mathcal{X} 可以表示为:

$$\mathcal{X}(t, k) = \sum_{n=0}^{N-1} x(n + tH)w(n) \exp\left(\frac{-2\pi i k n}{N}\right) \quad (4-1)$$

其中, $0 \leq t \leq M-1$, $0 \leq k \leq K$; T 为帧数, N 是傅里叶变换的帧大小和长度; $0 \leq w(n) \leq N-1$ 为窗函数, H 为帧移。

输入 X 的功率谱图 Y 可以由 \mathcal{X} 求出:

$$Y(t, k) = |\mathcal{X}(t, k)|^2 \quad (4-2)$$

接下来, 通过对 Y 进行中值滤波来计算谐波增强谱图 \tilde{Y}_h 和冲击源增强谱图 \tilde{Y}_p 。假设 $A = \{a_n \in \mathbb{R} | 0 \leq n \leq N-1\}$ 为一系列实数组成的集合, N 为集合中的实数个数, 且对于 $0 \leq n \leq N-1$ 时满足 $a_{n+1} > a_n$, 则对 A 的中值滤波定义为:

$$\text{median}(A) = \begin{cases} \frac{a_{N-1}}{2}, & N \text{ 为奇数} \\ \frac{a_{\frac{N}{2}} + a_{\frac{N}{2}-1}}{2}, & \text{其它} \end{cases} \quad (4-3)$$

接着, 根据中值滤波的定义, 通过对 Y 分别进行一次水平和一次垂直方向的中值滤波, 可以得到谐波增强谱图 \tilde{Y}_h 和冲击源增强谱图 \tilde{Y}_p :

$$\tilde{Y}_h(t, k) = \text{median}\left(Y(t - l_h, k), \dots, Y(t + l_h, k)\right) \quad (4-4)$$

$$\tilde{Y}_p(t, k) = \text{median}\left(Y(t, k - l_p), \dots, Y(t, k + l_p)\right) \quad (4-5)$$

其中, $l_h, l_p \in \mathbb{N}$, $2l_h + 1$ 与 $2l_p + 1$ 为滤波器长度。

然后, 引入一个变量 β , 满足 $\beta \in \mathbb{R}$ 且 $\beta > 1$, 称为分离因子。在满足 $\tilde{Y}_h(t, k) / \tilde{Y}_p(t, k) > \beta$ 或 $\tilde{Y}_p(t, k) / \tilde{Y}_h(t, k) > \beta$ 时, 原始输入信号 $X(t, k)$ 可以直观的判断为谐

波或冲击源分量。通过这个法则，可以定义二进制掩码 M_h 和 M_p ：

$$M_h(t, k) = \frac{\tilde{Y}_h(t, k)}{\tilde{Y}_p(t, k) + \epsilon} > \beta \quad (4-6)$$

$$M_p(t, k) = \frac{\tilde{Y}_p(t, k)}{\tilde{Y}_h(t, k) + \epsilon} \geq \beta \quad (4-7)$$

其中 ϵ 是一个小常数以避免发生除零错误，运算符 \geq 和 $>$ 保证了输出为二进制结果 0 和 1。将掩码 M_h 和 M_p 应用于原始频谱图 $X(t, k)$ ，即可得到谐波和冲击源分量的频谱图：

$$X_h(t, k) = X(t, k) \cdot M_h(t, k) \quad (4-8)$$

$$X_p(t, k) = X(t, k) \cdot M_p(t, k) \quad (4-9)$$

最后，通过应用逆短时傅里叶变换将这些谱图转换到时域，即可计算出所需信号 x_h 与 x_p 。

本章中分离过程借助 Librosa 库中的 `decompose.hpss` 方法，用于实验的分离因子为 1.05。在分离之前，将立体声音频转换为单声道。如图 4-3 所示，将 HPSS 算法应用在输入信号上时，在 Mel 频谱图上谐波倾向于形成水平结构（在时间方向上），而冲击源倾向于形成垂直结构（在频率方向上）。

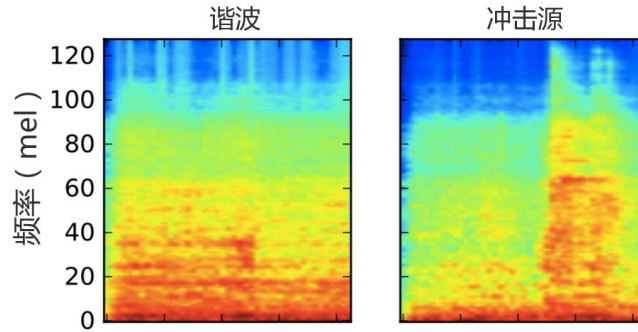


图 4-3 谐波冲击源分离法的 MFCC 频谱图

4.3 改进系统的结构设计

在上一章中的卷积神经网络包含了 2 个卷积模块，且使用单一的 Mel 频谱图作为输入。由于网络结构较简单，系统性能只能片面的通过调整参数来提升。然而过多的参数会使模型对数据产生依赖性，导致泛化性不强。因此通过改变网络结构，如增加深度等的办法来增强系统对不同数据集的分类能力，为增强系统性能的一个重要手段。

近年来，在计算机视觉领域中使用深度卷积神经网络已经很普遍。网络深度

的增加带来的一大好处就是系统的灵活性大大增强。在目前广泛应用的框架中，VGGNet（Simonyan K et al., 2014）得益于其简洁的架构和很强的拓展性，受到了广泛的应用。VGGNet 由牛津大学计算视觉组与谷歌 DeepMind 共同研发。VGGNet 探究了卷积神经网络深度与其性能间的关系，证明了网络性能可以通过增加深度来增强。其一大特点在于整个神经网络都使用了 3×3 的卷积核尺寸与 2×2 的最大池化尺寸。此外，尽管 VGGNet 的层次比常规神经网络更深，参数更多，但 VGGNet 只需很少的迭代次数就可收敛，原因在于网络的深度和小尺寸的滤波器起到了隐式的规范化作用；另一方面，其在特定的层使用预训练得到的数据进行参数的初始化。

典型的 VGGNet 结构如图 4-4 所示。

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

图 4-4 典型的 VGGNet 结构

本章的卷积神经网络受 VGGNet 的启发，决定也使用 3×3 的卷积核尺寸，并尝试通过增加卷积层数来使系统分类性能得到提升。所设计的系统的整体架构如图 4-5 所示。

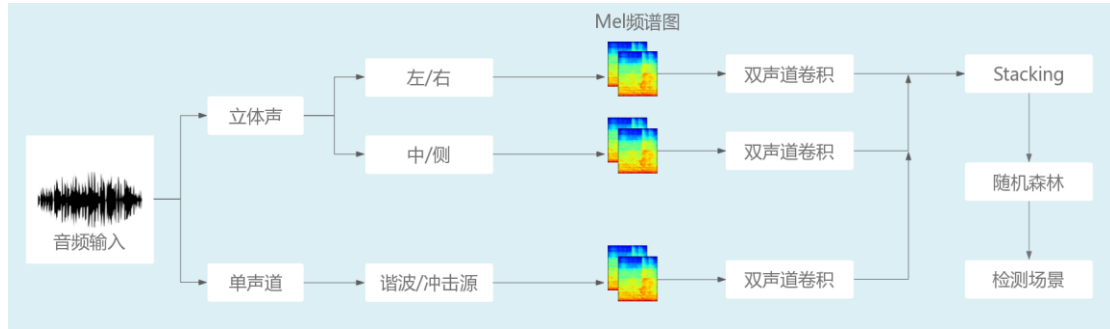


图 4-5 改进系统的结构设计

音频输入后，系统首先将音频分为两部分，第一部分做双声道分离，分离出左、右两声道，并分别求和、求差得到 Mid、Side 声道；第二部分压缩成单声道，然后经 HPSS 算法处理得到谐波信号和冲击源信号。之后，再对上一步分离出的三对音频信号进行 MFCC 频谱图提取，作为特征输入到双声道卷积模型中。接着，对三组双声道模型进行集成，集成时采用 4.4.2 节中的 Stacking 方法，选用的元学习算法为 4.4.4 节中的随机森林算法。经过对三组模型的集成学习，最终输出检测场景。其中双声道卷积模型的细节如图 4-6 所示。

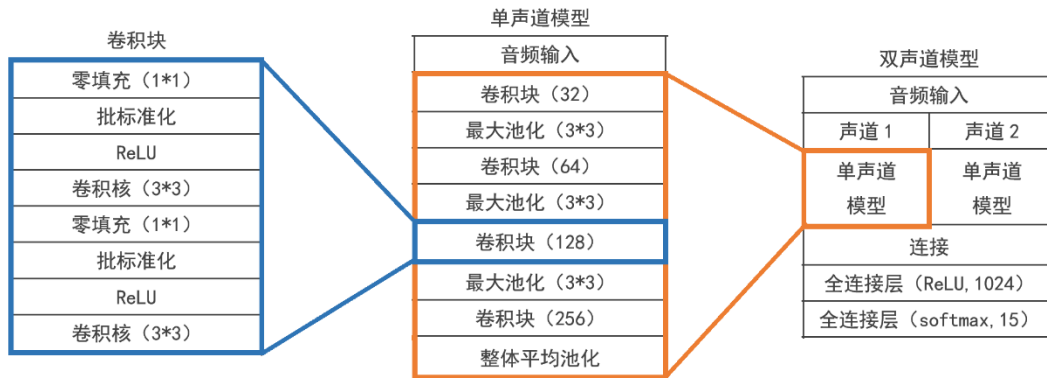


图 4-6 双声道卷积模型实现细节

整个网络组织包括三层：卷积块、单声道模型和双声道模型。其中，卷积块负责系统核心的卷积操作，且包含了零填充、批标准化、激活函数等步骤。卷积块中的每一个卷积核的尺寸都为 3×3 ，零填充尺寸为 1×1 。单声道模型由卷积块与最大池化、整体平均池化步骤组成，负责处理输入的其中一个声道，每一个声道上设有 4 层卷积块。类似于 VGGNet 的设计，每一个卷积块的滤波器数量依次翻倍，为 32、64、128、256，旨在提取不同尺度上的特征。最后，双通道模型又由两个单声道模型连接而成，连接后通过两层全连接层并输出结果。整个双声道卷积模型中一共有 8 个卷积块，且依然保留了整流线性单元 (ReLU) 和批标准化步骤，借此提高系统分类的准确性。相比于 VGGNet 的改进主要来自于在卷积

过程之前与应用于输入数据的批标准化。

4.4 集成学习

本章经过音频处理后一共得到了三组特征，且每一组特征图后都有对应的双声道卷积神经网络模型进行分类操作，故在集成过程前一共有三组独立模型。那么是否存在一种方法能结合三组模型的优点，使分类性能、泛化性能更为强大，是本节我们要讨论的问题。本节将从集成学习的概念开始，阐述本章使用的 Stacking 方法，并给出该方法在改进系统中的具体应用。

4.4.1 集成学习的概念

由于本章系统使用了不同的特征训练卷积神经网络，各网络结构所生成的结果存在差异。为了尽可能多的提升系统性能，可以引入集成学习（ensemble learning）方法。集成学习（周志华，2016）的基本思想为通过几个弱分类器的组合形成一个强分类器，即便某些弱分类器进行了错误的预测，也可以借助其他预测正确的弱分类器纠正回来，进而达到提升系统性能的效果。

假设 x 为一个输入， $m_i, i = 1 \dots k$ 为一组分类器，分类器的输出为每个类 $c_j, j = 1 \dots k$ 的概率分布 $m_i(x, c_j)$ ，则集成分类器的最终输出 $y(x)$ 可以表示为：

$$y(x) = \arg \max_{c_j} \sum_{i=1}^k w_i m_i(x, c_j) \quad (4-10)$$

其中， w_i 为分类器 m_i 的权重。集成学习就是根据分类目标计算每个分类器的最佳权重的方法。

相比于仅从许多普通模型中选择一个性能最好的模型，集成学习的优势体现在以下两个方面：首先是从统计的角度来看，加入可供学习的数据集不足，那么训练出的模型性能参差不齐，每个都会存在预测错误的风险。此时如果只使用一个分类器会导致泛化性能一般，正如本文第3章的系统，而结合多个分类器的结果可以减小预测错误的风险。其次是从运算的方面看，通常使用的网络优化方法可能会使系统陷入局部最优的情况，而将多个分类器结合后，由于经过了多次运算，可以减少系统陷入局部最优的情况，从而逼近整体最优解。最后是从表征的角度来看，在许多学习任务中，真实的未知假设不能被假设空间中的任何假设所表示。通过组合现有的假设空间，可以形成对真实未知假设的更准确的近似。

4.4.2 Stacking 方法

目前流行的集成学习算法包括 Stacking、Bagging、Boosting（李航，2012）、集成选择等。本文选择的集成学习算法为 Stacking 方法。Stacking 中文译为堆叠法，也称为超级学习或堆叠回归法，是一种高阶的集成学习算法。Stacking 是以一阶学习过程的输出作为输入开展二阶学习的过程，也称为“元学习”。尽管 Stacking 的概念（Wolpert，1992）早在 1992 年就被提出，但是直到 2007 年发表的一篇论文（Van et al.，2007）才提供了 Stacking 的理论保证。Stacking 方法之所以成为一个流行的集成学习方法，不仅因为它的实现相当简单，而且因为它可以显著的提升系统的泛化能力，这与本章需要的改进的方面相契合。图 4-7 直观的展示了 Stacking 方法的基本原理。

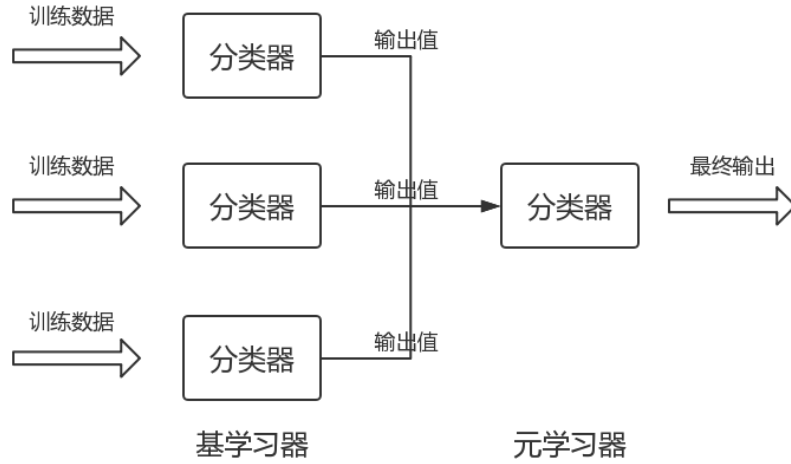


图 4-7 Stacking 方法的基本原理

学习器分为基学习器和元学习器两层，把基学习器的输出作为元学习器的输入。通过训练元模型来组合已经训练好的多个基模型的预测。且在 Stacking 方法中要求基模型产生不同的预测，即不相关预测。下面将先阐述 Stacking 方法的一般实现过程，下一小节中再讨论该方法在本章模型中的应用。

假设训练集为 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ ， N 为训练集 T 中样本的个数。基学习算法组为： $\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_M$ ， M 为基学习算法的个数。元学习算法为 \mathcal{L} 。首先，对于每一个基学习算法，使用训练集 T 生成基学习器 h_m ：

$$h_m = \mathcal{L}_m(T), \quad m = 1, \dots, M \quad (4-11)$$

然后，开辟一个新训练集 T' ，使 $T' = \emptyset$ 。接着，对于训练集 T 中的每一个样本，用每一个基学习器生成元训练集 T' ：

$$z_{nm} = h_m(x_n), \quad m = 1, \dots, M, n = 1, \dots, N \quad (4-12)$$

$$T' = T' \cup ((z_{n1}, z_{n2}, \dots, z_{nM}), y_n) \quad (4-13)$$

接着，在训练集 T' 上用元学习算法 \mathcal{L} 生成元学习器 h' ：

$$h' = \mathcal{L}(T') \quad (4-14)$$

最终的输出为：

$$H(x) = h'(h_1(x), h_2(x), \dots, h_M(x)) \quad (4-15)$$

训练集 T' 是基学习器在训练过程中产生的。如果直接用基学习器的训练集 T 来生成训练集 T' ，则很容易造成严重的过拟合。所以通常的解决办法是引入前两章中用于分割开发集的K折交叉验证法。通过将训练集 T 随机划分为k个大小相同的子集 T_1, T_2, \dots, T_k 。 T_i 表示第i折的训练集， $\bar{T}_i = T \setminus T_i$ 表示第i折的测试集。给定M个基学习算法 $\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_M$ ，通过在 T_i 上使用第m个基学习算法得到基学习器 $h_m^{(i)}$ 。对于 \bar{T}_i 中的每一个样本 x_n ，使 $z_{nm} = h_m^{(i)}(x_n)$ ，那么由样本 x_n 所生成的元训练样例的标记部分为 y_n ，示例部分为 $z_n = (z_{n1}, z_{n2}, \dots, z_{nM})$ 。故在K折交叉验证完成之后，将生成的元训练集 $T' = \{(z_n, y_n)\}_{n=1}^N$ 用于训练元学习器。

在分析完Stacking原理后我们可以总结出其特点，即在集成过程中充分利用不同算法从不同的数据空间角度和数据结构角度对数据的不同观测，来取长补短优化结果。

4.4.3 Stacking 在改进系统中的应用

参考上一小节的方法，由于在音频处理过程之后产生了3组特征，且每组特征都由双声道神经网络独立训练，因此为3个独立的模型，满足Stacking方法对模型预测不相关性的需要。

首先将训练集中4680个文件拆分为训练数据与测试数据，其中训练数据共3825个，每类255个；测试数据共855个，每类57个。其中测试数据占训练集数据总数约22.35%。然后，在训练数据上再将数据等分为3折，每1折的音频数据为1275个。注意，上述数据分离时均保证每一场景的音频文件数量一致。

先使用本章设计的卷积神经网络模型训练基于LR特征的基学习器。将3825个训练数据用3折交叉验证法分为三组数据（即3个fold），其中每一组数据包含2折用于学习，学习完成后用剩下的1折进行预测，预测的结果将作为元训练集的训练数据，此过程每组数据进行一次，共3次。至此，元训练集的特征数量，即预测结果与初始的训练数据数量相同，为3825个，且都由不同组的不同预测数据进行预测得出。元训练集的生成过程如图4-8所示。

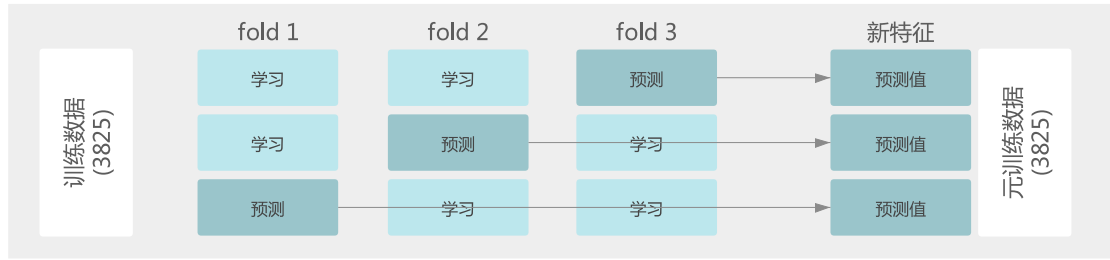


图 4-8 元训练集的生成示例

与此同时，每一组数据在训练完成后，其训练好的模型还要在 855 个测试数据上进行测试，以生成 3 组预测结果。然后，再将 3 组预测结果用多数优胜法进行投票，得到一个含 855 个数据的预测值，作为元测试集的测试数据。由于测试过程中没有使用训练集的数据，因此没有数据泄漏，进而避免了过拟合的发生。元测试集的生成过程如图 4-9 所示。

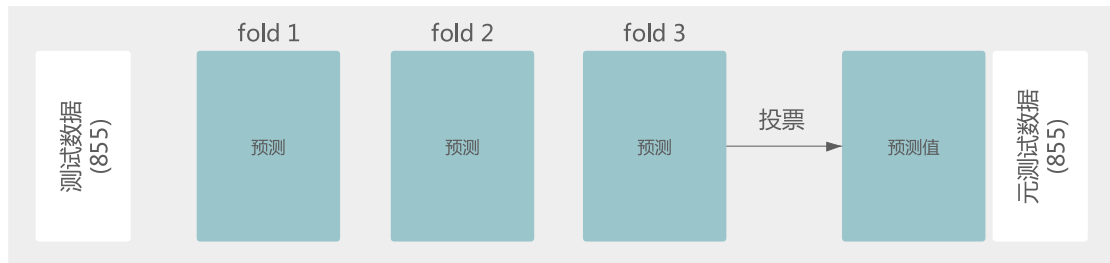


图 4-9 元测试集的生成示例

至此，LR 模型的训练完毕，再以相同的方法训练 MS 和 HPSS 模型，得到另外两组元训练集与测试集。这样，就可以得到一个 3×3825 个数据的元训练集与一个 3×855 的元测试集。

4.4.4 随机森林算法

在生成了元训练集与测试集后，需要借助元学习算法对数据进行元学习以产生最终的输出。目前得以广泛使用的元学习算法有基于统计方法的投票法和平均法；基于经典机器学习算法的逻辑回归和决策树；基于非线性机器学习算法的 KNN (k-Nearest Neighbors)、随机森林 (Random Forest) 以及基于多响应回归的 MLR (Multi-Response Linear Regression) (徐慧丽, 2018)。

结合元数据集的实际情况，本章决定使用随机森林 (Breiman, 2001) 作为元学习器算法。原因在于元训练与测试数据集较大，且输入特征维数较多，使用随机森林进行分类训练速度快且对数据的适应能力强，这就意味着只需调整很少的参数就能使模型达到不错的分类效果。此外，由于引入了随机性，使得模型不易过拟合。随机森林由决策树算法改进而来，该算法将多颗具有相同分布的决策

树合并，其中每棵树在建立时依赖于一个独立抽取的样本。尽管单颗树可能分类性能一般，但经过大量随机产生的决策树后，其分类性能会大大增强。下面简述使用随机森林进行分类的实现过程：

- (1) 假设原训练集为 N ，用自助法 (bootstrap) (Efron and Tibshirani, 1994) 从中有放回的随机抽取 k 个样本，共抽取 n_{tree} 次以得到 n_{tree} 个新训练集。
 - (2) 用 n_{tree} 个新训练集构建 n_{tree} 个分类树，且对于每个分类树，根据节点纯度最小原则进行自顶向下的分裂。
 - (3) 对分类树一直进行分裂且不剪枝，直到树中每一节点都归于对应的类。
 - (4) 将生成的 n_{tree} 颗分类树组成随机森林，用该随机森林对新数据进行分类，分类结果取决于分类树的投票数。
- 其中，衡量纯度的方法为 Gini 系数：

$$Gini = 1 - \sum_{m=1}^M p_m(1 - p_m) = 1 - \sum_{m=1}^M p_m \quad (4-16)$$

其中， M 为类别总数， p_m 为样本属于类别 m 的概率。Gini 系数越大，说明纯度越低，反之说明纯度越高。

在本文的元学习器训练与测试中，使用了 sklearn 库的 RandomForestClassifier 内置模型。元学习器的输入为上一小节介绍的元训练集与测试集，输出为分类后的音频场景序列。除分类树个数外均使用默认参数，其中不对树的最大深度做限制，分裂标准使用 Gini 系数。实验过程中将设置几组不同的分类树个数，以判定元学习器的相对最佳学习参数。

4.5 实验结果与分析

4.5.1 实验环境与数据集处理

为了与之前的系统形成对照，本章实验依然使用TUT Acoustic Scenes 2017的两组数据作为数据集。网络模型优化算法依然取Adam Optimizer，且参数与上一章保持一致，完整训练周期为100个迭代周期，批大小为16。

在进行训练时，先用本章设计的卷积神经网络模型独立完整训练由LR、MS、HPSS方法进行音频处理的个体学习器，并记录其分类准确率。与之对照的，另建一个输入特征为单通道的只提取Mel频谱图的学习器，网络结构与本章所设计的结构相同。该对照模型不进行集成，只用于比较本章研究研究ichu1tic的音频处理方法是否对分类准确率有改进。之后，再根据Stacking方法对各模型进行集成，

其中元学习器的分类树个数分别设置为1000、2000、3000，且对于每一不同的元学习器参数都进行完整训练，找到最佳分类树个数作为最终系统的参数。最后，将调整好的改进系统在开发集上训练，并在验证集上测试。

4.5.2 结果分析

在完整训练结束之后，各种方法在开发集上的平均分类准确率如表 4-1 所示。

表 4-1 各种方法在开发集上的平均准确率（单位：%）

算法	平均准确率
第 3 章	93.4
单通道	84.5
LR	87.3
MS	87.8
HPSS	86.7
RF(1000)	91.8
RF(2000)	92.1
RF(3000)	92.0

从表 4-1 中可以看出，在引入了 LR、MS、HPSS 音频处理方法后，系统的分类准确率相对于单通道系统更高，这印证了对音频处理的合理性。在集成了各种模型以后，模型的分类准确率又有了不小的提升。此外，将随机森林的分类树个数设置为 2000 能使系统发挥最佳性能。因此，采用该参数作为本章分类系统最终参数，并在验证集上使用该参数进行测试。

最终的改进系统模型在开发集与验证集上的各场景分类准确率如表 4-2，观察表 4-2 我们可以发现，尽管验证集上的平均分类准确率相比开发集还是有较为明显的下降，但是已经远好于第 3 章系统 59.7%的结果。由此可见改进后的系统泛化能力得到了明显的提升，尤其在地铁站、森林小径、汽车、家的场景下有超过 90%的分类准确率。但公园与图书馆场景下的分类准确率仍然不理想，还有一定的改进空间。

表 4-2 改进系统的各场景分类结果（单位：%）

音频场景	开发集	验证集
沙滩	89.6	76.3
公交	98.2	71.9
咖啡/餐馆	88.3	81.2
汽车	99.0	92.4
市中心	89.7	88.7
森林小径	99.8	95.5
杂货铺	93.6	75.8
家	84.1	93.6
图书馆	88.5	52.1
地铁站	100.0	100.0
办公室	98.9	83.4
公园	80.3	46.8
小区	86.8	74.3
火车	91.4	90.6
电车	92.7	61.0
平均	92.1	78.9

4.5.3 与其他系统的对比

本小节分别将本章设计的改进系统与第 2 章和第 3 章的系统在开发和验证两个数据集上进行对比，以便分析出改进系统的具体性能。从表 4-3 中我们可以看出，无论是在开发集还是在验证集上，改进系统的分类准确率在绝大多数场景上均大幅领先第二张的 GMM 系统。尤其在沙滩、公交、图书馆、公园等场景的验证集上都有很大的优势。分析其可能的原因，一方面在于卷积神经网络本身对数据的学习能力更强，另一方面在于以上场景的声场空间较大且有固定的底噪，刚好契合了本章音频处理时对固定场景音频环境的分析。

分析表 4-4 中我们可以看出，尽管在开发集中，改进系统并没有取得整体上的优势，但在验证集上改进系统的总体分类准确率相对第 3 章的系统有了约 19% 的提升。这说明经过改进后系统的泛化性能有了明显的增强，也说明通过对网络深度的增加以及网络参数的简化，系统的灵活性能有了很大的提升。且同上一章中容易混淆的几个场景对比，公交-电车、公园-小区等易混淆的场景对在验证集上的准确率有了明显的提高，这在相当程度上印证了音频处理的合理性。但是从

具体类别来看，公园、图书馆场景的准确率还有很大的提升空间。

表 4-3 GMM 系统与改进系统的比较

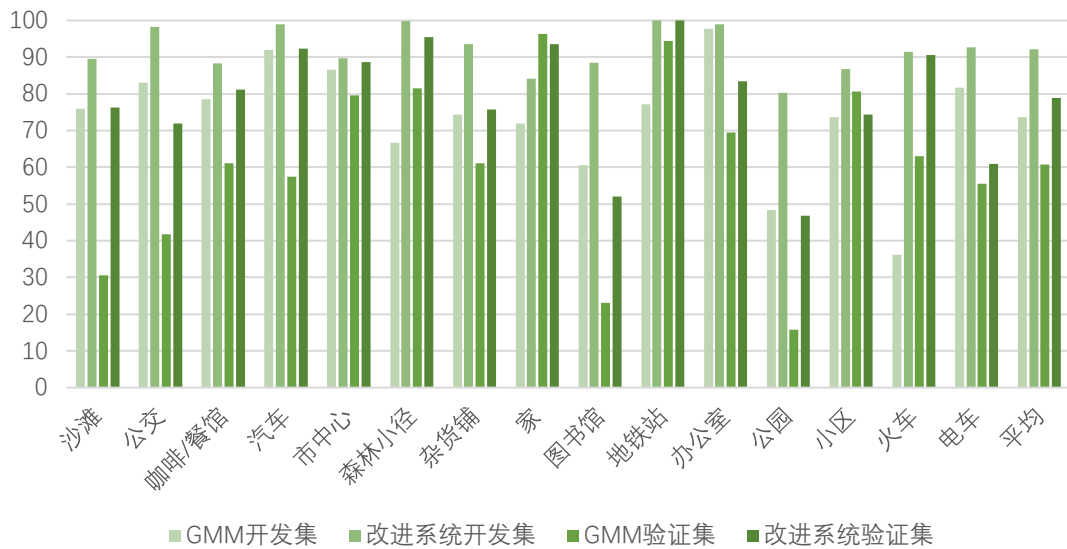
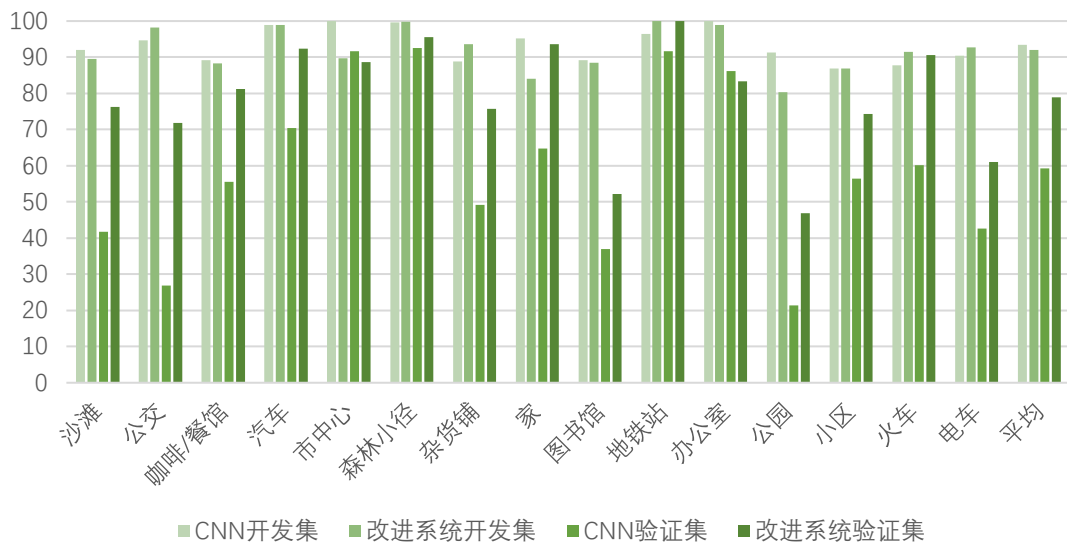


表 4-4 CNN 系统与改进系统的比较



4.6 本章小结

本章首先从第三章设计系统的结果出发,分析了之前系统的不足之处。接着,给出了两个方面的研究思路,并据此设计了改进系统。在音频处理方面,采用了双耳表示法与谐波冲击源分离法。在网络结构方面,通过引入类 VGGNet 结构以提高网络结构的灵活性。接下来,又详细阐述了将多个模型集成的集成学习方法。最后,进行了实验验证,一是表明音频处理可以使系统对空间信息进行利用,从而进一步提高分类准确率;二是通过集成学习得到最终改进系统比第三章系统的泛化性能更好。其中在测试集上的分类准确率为 92.1%,验证集上的分类准确率为 78.9%。

结 论

音频场景分类作为音频处理任务的一个重要的分支,近年来已经得到越来越多专家学者的研究,并具有广阔的应用空间。尽管目前已经有将卷积神经网络使用到音频场景分类中的例子,但之前不少研究中的特征提取方法单一,且网络结构的设计推广性不佳。本文尝试对以上这些问题做出改进,建立了几组分类模型,并逐步提高了场景分类准确率。本文所进行的研究工作及结论如下:

设计了一套基于 MFCC 和 GMM 的音频场景分类基线系统,并在开发与验证两个数据集上得出了分类结果,将此结果用于之后系统的对照。

设计并优化了一套基于卷积神经网络的音频场景分类系统。通过在预备实验中调整滤波器尺寸与数量,得到了相对最佳的系统参数,并应用于分类系统之中。在这个系统中,开发集上的分类结果大幅优于基线系统,此结果表明卷积神经网络在小数据集上的学习能力相比典型的传统机器学习方法更强。此外,由于在验证集上的分类情况不慎理想,又根据验证集的分类混淆矩阵分析了系统泛化性能不佳的原因:一是将双通道音频压缩成单通道导致无法捕捉空间信息;二是系统结构简单且参数过多,过分依赖于训练数据,导致推广性不佳。

根据之前基于卷积神经网络的音频场景分类系统中的问题,设计了一个改进的分类系统。在音频处理方面,改进系统中使用了双耳表示法与谐波冲击源分离法,将音频输入处理为 3 组:左右声道、中侧声道与谐波冲击源,并逐组提取其梅尔频谱图。音频处理使分类系统对空间的感知能力得到了提升,进而提升了分类准确率。在网络结构方面,本文在流行的 VGGNet 结构上加以改进,对网络结构的完善使得网络的泛化性能也由于之前的系统。此外改进系统还使用了 Stacking 集成学习方法,将 3 组网络进行集成,集成后系统分类能力得到进一步的提升。最终,系统在开发集与验证集中皆达到了大幅优于基线系统的结果。

尽管本文完整实现了基于卷积神经网络的音频场景分类模型,但受限于时间与精力,系统中还存在着一些改进的空间。具体可以总结为以下两点:

(1) 为了使文中设计的三组系统形成对比,仅使用了 TUT Acoustic Scenes 2017 这一个数据集。系统在别的数据集,尤其是由不同设备录制的音频场景文件的分类性能还有待验证。

(2) 由于卷积神经网络模型较为复杂,本文仅从滤波器尺寸数量、与网络结构方面进行了研究,其他诸如学习速率、网络模型优化算法等方面依然存在着广泛的讨论空间。

致 谢

参考文献

- 李航. 2012. 统计学习方法[J]. 清华大学出版社, 北京.
- 金海. 2016. 基于深度神经网络的音频事件检测[D]. 华南理工大学.
- 徐慧丽. 2018. Stacking 算法的研究及改进[D]. 华南理工大学.
- 郑继明, 魏国华, 吴渝. 2009. 有效的基于内容的音频特征提取方法[D].
- 周志华. 2016. 机器学习[M]. 清华大学出版社.
- Aytar Y, Vondrick C, Torralba A. 2016. Soundnet: Learning sound representations from unlabeled video[C]//Advances in Neural Information Processing Systems. 892-900.
- Ballas J A. 1993. Common factors in the identification of an assortment of brief everyday sounds[J]. Journal of experimental psychology: human perception and performance, 19(2): 250.
- Barchiesi D, Giannoulis D, Stowell D, et al. 2015. Acoustic scene classification: Classifying environments from the sounds they produce[J]. IEEE Signal Processing Magazine, 32(3): 16-34.
- Breiman L. 2001. Random forests[J]. Machine learning, 45(1): 5-32.
- Clarkson B, Sawhney N, Pentland A. 1998. Auditory context awareness via wearable computing[J]. Energy, 400(600): 20.
- Davis S, Mermelstein P. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences [J]. IEEE transactions on acoustics, speech, and signal processing, 28 (4): 357-366.
- Defréville B, Pachet F, Rosin C, et al. 2006. Automatic recognition of urban sound sources[C]//Audio Engineering Society Convention 120. Audio Engineering Society.
- Dempster A . 1977. Maximum likelihood from incomplete data via the EM algorithm[J]. Journal of the Royal Statistical Society, Series B, 39.
- Dubois D, Guastavino C, Raimbault M. 2006. A cognitive approach to urban soundscapes: Using verbal data to access everyday life auditory categories[J]. Acta acustica united with acustica, 92(6): 865-874.
- Efron B, Tibshirani R J. 1994. An introduction to the bootstrap[M]. CRC press.
- Eghbal-Zadeh H, Lehner B, Dorfer M, et al. 2016. CP-JKU submissions for DCASE-2016: a hybrid approach using binaural i-vectors and deep convolutional neural networks[J]. IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE).
- Eronen A J, Peltonen V T, Tuomi J T, et al. 2006. Audio-based context recognition[J]. IEEE Transactions on Audio Speech & Language Processing, 14(1):321-329.

- Eronen A, Tuomi J, Klapuri A, et al. 2003. Audio-based context awareness-acoustic modeling and perceptual evaluation[C]//Acoustics, Speech, and Signal Processing, 2003.
- Fukushima K. 1980. Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position[J]. Biological Cybernetics, 36(4): 193-202.
- Guyot P, Pinquier J, André-Obrecht R. 2013. Water sound recognition based on physical models[C]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE: 793-797.
- Heittola T, Mesaros A, Eronen A, et al. 2013. Context-dependent sound event detection[J]. EURASIP Journal on Audio, Speech, and Music Processing, 2013(1): 1.
- Hinton G E, Osindero S, Teh Y W, et al. 2006. A fast learning algorithm for deep belief nets[J]. Neural Computation, 18(7): 1527-1554.
- Hinton G E, Srivastava N, Krizhevsky A, et al. 2012. Improving neural networks by preventing co-adaptation of feature detectors[J]. arXiv preprint arXiv:1207.0580.
- Hu G, Wang D L. 2007. Auditory segmentation based on onset and offset analysis[J]. IEEE Transactions on Audio, Speech, and Language Processing, 15(2): 396-405.
- Jung J W, Heo H S, Yang I L H, et al. 2017. DNN-based audio scene classification for DCASE 2017: dual input features, balancing cost, and stochastic data duplication[J]. System, 4(5).
- Kingma D P, Ba J. Adam. 2014. A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980.
- Krijnders J, t Holt G. 2013. Tone-fit and MFCC scene classification compared to human recognition[J]. Energy [dB], 400(450): 500.
- Krizhevsky A, Sutskever I, Hinton G E. 2012. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 1097-1105.
- LeCun Y, Bottou L, Bengio Y, et al. 1998. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 86(11): 2278-2324.
- Lidy T, Schindler A. 2016. CQT-based convolutional neural networks for audio scene classification[C]//Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016). DCASE2016 Challenge, 90: 1032-1048.
- Lin M, Chen Q, Yan S. 2013. Network in network[J]. arXiv preprint arXiv:1312.4400.
- Malkin R G, Waibel A. 2005. Classifying user environment for mobile applications using linear autoencoding of ambient audio[C]// ICASSP '05. IEEE International Conference on Acoustics, Speech, and Signal Processing.

- Maxime J, Alameda-Pineda X, Girin L, et al. 2014. Sound representation and classification benchmark for domestic robots[C]//2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE: 6285-6292.
- Mesaros A, Heittola T, Virtanen T. 2018. Acoustic scene classification: an overview of dcase 2017 challenge entries[C]//2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC). IEEE, 411-415.
- Moore A W, Jorgenson J W. 1993. Median filtering for removal of low-frequency background drift[J]. Analytical chemistry, 65(2): 188-191.
- Nuttall A H . 1972. Some Integrals Involving the Q-Function[J]. IEEE Transactions on Information Theory, 21(1):95-96.
- Ono N, Miyamoto K, Le Roux J, et al. 2008. Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram[C]//2008 16th European Signal Processing Conference. IEEE, 1-4.
- Park J, Shin J, Lee K. 2017. Exploiting continuity/discontinuity of basis vectors in spectrogram decomposition for harmonic-percussive sound separation[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(5): 1061-1074.
- Patil K, Elhilali M. 2002. Multiresolution auditory representations for scene classification[J]. cortex, 87(1): 516-527.
- Peltonen V T K, Eronen A J, Parviainen M P, et al. 2001. Recognition of everyday auditory scenes: potentials, latencies and cues[J]. PREPRINTS-AUDIO ENGINEERING SOCIETY.
- Piczak K J. 2015. Environmental sound classification with convolutional neural networks[C]//2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP). IEEE, 1-6.
- Radhakrishnan R, Divakaran A, Smaragdis A. 2005. Audio analysis for surveillance applications[C]//IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. IEEE, 2005: 158-161.
- Reynolds D A, Rose R C. 1995. Robust text-independent speaker identification using Gaussian mixture speaker models[J]. IEEE transactions on speech and audio processing, 3(1): 72-83.
- Rijsbergen C J V. 1979. Information Retrieval[M].
- Salakhutdinov R, Mnih A, Hinton G E, et al. 2007. Restricted Boltzmann machines for collaborative filtering[C]. international conference on machine learning, 791-798.
- Santoso A, Wang C Y, Wang J C. 2016. Acoustic scene classification using network-in-network based convolutional neural network[R]. DCASE2016 Challenge, Tech. Rep.

- Sawhney N, Maes P. 1997. Situational awareness from environmental sounds[J]. Technical Report, Massachusetts Institute of Technology.
- Schilit B N, Adams N, Want R. 1994. Context-aware computing applications[M]. Xerox Corporation, Palo Alto Research Center.
- Shimodaira H. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function[J]. Journal of statistical planning and inference, 90(2): 227-244.
- Simonyan K, Zisserman A. 2014. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556.
- Tardieu J, Susini P, Poisson F, et al. 2008. Perceptual study of soundscapes in train stations[J]. Applied Acoustics, 69(12): 1224-1239.
- Van der Laan M J, Polley E C, Hubbard A E. 2007. Super learner[J]. Statistical applications in genetics and molecular biology, 6(1).
- Wiesel T N, Hubel D H. 1965. EXTENT OF RECOVERY FROM THE EFFECTS OF VISUAL DEPRIVATION IN KITTENS[J]. Journal of Neurophysiology, 28(6):1060-1072.
- Wolpert D H. 1992. Stacked generalization[J]. Neural networks, 5(2): 241-259.
- Xu Y, Li W J, Lee K K. 2008. Intelligent wearable interfaces[M]. John Wiley & Sons.
- Zeinali H, Burget L, Cernocky J. 2018. Convolutional neural networks and x-vector embedding for dcase2018 acoustic scene classification challenge[J]. arXiv preprint arXiv:1810.04273.

攻读学位期间取得学术成果

已发表论文:

第一作者.2019.数字通信世界.普刊.