学校代码：10284
分 类 号：TP302.7
密　　级：公开
U D C：004.93
学　　号：MG20370040

南京大学

硕 士 学 位 论 文

论 文 题 目　基于编解码理论的信号表示研究

作 者 姓 名　向浩然
专 业 名 称　计算机科学与技术
研 究 方 向　信号处理
导 师 姓 名　申富饶

2023 年 05 月 23 日

答辩委员会主席 戴新宇 教授

评　　阅　　人 戴新宇 教授

徐明华 教授

论文答辩日期　2023　年 05 月 22 日

研究生签名：向浩然

导师签名：

Research of Signal Representation Based on Encoding and Decoding Theory

by

XIANG Hao-ran

Supervised by

Professor SHEN Fu-Rao

A dissertation submitted to
the graduate school of Nanjing University
in partial fulfilment of the requirements for the degree of
MASTER
in
Computer Science and Technology

School of Artificial Intelligence
Nanjing University

May 23, 2023

南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目：基于编解码理论的信号表示研究

计算机科学与技术 专业 2020 级硕士生姓名：向浩然

指导教师(姓名、职称)：申富饶 教授

摘　要

随着科学技术的迅猛发展，人类进入了信息时代。如何更好地理解信号、表示信号是信号处理领域重要课题。信号表示算法在深度学习时代蓬勃发展，能够应用到音乐信号推荐，雷达信号分选以及语音信号增强等众多领域。然而，现有的算法分散在各个不同的研究领域，没有形成系统的框架。一方面，在各个相对独立的领域中，基于深度学习编解码理论的信号表示算法存在缺陷，比如：信号具有多义性，而现有的算法不能对信号中丰富的信息进行解耦和有效表示；对信号的表示并不仅仅局限于信号内容，也会利用到其它与之相关的先验知识或特征，而目前的研究对这些信息的利用是不充分的。另一方面，目前的信号表示的研究在实际应用中存在困难，存在模型参数量太大、速度较慢、延迟过高等问题，这限制了基于深度学习的信号表示算法在实际中的应用。为了解决以上信号表示中缺乏系统理论和其它存在的问题，并将形式化的编解码信号表示理论与实际应用相结合，本文聚焦于信号表示的研究和应用，主要内容如下：

1. 形式化并提出了一种基于编解码理论的信号表示框架，并基于该理论框架讨论信号表示算法中的重要组成部分。在该框架下，我们将信号表示问题分为了信号纯表示、脉冲信号解码表示和连续信号解码表示三类，每一类算法都在不同的实际领域中发挥着重要作用。

2. 对信号纯表示问题，我们构建了一种信号纯表示框架。该框架融合了两种信号纯表示的常见范式。首先通过弱监督学习获取信号内容表示，再用解纠缠模块对信号内容表示的多义性进行分解，最后基于图神经网络算法融合信号内容表示和关系表示。我们还设计了多重关系损失训练信号纯表示相关的内容表示模型、解纠缠模型和关系表示模型，从而更好地度量信号空间，有效表示信号。

3. 对脉冲信号解码表示问题，我们提出了深度脉冲信号掩膜算法，该算法的核心是递归表示网络。递归表示网络基于空洞卷积与双路径注意力机制，递归地预测脉冲分量表示掩膜，从而得到脉冲分量表示。它能够解决因环境噪声和多种调制模式共存而导致的歧义性问题。进一步的，深度脉冲掩膜算法还提供了可选的预处理和后处理聚类方法，进一步利用带噪声的脉冲描述字信息，得到更鲁棒的脉冲表示和分选结果。

4. 对连续信号解码表示问题，我们设计了实时频域卷积时序网络。一方面，我们不仅设计了带缓存的因果卷积模块以和特殊的门控机制支持流式实时信号解码表示，还设计了合适的时频分量掩膜表示和损失函数面向具体任务解决问题。另一方面，我们对比和使用了几种模型轻量化算法，尽可能地压缩实时频域卷积时序网络的参数量，提高实时率，使得它能够在边缘设备上使用。

对于以上每种信号表示算法，我们均在大量数据和实际任务上进行了实验验证，其效果能达到或超越现有的信号表示算法水准，在主观感受、客观评价指标下都有良好表现，对信号中存在的噪声的鲁棒性好。在形式化理论框架和改进理论缺陷的研究基础上，本文基于编解码理论的信号表示框架实现了与实际应用的结合，并取得了显著成效。

关键词： 信号表示；信号处理；音乐推荐；雷达分选；语音增强

南京大学研究生毕业论文英文摘要首页用纸

THESIS: Research of Signal Representation Based on Encoding and Decoding Theory

SPECIALIZATION: Computer Science and Technology

POSTGRADUATE: XIANG Hao-ran

MENTOR: Professor SHEN Fu-Rao

ABSTRACT

With the rapid development of science and technology, mankind has entered the information age. How to better understand signals and represent signals is an important topic in the field of signal processing. Signal representation algorithms are booming in the era of deep learning and can be applied to many fields such as music recommendation, radar pulse sorting, and speech enhancement. However, existing algorithms are scattered in various research fields and do not form a systematic framework. On the one hand, in various relatively independent fields, signal representation algorithms based on deep learning encoding and decoding theory have shortcomings, such as: signals have ambiguity, and existing algorithms cannot decouple and effectively represent the rich information in signals; The representation of signals is not limited to the content of the signal, but also makes use of other prior knowledge or features related to it, and the current research does not make sufficient use of this information. On the other hand, the current research on signal representation has difficulties in practical applications, and there are problems such as too many model parameters, slow speed, and high delay, which limits the application of signal representation algorithms based on deep learning in practice. In order to solve the lack of systematic theory and other existing problems in the signal representation mentioned above, and to combine the formal theory of encoding and decoding signal representation with practical applications, this paper focuses on the research and application of signal representation. The main content is as follows:

1. A unified theoretical framework for signal representation is constructed, and the important components of signal representation algorithms are discussed based on this theoretical framework. Under this framework, we divide the signal representation problem into three categories: signal pure representation, pulse signal decoding representation and continuous signal decoding representation, and each type of them plays an important role in different practical fields.

2. For the signal pure representation problem, we construct a Signal Pure Representation Framework (SPRF). The framework integrates two common patterns of pure signal representation training. First, the signal content representation is obtained through weak supervised learning, then the ambiguity of the signal content representation is decomposed by the disentangling module, and finally the signal content representation and relationship representation are fused based on the graph neural network. We design the multi-relationship loss to train signal pure representation related content representation model, disentangle model and relationship representation model, so as to better measure the signal space and effectively represent the signals.

3. For the pulse signal representation and decoding, we propose a Deep Pulse Signal Mask (DPSM) algorithm. The core of this algorithm is the Recursive Representation Network (RRN). The RRN is based on the dilation convolution and dual-path attention mechanism, and recursively predicts the pulse component representation mask to obtain the pulse component representation. It can solve the ambiguity problem caused by the coexistence of environmental noise and multiple modulation modes. Further, the DPSM algorithm also provides optional pre-processing and post-processing clustering methods, and further uses the noisy pulse descriptor information to obtain more robust pulse representation and sorting results.

4. For the continuous signal representation and decoding problem, we design a Real-Time Frequency Convolution Recursive Network (RTFCRN). On the one hand, we design a causal convolution module with cache and a special gating mechanism to support streaming real-time signal decoding representation, and design appropriate time-frequency masks and loss functions to solve problems for specific tasks. On the other hand, we compare and use several model lightweight algorithms to compress the

number of parameters of the RTFCRN as much as possible to improve the real-time rate, so that it can be used on edge devices.

For each of the above signal representation algorithms, we have carried out experimental verification on a large number of data and practical tasks, and its effect can reach or exceed the level of existing signal representation algorithms. It has good performance under subjective feelings and objective evaluation metrics, and is robust to the noise in the signal. Based on the research on the formal theoretical framework and the improvement of theoretical deficiencies, this paper combines the signal representation framework based on encoding and decoding theory with practical applications and has achieved significant results.

KEYWORDS: Signal Representation, Signal Processing, Music Recommendation, Radar Signal Sorting, Speech Enhancement

目　　录

插图清单

附表清单

第一章　绪论

1.1　研究背景与意义

信号（Signal）是信息的表现形式或传送载体，从广义上看，文字、图像、声音，甚至是湖面的波纹，只要能够传递信息，都能够被称为信号。随着时间变化而变化的信号是传统意义上所研究的信号，其自变量为时间，可被分为模拟信号和数字信号。本文研究的是时序信号的表示，也包括信号分量的表示。信号最基本的表示形式是时域表示。信号表示（Signal Representation）是信号处理的基础，研究的是如何将时域信号变换为其它更易于处理的表示形式，所以信号表示也被称为信号变换。同样的时序信息能够被不同形式的信号所表示，而不同的表示形式会极大影响信号处理的难度。模拟信号是用连续变化的物理量表示的信息，其信号的幅度随时间作连续变化，其代表的信息能够呈现为任意数值。数字信号则是离散化的信号，通常用有限数字中的一个数字来表示，是在模拟信号的基础上经过采样、量化和编码而形成的。数字信号是计算机能够处理的信号，而数字信号处理（Digital Signal Processing）[1] 是研究数字信号表示、滤波、检测、调制解调等算法的传统信号处理方法，其长久以来的研究成果为信号提供了多样化的表示算法。但随着信号处理问题逐渐从单纯的处理信号转变为对信号的分解和理解，传统信号处理算法的能力不够用了。构造新的信号表示方法是信号处理中的关键问题之一，多样化和具有面向任务特质的信号表示通常能够让信号处理事半功倍。它对信号的传输、分解和使用均起着重要作用，也能够帮助我们更好地理解信号的内容，构建出信号与信号之间的关系。

随着互联网的普及，信号的表示和处理逐步成为了各种现实应用的基石，涉及到生活和生产的方方面面。信号的稀疏表示[2-3] 在数据压缩领域得到广泛应用。稀疏表示与压缩感知理论指出，稀疏信号可通过一组线性测量值重建得到原信号，在获得测量值时，采样率可低于奈奎斯特采样频率。研究发现，信号的稀疏表示能够自然地贴近信号本质特征，已经广泛应用于语音、视频和图像的编码领域，在谱估计、系统控制、盲源分离、生物医学成像等方面也有较好

的应用。近年来，随着人工智能（Artificial Intelligence，AI）和深度学习（Deep Learning）的兴起，计算机视觉[4-7]、自然语言处理[8-11]、强化学习[12-15]和搜索推荐[16-19]等领域都焕发了新的生机。深度学习是端到端的学习算法，它能够自动地学习高阶非线形数据的表示，不需要手工设计规则和提取特征，从而能更轻易地提取有效表示。深度学习也同样影响到了信号处理领域，信号表示[20]实际上也是随着神经网络而出现的概念，在这之前属于信号变换算法的范畴。在这之中，音乐信号，雷达、水声信号以及语音信号处理等领域的研究比较多。

近年来，自动化的音乐推荐和检索已成为一个热门的研究的问题，因为现在许多音乐都是以数字方式进行销售和消费的。音乐信号表示的研究涉及到乐谱跟踪、智能音乐浏览界面、自动音乐分类等多个方面[21-22]。将音乐作为产品，通过协同过滤或者热门推荐等方式推荐给用户，这在网易云、抖音、腾讯音乐等音乐软件中都得到了广泛的使用。然而，这种方法存在冷启动和长尾问题，因此它对推荐新的和不受欢迎的歌曲无效。提取的音乐信号表示特征信息的不足，导致了推荐效果的不理想。因此，如何通过深度学习算法提取音乐的内容和关系特征，对新的音乐信号进行处理，从而得到更有意义的表示就成为了关键的问题。

电子战已成为现代战争的重要组成部分，电子支援措施（Electronic Support Measures，ESM）在电子战背景中发挥着重要作用[23]。ESM 系统的功能是拦截和分析电磁信号，构造信号表示，并快速识别威胁信号源[24]。雷达信号的表示和分选是 ESM 系统处理的关键技术。ESM 系统接收的脉冲信号由脉冲描述字（Pulse Description Word，PDW）描述。PDW 通常包括五个基本参数：脉冲宽度（Pulse Width，PW）、射频（Radio Frequency，RF）、脉冲幅度（Pulse Amplitude，PA）、到达方向（Direction of Arrival，DoA）和到达时间（Time of Arrival，ToA）[25]。其中，到达时间序列的一阶差称为脉冲重复间隔（Pulse Repetition Interval，PRI），这是脉冲序列的固有属性[26]。在雷达信号特征参数中，PRI 是工作方式最多样、参数范围最大、参数变化最快的参数之一。因为雷达工作模式的高度复杂性、电磁信号空间的高密度性、信号参数的复杂性、信号截获的低概率性，雷达脉冲信号的表示难度很大[27]。表示和分选雷达信号是后续执行不同的子任务的基石，例如信号源分类、模式识别和信号跟踪[28]。

语音信号是迄今为止最常用的交流方式。语音通信的应用越来越广泛，大

多通过电视电话会议、微信语音聊天等方式实现。随着这些应用的流行，人们不仅对语音质量的需求在逐渐变大，对语音质量的要求也迈上了新的台阶。传输用传统的采样量化得到的数字语音,会占用较多信道资源，也很容易受到环境噪声的影响。如何有效地进行语音信号编码和表示从而改善语音的通信质量，并降低传输语音的比特率，减少信道资源的占用和提高通信的实时率，是通信和语音信号处理中的重要问题[29]。所以，语音信号增强是语音信号表示中一个令人关注的研究方向[30]。由于噪声干扰会严重降低语音信号的感知质量和语音通信的可懂度，也会影响下游的语音识别等任务，因此语音增强具有坚实的实际应用价值。它在说话人识别、视频会议、通过通信信道的语音传输、基于语音的生物识别系统、移动电话、助听器、麦克风、语音等方面具有重要意义。模式挖掘方法在语音信号表示中发挥着关键作用。

在深度学习蓬勃发展的背景下，本文旨在研究基于编解码理论的信号表示算法，以实现对信号的准确表达和描述。本文将贴近实际应用来构建系统的信号表示框架，并对不同任务和场景进行分析实践，促进基于深度学习信号的表示算法的发展和应用。

1.2 研究现状与问题

信号表示算法的在实际应用中的重要性越来越凸显。基于传统方法的信号表示算法具有很好的普适性。无论对于什么样的任务或者应用场景，它都能很好地对信号进行表示。然而，也正是因为其通用性，导致它在特定场景上不能完全挖掘到信号内在的特点。信号内容的表示常常涉及到多个维度，语音信号可能包括人声、语言、语义、情绪和信道信息等；雷达信号可能包括脉冲重复周期、脉冲到达角、信号载频、调制类型等；音乐信号可能包括人声音色、曲风、情绪、主题和语言等。多个维度的信号表示，体现了信号不同角度描述的多义性。除了内容表示以外，信号表示常常还涉及到关系表示，比如信号模拟、人声模仿、相似歌曲等等的关系。在研究通用的信号表示时，我们可以尽可能地考虑这些信号的内容信息，并依据下游任务的需要从中提取关注的信息,这会体现为某种约束；也可以从某一具体的方面入手，对信号进行表示，得到匹配特定任务的信号表示。传统的信号表示方法不能很好地分解信号，体现出信号多维度

的特性。而基于数据本身的深度学习信号表示方法则可以更好地解决这个问题。本文中，基于编解码理论的信号表示算法能够被分为三个类别，分别是信号纯表示、脉冲信号解码表示和连续信号解码表示。这样的分类原因会在正文中详细说明，它可以辅助我们更系统地组织和解决基于深度学习编解码理论的信号表示问题。我们将在本节中分别介绍每类问题的研究现状。

早在二十世纪六十年代，信号表示算法就已经是人们研究的重点之一。1965年，Cooley，James W 和 Tukey 等人提出的快速傅里叶变换算法（Fast Fourier Transform，FFT）[31] 使得离散傅里叶变换的实现变得接近实时，并让其应用领域拓展到了数字图像处理、数据采集、现代雷达、故障检测记录等各个方面。它是数字信号处理的基石，也是所有传统信号表示算法的基石。其衍生出的短时傅里叶变换（Short-Time Fourier Transform，STFT）[32]、梅尔倒谱系数（Mel Frequency Cepstral Coefficients，MFCC）[33] 等表示算法也被广泛运用于信号处理领域。信号稀疏表示则是基于统计学习和模式识别的另一个角度的信号表示研究[2]。它指的是在给定的超完备字典[34] 中用尽可能少的原子来表示信号，进而得到信号更为简洁的表示。信号稀疏表示方向的研究热点主要集中在稀疏分解算法[35-36]、超完备原子字典等方面。

信号纯表示问题主要应用于信号分类、信号识别等领域，最初主要通过信号的统计量来作为信号的表示[37]。在音乐信息检索领域，信号纯表示问题涉及最多。音乐信息检索（Music Information Retrieval，MIR）是多媒体信息检索的一个子领域，在过去二十年中，研究领域不断扩大。到目前为止，在 MIR 的大部分研究中，音乐信号表示仍然主要是使用信号处理技术从音频中提取的。然而，在挖掘和推荐当今不断增长的数字音乐存储库中的音乐时，基于信号内容的算法迄今尚未在大规模系统中得到成功应用。事实上，通过协同过滤方法和使用上下文元数据的音乐信号表示和检索系统具有更高的用户接受度，但基于内容的音乐信号表示算法依然是十分具有发展前景的[38]。基于内容的音乐检索技术如何设计更好的表示方法来表示音乐信号是音乐信息检索领域的一个经典话题，基于相似性的音乐推荐也广泛应用于商业音乐应用。在以前的工作中，有两种类型的音乐信号表示方法。第一种是基于关系的信号纯表示方法，它基于社会关系[39]、用户-音乐关系[40-41]、音乐共现关系等[42-43] 定义信号的相似性；第二种是基于内容的信号纯表示方法[44]，它利用音乐内容和元信息对信号进行建模，

并在隐特征空间中计算它们之间的距离，以确定相似度[45-51]。基于内容的信号表示不能直接建立音乐信号表示之间的相似关系，而基于关系的算法存在长尾问题，很难达到最好的性能。由于纯表示问题中不需要将表示再解码为信号，所以通常不关心重建信号的损失。在信号纯表示问题上，需要探索一种算法对信号表示进行面向任务的约束，从而使得其表示更加契合具体应用需求，并能够很好地结合内容和关系两方面的表示。

脉冲信号解码表示问题则主要应用于雷达、水声信号的识别、分选和跟踪等方面，自二十世纪八十年代起就得到人们的关注[52]。雷达脉冲信号的分选是其中一个重要的课题。为了确保实际应用中的识别和分选准确性，雷达信号分选方法使用脉冲描述字参数中的一个或多个用于表示雷达脉冲信号。传统的雷达脉冲信号解码表示和分选算法，如累积差分直方图法（Cumulative Difference Histogram Method，CDIF)[53]、序列差分直方图法（Sequence Difference Histogram Method，SDIF)[54] 和改进直方图法（Improved Histogram Method，IHM)[55] 等，均具有清晰直观的理论，但对环境噪声影响和信号抖动、缺失不够鲁棒。基于脉冲描述字的信号表示聚类算法包括 K-means 聚类[56]、模糊聚类[57]、支持向量聚类[58]、层次聚类[59] 和高斯混合聚类[60-61] 等等。这种类型的算法主要基于相同脉冲信号具有相似参数分布的假设，并且通过设计脉冲描述字特征之间的距离来执行聚类。然而，聚类方法主要关注参数的相似性而不是信号表示的相似性，因此它可能无法很好地完成目标任务。Gasperini 和 Stefano 提出将雷达脉冲信号编码成图像，然后基于图像分割算法进行聚类[62]。然而，也存在诸如聚类的时间分辨率低和难以确定聚类簇的数目等问题。基于特征轨迹的预测方法[63] 主要使用与脉冲幅度相结合的循环神经网络（Recurrent Neural Networks, RNN)[64-65] 来执行信号分类。它是受到卡尔曼滤波器和霍夫变换分选算法的启发[66]。当脉冲信号能被精确测量时，它可以取得良好的结果。但当脉冲参数中的任何一个缺失，或出现误导性干扰时，表示和分选结果往往会波动。基于去噪自动编码器（Denoising Autoencoder，DA）算法可以很好地对多模脉冲信号进行解码表示[67]。然而，只有当雷达脉冲重复间隔和雷达脉冲信号分量已知时，这些算法才能很好地执行。在实际环境中，这种知识在大多数情况下是不存在的。脉冲重复间隔变换算法是一种传统的单参数脉冲信号分选方法[68]。因为它只能使用到达时间信息，时常不具有足够的信息量。脉冲信号是稀疏信号，多个脉冲分量很容易在

合适的特征空间中（如脉冲重复间隔自相关变换域）中被分选出来[69]。传统的脉冲重复间隔变换方法首先需要除到达时间之外的其他脉冲描述字参数进行预分选[70]。在分选过程中，不仅需要高时间分辨率（约 10 纳秒）及自相关函数的计算，还需要序列搜索（Sequence Search，SS）和人工选择阈值进行辅助。序列搜索是指使用估计的脉冲重复间隔信息来顺序地搜索与当前估计脉冲重复间隔一致的脉冲信号点，并根据搜索结果更准确地估计脉冲重复间隔。它常常与直方图形式的脉冲信号表示结合进行信号分选。目前，基于深度神经网络的脉冲信号解码表示算法[71-72] 对于仅具有到达时间特征的复杂雷达脉冲信号的表示和分选建模得更好。

连续信号解码表示问题最主要研究和应用在语音信号表示和处理领域，比如语音识别、语音合成和语音增强等等。语音增强算法也是实时音视频、线上会议等应用的关键。传统的单通道语音增强算法主要基于滤波算法进行语音信号表示和增强，比如谱减法[73]、自适应维纳滤波[74] 和其他自适应滤波算法。多通道语音增强还会使用多个麦克风的空间信息和自适应波束形成算法，如广义旁瓣消除（Generalized Sidelobe Cancellation，GSC）[75] 和最小方差无失真响应（Minimum Variance Distortionless Response，MVDR）[76] 来减少噪声。这些相对传统的算法在过去甚至现在都发挥了关键作用，尤其是在实时性要求较高的场景中。然而，这些连续信号解码表示算法在大多数情况下只能处理相对稳定的噪声信号，如白噪声，并且对突发噪声的适应性较弱。此外，仅从信号层面来估计噪声，这不是语义的，所以降噪效果比较一般。随着深度学习技术的不断发展，语音增强的研究也取得了长足的进步。RNNoise 是一种基于递归神经网络的实时语音增强算法。它结合了传统方法和深度学习的优点，至今仍被广泛使用[77]。深度学习语音信号解码表示和增强则可以分为时域增强和时频域增强。它通过短时傅里叶变换对语音信号进行编码，再通过短时傅里叶逆变换对语音信号表示进行解码。它引入了先验知识并将信号映射到稀疏正交空间中。时域增强则是通过一维卷积神经网络直接对信号进行编码和解码。时频域语音增强可以分为基于映射的方法和基于掩膜的方法。基于映射的方法直接预测干净语音的时频谱，而基于掩膜的方法预测每个时频点的语音信号出现概率。以前的研究通常只将干净的语音幅度谱作为语音信号表示，而忽略了它的相位谱，例如理想二进制掩膜（Ideal Binary Mask，IBM）和理想比率掩膜（Ideal Ratio Mask，IRM）

方法。它显著地限制了语音信号解码表示模型的性能，并且估计的语音信号相位也具有显著的偏差。后来的算法基于复杂的频谱映射和复杂的理想比率掩膜，甚至使用复杂的网络进行语音表示的估计，使得语音增强性能显著提升[78]。之后，出现了各种性能良好的语音信号解码表示算法。在单通道语音增强方面，有基于 LSTM 和其他时序网络建模的 FullSubNet[79] 和使用高斯位置编码的基于自注意的 T-GSA[80]。TCN 算法则是基于时间卷积的模型[81]。参考语音合成算法，基于生成对抗网络（Generative Adversarial Network，GAN）的 HiFi-GAN 也被作为语音增强算法[82]。在多通道语音增强方面，有使用后处理 MVDR 神经网络的波束形成方法[83]，以及直接使用神经网络的广义波束形成方法[84] 等。

1.3 研究内容

传统信号表示算法有着不能明确表示信号多方面意义的局限性，在复杂任务中通常只能作为信号表示的基础。基于编解码理论的信号表示算法在多个不同的研究领域中通常是相互独立的，未能提取其内在共性，形成系统的信号表示框架。实际上，各个领域的信号表示算法有类似的内在逻辑，能够相互启发。本文研究的是时序信号的表示算法理论与应用，其中也包括整体信号中部分分量的表示算法。

本文基于编解码理论，形式化了信号表示理论框架，并将其分为信号纯表示问题、脉冲信号解码表示问题和连续信号解码表示问题三个类别。对于这三类问题，我们不仅会研究它们的共通性质，也会结合它们各自的特点使用不同的理论，解决信号表示理论面对的问题和挑战。在编解码信号表示理论框架的基础上，我们还将结合具体的应用场景对特定类型信号表示效果进行实验，在仿真数据和实际数据上验证算法的有效性。聚焦于信号表示理论的系统化，解决其中存在的缺陷，并将其与实际应用结合，本文的主要研究成果如下：

1. 基于编解码理论将信号表示问题形式化，分析其中的关键组成部分：编解码函数、任务相关变换函数、信号表示约束函数和目标函数。基于输入信号类型和解码函数的形式将信号表示问题划分信号纯表示问题、脉冲信号解码表示问题和连续信号解码表示问题。基于传统信号表示理论，我们在简单的仿真实验上验证了传统信号表示理论框架，并分析了其局限性。

2. 针对信号纯表示问题，本文提出了信号纯表示框架（Signal Pure Representation Framework，SPRF）。该框架在理论方面，创新性地将信号内容表示和信号关系表示两种主流的信号纯表示方法统一在一起，并基于信号子空间表示方法更好地解决信号的多义性问题。以音乐信号推荐任务为切入点，该框架的内容表示模块以信号时频谱作为输入，通过深度学习算法和音乐信号之间存在的共同艺人关系，获取其共艺人（co-author）表示和内容表示；针对内容表示的局限性，SPRF 的解纠缠模块结合音乐信号的元数据，基于注意力机制和信息加权组合算法，对内容表示进行解纠缠，分解到不同的子空间；SPRF 的关系表示模块从交互数据和音乐知识图谱中挖掘音乐信号之间的关系，并通过 GraphSAGE 建模信号关系表示。为了融合内容和关系表示，通过度量学习建立相似性损失，实现鲁棒、准确的信号纯表示建模。该理论被应用到音乐推荐的工程实践中，解决音乐推荐中存在的多义性问题和长尾问题。

3. 针对脉冲信号解码表示问题，本文提出了深度脉冲信号掩膜（Deep Pulse Signal Mask，DPSM）算法。该算法针对脉冲信号非平稳、多噪声、分量个数未知等问题，以有监督非负矩阵分解算法为基础，首次提出了递归表示网络。该网络以雷达脉冲信号分选任务为切入点，通过双路径注意力机制递归提取脉冲信号分量的系数掩膜，预测脉冲信号分量数目。针对非平稳脉冲信号特点，分别提出了两种特殊的交叉熵损失函数以进行置换不变性训练，解决了脉冲解码表示和分选中存在的通道置换问题。为了避免因雷达调制模式和未知分量数目影响而出现的歧义性问题，DPSM 算法在递归表示网络基础上增加了在线预处理微调和后处理重聚类模块，充分利用了脉冲描述字中的带噪参数信息。该理论被应用到雷达信号分选的工程实践中，进一步增加了雷达信号解码表示和分选的准确率。

4. 针对连续信号解码表示问题，本文提出了实时频域卷积时序网络（Real-Time Frequency Convolutional Recurrent Network，RTFCRN）。该网络聚焦于以轻量化的结构进行连续信号解码表示，并实现实时流式处理。以语音信号增强任务为切入点，该网络能够处理各种场景噪声和混响影响下的多通道语音信号。为了使连续信号解码表示具有因果性和实时性，我们设计了带缓存的因果空洞卷积模块对连续信号进行时频域编解码。接着通过门控循环单元提取时序信息，从而得到干净语音信号分量的复数理想比率掩膜表示。RTFCRN 缓解了信号相

位预测不准确的问题。基于改进的尺度不变信噪比和人耳客观听感设计的损失函数，使得连续信号的解码表示更符合人耳感知。该理论被应用到语音增强的工程实践中，针对原始网络参数量大和难以实时推理的问题，设计和应用了敏感度剪枝、模型蒸馏和量化等轻量化算法对网络结构进行压缩，从而实现了性能优异、实时率高的流式信号解码表示和增强算法。

1.4 本文的组织结构

图 1-1: 本文的组织结构脉络

本文围绕基于编解码理论的信号表示算法及其实际应用进行研究和讨论，将多个领域的信号表示算法纳入到同一基本框架下。在此理论框架下，我们将信号表示算法分为三个类别，并分析了不同类别信号表示算法的各自特点，提出了针对性的算法框架和模型。同时，我们聚焦每类算法的应用场景，解决实际应用中存在的信号多义性和复杂性问题，在提升算法效率的同时强化信号表示的准确性和鲁棒性。如图1-1所示，本文一共分为七章，第一章为绪论，主要介绍研究信号表示算法的背景、意义，并介绍了信号表示的起源及存在的问题，介绍了三类基于编解码理论的信号表示算法研究现状和存在的挑战。第二章为

相关工作，首先介绍了常见的传统信号表示算法，接着分类别重点介绍了目前的基于深度学习的信号表示算法及其应用领域。第三章为信号表示编解码框架和分类，主要将信号表示理论形式化为基于编解码理论的优化问题，并基于输入信号类别和解码函数的不同将其分为三种类型，针对每种类型的信号表示理论进行了简单实验和分析。第四章为信号纯表示研究，这章对信号纯表示框架进行介绍，基于度量学习统一信号内容表示和关系表示，并在音乐推荐任务场景下进行了算法的实验验证和讨论。第五章为脉冲信号解码表示研究，这章介绍了深度脉冲信号掩膜算法，包括递归掩码表示网络及算法预处理微调和后处理高斯混合聚类，并在雷达分选任务场景下进行了算法的模拟验证和说明。第六章为连续信号解码表示研究，主要介绍了实时频域卷积时序网络，并基于模型轻量化算法对网络参数量进行压缩，在语音增强任务场景下进行了算法的实际验证和参数量、实时性的测试和对比。第七章为本文章工作的总结，并对下一步的研究进行展望。本文中所用的数学符号的意义在每个章节之间相互独立，在每章第一次出现时具体给出其含义。

第二章　相关工作

信号表示是各个信号相关领域特别是信号处理领域最重要的研究方向之一，在二十世纪六十年代研究人员就开始了深入研究，并收获了许多能够应用于实际的成功。早期的信号表示主要是基于复变函数和数字信号处理的相关理论，主要关注于信号的变换。随着深度学习算法的不断发展和更新，目前基于深度学习编解码理论的信号表示算法已经逐渐成为主流。本章将会首先介绍传统信号表示方面的部分经典工作，接着分类别介绍基于深度学习编解码理论的典型信号表示算法。

2.1　传统信号表示方法

基于传统信号处理的相关算法是信号表示的早期方法，不仅包括傅里叶变换（Fourier Transform，FT）、短时傅里叶变换、小波变换（Wavelet Transform，WT）、希尔伯特变换（Hilbert Transform，HT）等基于时频分析的算法，也包括维纳滤波、自适应波束形成等自适应滤波算法，还包括非负矩阵分解、直方图变换等其它信号表示的算法。本节将会简单介绍这些传统信号表示方法。

傅里叶变换是一种将信号从时域变换到频域的信号表示方法，它在信号处理领域有广泛的应用。傅里叶变换在信号分析和线性时不变系统中起着重要作用。变换的有效性源于它依据线性时不变系统下的特征函数提供了一种独特的信号表示，即复指数。傅里叶变换被表示为：

$$h(w) = \int_{-\infty}^{\infty} x(t) e^{-jwt} dt. \tag{2-1}$$

其中 $h(w)$ 是信号频域表示，$x(t)$ 是输入的信号，并且 $j = \sqrt{-1}$。类似的，傅里叶逆变换被表示为：

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} h(w) e^{jwt} dw. \tag{2-2}$$

而离散傅里叶变换（Discrete Fourier Transform，DFT）被定义为：

$$h[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-j2\pi kn}{N}}, \quad k = 0, 1, \ldots, N-1. \tag{2-3}$$

同样的，$h[k]$ 是离散信号的频域表示，$x[n]$ 是输入的离散信号。在实际应用中，因为计算机只能处理有限长的离散信号，所以需要离散傅里叶变换，其计算量比较大。1965 年提出的快速傅里叶变换算法[31] 基于分治思想，利用离散傅里叶变换的周期性和对称性，通过蝶形运算简化计算，使得离散傅里叶变换的实现变得接近实时。FFT 使得信号频域表示被应用到了数字图像处理、数据采集、现代雷达、故障检测记录等各个领域。除了一些速度要求非常高的场合之外，这种信号表示算法基本上可以满足工业应用的要求，是数字信号处理的重要基石。

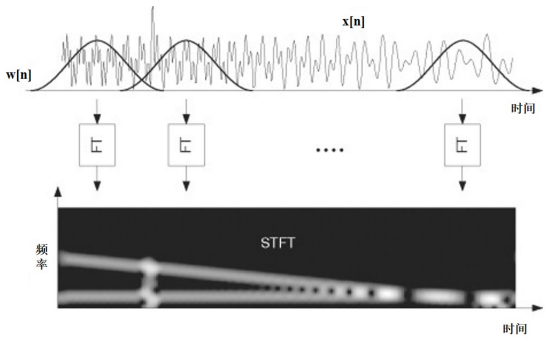

图 2-1: 短时傅里叶变换加窗和表示过程示意图

随着数字信号处理的不断发展，傅里叶变换也逐步不能满足现实中对信号表示的各种要求。仅仅通过时域信号或者幅度谱，很难分析出一段非平稳信号的特征。1946 年，Dennis Gabor 首次提出了窗口傅里叶变换，即后来被称为 Gabor 变换的短时傅里叶变换。短时傅里叶变换算法[85] 研究的是对短时平稳信号的表示，它能够平衡和兼顾时域和频域的信息，在信号的可视化、声音识别和处理等领域有着众多应用。它是加窗信号的傅里叶变换，为信号的频率分量随时间变化的情况提供时间局部化的频率信息，而标准傅里叶变换仅仅提供在整个信号时间间隔上平均的频率信息[86]。不同的窗长度得到的信号时频表示的带宽也是不同的。长的时间窗常被用于计算窄带时频表示，短的时间窗则被用于计算宽带时频表示。窄带时频表示有着比较高的频率分辨率，但时间分辨率较低。良好的频率分辨率可以让信号的每个谐波分量更容易被辨别，在时频图上以水平条

纹的形式出现。短时傅里叶变换对被表示为：

$$h[t,k]=\sum_{n=0}^{N-1}x[n]w[n-t]e^{\frac{-j2\pi kn}{N}}, \tag{2-4}$$

$$x[n]=\sum_{t}\sum_{k}h[t,k]w[n-t]e^{\frac{j2\pi kn}{N}}. \tag{2-5}$$

其中 $w[n-t]$ 是 N 点的窗函数，STFT 也可被解释为乘积 $x[n]w[n-t]$ 的傅里叶变换。它的加窗分帧过程如图2-1[86] 所示，图中的 $g(t)$ 同样表示窗函数，$x(t)$ 对应公式中的 $x[n]$。$h[t,k]$ 是时间和频率的二维函数，它将信号的时域和频域联系起来，从而能够基于信号表示进行时频分析。

短时傅里叶变换算法的时频分辨率是固定的，不能满足在低频部分具有高频率分辨率，在高频部分具有高时间分辨率的需求。Jean Morlet 在 1982 年提出了小波变换的概念，并提供了一个地震波分析的新数学工具。基于小波理论的小波变换[87] 能够提供一个随频率改变的时频窗，能够得到分辨率可变的时频信号表示。小波理论的现代应用多种多样，如波传播、数据压缩、信号处理、图像处理、模式识别、计算机图形学、飞机和潜艇的检测、CAT 扫描的改进以及一些其他医学图像技术等。小波是一种快速衰减的振荡，其等效数学条件为：

$$\int_{-\infty}^{\infty}|\psi(t)|^2\,dt<\infty, \tag{2-6}$$

$$\int_{-\infty}^{\infty}|\psi(t)|\,dt=0, \tag{2-7}$$

$$\int_{-\infty}^{\infty}\frac{|\Psi(w)|^2}{|w|}dw<\infty. \tag{2-8}$$

$\psi(t)$ 被称为母小波或基本小波，而 $\Psi(w)$ 是它的傅里叶变换。而小波变换则被定义为：

$$h(a,b)=\frac{1}{\sqrt{a}}\int_{-\infty}^{\infty}x(t)\psi^*(\frac{t-b}{a})dt. \tag{2-9}$$

母小波伸缩和平移变换后得到的函数族被称为分析小波，也是小波变换的基函数。ψ^* 表示分析小波的共轭。参数 a 是缩放参数或比例，它测量压缩程度。参数 b 是确定小波的时间位置的平移参数。如果 $|a|<1$，则分析小波是母小波的压缩版本，主要对应于较高的频率。而当 $|a|>1$ 时，则分析小波具有比母小波更大的时间宽度，并对应于较低的频率。因此，基于小波的信号表示具有与其频

率相适应的时间宽度。小波变换一定程度上解决了时频表示的分辨率问题，然而它依然受到海森堡不确定原理的限制。另一方面，小波变换的小波基通常是手动选取的，这表示其并不是一个自适应信号表示算法。希尔伯特变换[88] 和希尔伯特黄变换（Hilbert-Huang Transform，HHT）[89] 在一定程度上克服了这个缺点，具体细节不再赘述。

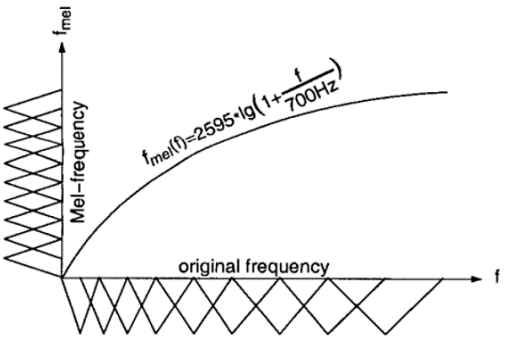

图 2-2: 梅尔频谱变换和滤波器组示意图

在语音信号处理、雷达信号处理、水声信号处理等不同领域，各自有一些独特的信号表示算法。梅尔频率倒谱系数（Mel Frequency Cepstrum Coefficients，MFCC）是在二十世纪八十年代由 Davis 和 Mermelstein 等人提出来的声学表示[90]，该方法被广泛应用于语音信号表示中，是短时傅里叶变换进一步的滤波和倒谱表示。研究表明，人类听觉只聚焦在频谱中某些特定的区域，而梅尔频率分析就是基于该特点而设计的。Davis 等人受到人类听觉系统的启发设计了梅尔滤波器组。如图2-2[91] 所示，梅尔标度是频率的非线性变换。从而，梅尔滤波器组是由频率坐标轴上的一系列滤波器组成的，其中大量的滤波器在低频段密集分布，少量的滤波器在高频区域稀疏分布，这与人耳耳蜗的滤波器十分相似。类似的，LOFAR（LOw Frequency Analysis and Recording）谱和 DEMON（Dtection of Envelope Modulation On Noise）谱常用于水声信号表示中[92]，而直方图表示和自相关表示则在雷达信号处理中很常见[93]。不同领域在比较通用的信号表示算法基础上，会在表示中融入领域的相关知识。

信号表示常常与信号分量的分解或分离紧密相关。信号分量的表示同样广泛存在于各个领域。以最常见的降噪为例，凡是涉及到信号增强或者降噪的任务，必然需要从带噪信号表示中分离出干净信号的表示，或者估计噪声表示。而噪声表示或干净信号表示通常依赖于滤波算法，特别是自适应滤波。维纳滤波[94]

是 Norbert Wiener 在 1942 年为了解决控制问题而提出的滤波算法。维纳滤波器又被称为最小二乘滤波器或最小平方滤波器，是利用平稳随机过程的相关特性和频谱特性对带噪信号进行滤波的方法。频域维纳滤波表示为：

$$H(w) = \frac{X(w)}{X(w) + N(w)}. \tag{2-10}$$

其中 $X(w)$ 是干净信号功率谱表示，$N(w)$ 是噪声功率谱表示，而 $H(w)$ 则是频域维纳滤波解，据此能够求解出干净信号的频域表示。由于干净信号和噪声通常都是未知的，维纳滤波一般需要估计噪声功率谱，并迭代求解[95]。自适应滤波是近 30 年以来发展起来的信号处理方法，它能够自适应地对信号分量进行表示。它是在维纳滤波和卡尔曼滤波等线性滤波基础上发展起来的一种最佳滤波方法。它具有更强的适应性和更优的滤波性能，从而在工程实践中得到了广泛的应用。递归最小二乘（Recursive Least Squares，RLS）就是实现自适应线性滤波的其中一种快速算法[96]。

除了滤波算法，为了从信号表示中分解出有效的信号分量表示，常用的方法是基于谱的分解方法，如独立分量分析（Independent Component Analysis，ICA）和非负矩阵分解（Non-negative Matrix Factorization，NMF）。自从 Lee 和 Seung 提出非负矩阵分解算法[97] 以来，研究者们陆续提出了一系列基于非负矩阵分解的信号表示算法，有监督非负矩阵分解就是其中之一。它首先提取信号 x 的短时傅里叶谱表示 X，接着对时频谱 X 执行非负矩阵分解，如果 W 是基矩阵，而 H 是系数矩阵，那么分解如下：

$$X = WH, \quad \text{s.t.} \quad W, H \geq 0. \tag{2-11}$$

分解后使用系数矩阵 H 表示信号矩阵 X 可以降低信号矩阵的维数，获得信号表示的降维矩阵，从而减少存储空间。所以非负矩阵分解的基本原理是将信号分解为非负的基矩阵和相应的系数矩阵，然后根据每个分量对应的系数掩码表示，最终得到每个信号分量表示。假设信号有 K 个分量，那么在训练阶段基于监督知识训练每个分量的基矩阵 W_k，在测试阶段则固定 W_k，只计算系数矩阵 H_k，

从而获得每个信号分量的表示 X_k：

$$X = \sum_{k=1}^{K} X_k = \sum_{k=1}^{K} W_k H_k \tag{2-12}$$

$$X_k = \frac{W_k H_k}{WH} \odot X. \tag{2-13}$$

其中 $\odot$ 表示 Hadamard 乘积。此外，还有许多改进的非负矩阵分解方法，例如结合主成分分析（Principal Component Analysis，PCA）和非负矩阵分解提出的投影非负矩阵分解算法[98]、进一步增加稀疏性约束的稀疏非负矩阵分解算法[99]等。这些算法基本都基于信号的时频表示，所以对于脉冲信号的支持不太好。

2.2 基于深度学习的信号表示方法

传统的信号表示方法一般是为了凸显出信号中的某方面特性，基于这种先验知识或者偏好，人为地选择合适的基函数和变换函数，从而表示信号。它与基于深度学习的信号表示方法并不是非此即彼相互对立的。基于深度学习的信号表示方法通常是传统方法的更进一步，其输入通常是传统方法预处理过的信号表示。在此基础上，深度学习结合大量信号数据，从统计的角度进一步结合信号特性，其信号表示在特定任务上能够取得更好的结果。在本节中我们会分别介绍三种信号表示算法中的典型工作。

2.2.1 信号纯表示

基于编解码理论的信号纯表示主要包括两个方面的研究，一是如何对信号的多义性进行分解和表示，二是如何选择合适的算法从数据中获得信号表示。Lee 等人将条件相似网络（Conditional Similarity Networks，CSN）[100] 应用于基于三元组损失的音乐领域深度度量学习中[46]。其主要思想是，每个掩码在表示空间中对应于不同的相似性语义维度，对应于不同的概念，如音乐信号中的流派、情绪、乐器和速度。这表明，解纠缠的信号表达不仅能够实现多维度的搜索，并且通过其子维度，还可以改善得到的检索信息。然而，用于解纠缠的音乐信号表示学习的条件相似网络仅探索了深度度量学习策略，而基于分类的方法没有考虑到。考虑到两者之间具有密切关系，后续研究也会关注分类下的解纠缠问题，

特别是对于多标签信号数据。本文中，我们主要关注音乐推荐任务。

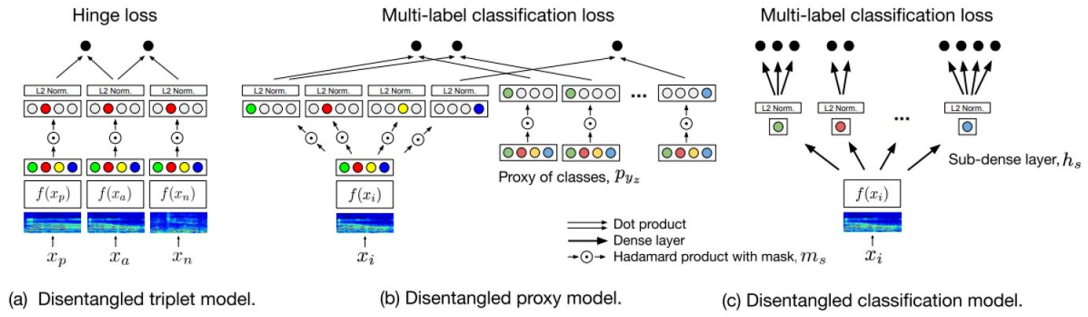

图 2-3: 信号解纠缠和纯表示算法

如图2-3[45] 所示，距离度量学习的目标是得到一个表示空间，在该空间中，相似的目标相互靠近，不同的目标逐渐远离。一个常见的策略是使用成对的样本或基于三元组的样本来训练模型。深度度量学习的一个重要优点学习是，它可以有效地模拟非常多的类的数量。然而，使用这种策略的模型训练得相对较慢，因为模型对输入样本的三元组进行操作。近来，研究者们提出了更有效的采样方法来加速模型的收敛，包括负样本挖掘，距离加权采样和基于代理的训练等。基于代理的训练为每个类分配一个或多个代理，由每类嵌入质心给出并优化。通过比较嵌入输入样本的学习空间而不是直接将其与正负样本进行比较。这显著减少了训练时间，同时提高了信号的检索性能。另一方面，分类模型通常被训练为使得类在深度神经网络的最后一个隐藏层的表示空间中是线性可分离的。由于分类模型没有在基于学习的表示空间中对距离进行优化，因此当直接用于基于相似度的信号表示时，它们可能表现不佳。为了克服这一问题，有的研究工作提出在训练期间在嵌入空间上应用归一化层，并表明这种简单的技术提高了基于相似性表示的模型性能[45]。

在基于三元组的度量学习中，我们将三元组定义为 $t = (x_a, x_p, x_n; y_z)$，其中 x_a 是锚点样本，x_p 是正样本，x_n 是负样本。x_a 和 x_p 具有相同的正样本标签 y_z，而 x_n 的标签 y_z 是负样本标签。那么，基本的三元组损失定义为：

$$L = \max(0, D(f(x_a), f(x_n)) - D(f(x_a), f(x_p)) + \Delta) \tag{2-14}$$

其中 $D(f(x_a), f(x_n)) = \cos(f(x_a), f(x_n))$，为距离度量，而 Δ 是阈值参数。假设

解纠缠的掩码表示为 m_s，那么解纠缠的三元组损失则可以表示为:

$$L = \max(0, D(f(x_a) \circ m_s, f(x_n) \circ m_s) - D(f(x_a) \circ m_s, f(x_p) \circ m_s) + \Delta) \quad (2\text{-}15)$$

其中 $\circ$ 表示哈达玛积。基于代理的度量学习的核心思想是代理学习嵌入并分配给每个类，从而测量到锚点到数据点的距离，而不是直接测量到成对或三元组数据的距离样本。这可以解释为一种监督聚类算法，其中代理扮演类中心的角色。在这种方法中，距离度量变为:

$$D(f(x_i), p_{y_z}) = \frac{f(x_i)}{\|f(x_i)\|} \cdot p_{y_z} \quad (2\text{-}16)$$

其中 x_i 是数据样本点，而 p_{y_z} 是类别 y_z 的代理。从而基于代理的损失函数为:

$$\hat{y}_z = sigmoid(D(f(x_i) \circ m_s, p_{y_z} \circ m_s)), \quad (2\text{-}17)$$

$$L = \sum_z \left[(-y_z log(\hat{y}_z) - (1 - y_z) log(1 - \hat{y}_z)\right]. \quad (2\text{-}18)$$

而基于分类的度量学习核心思想是在嵌入表示空间上应用归一化层。这种简单的技术确保了学习到的信号表示具有单位长度，并使基于相似性的检索比其它普通分类模型更有效。因此，每个类的基于分类的度量学习模型的预测得分为:

$$\hat{y}_z = sigmoid(\frac{f(x_i)}{\|f(x_i)\|} \cdot c_{y_z}) \quad (2\text{-}19)$$

其中 c_{y_z} 是每个类的中心，一般是神经网络的最后一个隐藏层。其损失函数是多分类交叉熵损失，而基于多任务学习的分类模型相当于解纠缠基于代理的模型，同时其实现要简单得多。

2.2.2 脉冲信号解码表示

基于编解码理论的脉冲信号解码表示主要包括聚类算法和序列预测算法两种类别。聚类算法基于脉冲点之间特征的相似性进行聚类，忽略了脉冲信号的时序特性，并且通常需要指定聚类的个数，不如序列预测算法灵活，但其优点是能更好地融合脉冲信号中的多源信息。使用时间序列特征的表示方法为复杂脉

冲信号的表示提供了一种可能的方式。这些研究将脉冲信号解码表示和分选问题分为三个部分，包括脉冲分量信号、信号交织过程和分选过程。通常假设脉冲分量信号遵循一些概率分布模型，如高斯分布模型、马尔可夫链模型和隐马尔可夫模型等。这些研究为脉冲信号表示问题的制定和相应的解决方案提供了多样化的探索和理解。本文中，我们主要关注雷达脉冲信号去交错任务。

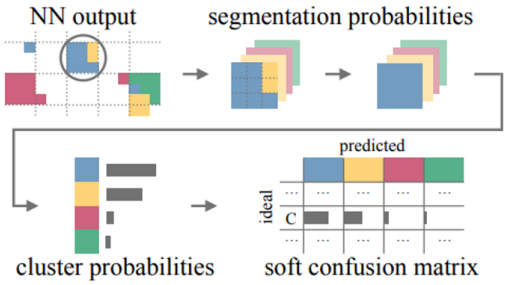

图 2-4: 脉冲信号聚类表示算法流程

在基于聚类的脉冲信号解码表示问题中，Gasperini 等人的想法是将该问题转化为图像分割问题，从该领域广泛的深度学习研究中获益[62]。图像分割是根据一些给定的标准将像素分割成区域的任务，即相似聚类，基于相似性对元素进行分组。如图2-4[62] 所示，该算法利用图像分割技术来执行聚类并解决脉冲信号的表示和分选问题。该方法使神经网络能够通过图像分割直接聚类输入元素。首先将信号编码为由谱图启发的可分割图像，然后将其输入到到经过图像分割训练的 U-Net 网络[101]，以对输入进行分组。神经网络通过调整的混淆矩阵（也被称为软混淆矩阵）中导出的损失函数进行优化。目标函数旨在提高聚类性能指标，如纯度等。软混淆矩阵 S 实际上是脉冲信号的原始表示，基于纯度的损失函数 L 表示为：

$$M(S) = \max\{C^j, \forall j\}, \tag{2-20}$$

$$Q(S) = \{S_i^j, S_i^j \notin M^j(S), \forall i, j\}, \tag{2-21}$$

$$L = \frac{\sum Q(S)}{\sum S}. \tag{2-22}$$

在基于序列算法的脉冲信号解码表示建模中，Zhu 等人则借鉴了机器翻译领域的方法，提出了基于深度学习的 NMT（Neural Machine Translation）模型[102]。图2-5[102] 描述了利用 NMT 模型的处理框架，它克服了传统时序算法的限制，

提供了一种简洁有效的方法，是一种良好的数据驱动框架。该 NMT 框架使用 seq2seq 学习和长短期记忆网络（Long Short Term Memory，LSTM）来实现。基于 NMT 分选的目标是预测脉冲信号 x_t 的标签序列，从而对脉冲点进行表示和分类。LSTM 及其变体已在包括脉冲信号分类在内的许多应用中得到广泛研究。为了捕获来自混合脉冲信号中相同分量的非相邻脉冲之间的结构关系，并为每个脉冲点分配相应的标签，NMT 框架采用了基于分层 seq2seq 的双向 LSTM（bi-LSTM）网络。bi-LSTM 层从前向和后向的相反方向迭代原始脉冲信号，生成脉

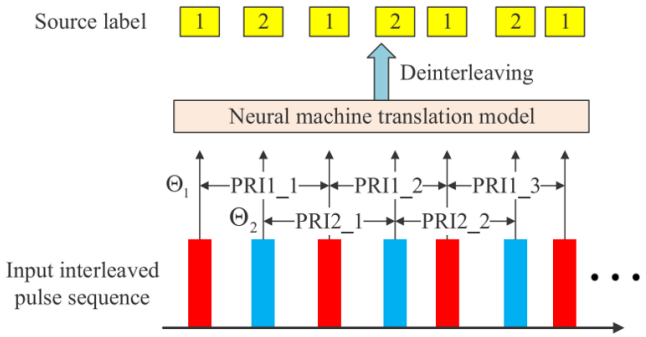

图 2-5: 基于 NMT 的脉冲信号分选框架

冲信号表示 $H^f = (h_1^f, h_2^f, \ldots, h_{T-1}^f)$ 和 $H^b = (h_1^b, h_2^b, \ldots, h_{T-1}^b)$:

$$h_t^f = LSTM(x_t, h_{t-1}^f), t = 1, 2, \ldots, T-1, \tag{2-23}$$

$$h_t^b = LSTM(x_t, h_{t-1}^b), t = T-1, T-2, \ldots, 1 \tag{2-24}$$

其中，$LSTM$ 表示由以下函数实现的 LSTM 单元函数:

$$f_t = \sigma(W_f x_t + R_f h_{t-1} + b_f), \tag{2-25}$$

$$i_t = \sigma(W_i x_t + R_i h_{t-1} + b_i), \tag{2-26}$$

$$a_t = tanh(W_a x_t + R_a h_{t-1} + b_a), \tag{2-27}$$

$$o_t = \sigma(W_o x_t + R_o h_{t-1} + b_o), \tag{2-28}$$

$$c_t = c_{t-1} f_t + a_t i_t, \tag{2-29}$$

$$h_t = \tanh(c_t) o_t. \tag{2-30}$$

其中 W 和 R 向量分别表示输入和递归网络的权重，b 则是偏差参数，f_t，i_t，o_t，a_t

分别表示遗忘门向量、输入门向量、输出门向量和输入向量，c_t 是单元状态。接着 H_f 和 H_b 被级联在一起以得到脉冲信号 x_t 的整个脉冲信号表示 $H = [H_f; H_b]$。

2.2.3 连续信号解码表示

基于编解码理论的连续信号解码表示分为时域算法与时频域算法两种，其核心都是对信号分量进行分解和表示。前者直接对时域信号进行表示，简单直接，通过神经网络学习得到信号表示的基函数避免了相位重建及其退化，但同时需要大量的数据才能得到较好的分量表示。后者在具备先验知识的时频分析表示基础上进一步学习更适合于任务的信号表示，学习难度相对较小，但也会受到信号相位表示不准等因素的影响。连续信号解码表示是深度学习和信号处理中的一个基本问题，特别是从单个混合信号中分离并表示多个信号分量。当要表示的信号分量属于同一类信号时，会出现额外的困难。例如分离并表示重叠的语音信号、隔离电气信号、从光曲线信号中识别多行星系中的系外行星、从光谱信号中检索化学混合物中的单个化合物等任务都特别困难，因为这些信号分量的来源性质十分相似。当然，从带噪语音信号中分离和表示出纯净语音信号，也就是语音增强时，不会出现这种情况。因此，设计一个在地面真实源和预测信道之间保持一致分配的模型对于具有类似源的任务至关重要。该研究主要应用于语音识别、语音分离、音乐分离等领域，我们主要关注语音分离和增强任务。

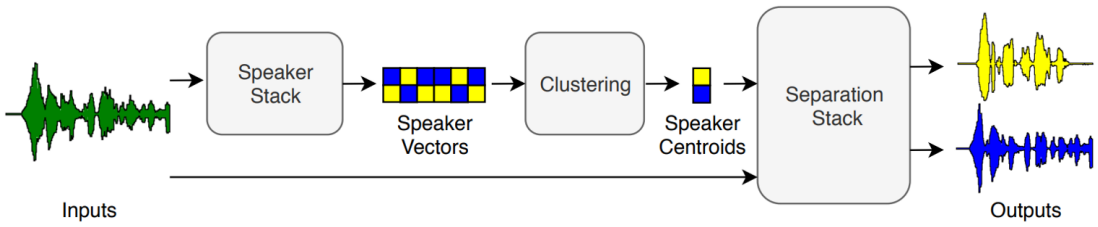

图 2-6: Wavesplit 说话人表示和分离框架

一种典型的语音信号分离和表示算法是由 Google 提出的 Wavesplit 算法，该算法希望能在测试时分离并表示新的语音信号分量，所以被称为开放说话人分离[103]。它训练期间利用了语音信号的说话人身份。该算法训练目标是识别瞬时说话人表示，使得这些表示可以被分组成单独的说话人聚类，并且聚类中心为单独的说话人信号，以提供长期的说话人表示。每个信号分量显式的长期表示提

取具有较好的创新性，有利于语音信号和非语音信号分离。这种表示改善了不一致的通道分配（通道交换）问题，这也是置换不变训练的常见类型。如图2-6[103]所示，Wavesplit 算法包括说话人向量表示和聚类，以及信号分离表示和分离两个部分。深度聚类方法设计了用于掩膜的聚类模型：该模型学习每个时频点的潜在表示，使得来自同一分量的时频点之间的距离低于来自不同源的时频点之间的距离。推理过程中将这些信号表示聚类，即按分量对时频点进行分组。置换不变训练通过搜索分量的排列组合，将预测的掩膜与真值掩膜进行比较，并将所有排列的最小误差用于训练模型。置换不变性训练承认语音分离中预测和标签的顺序是不相关的，即分离是一个集合预测问题。同时，Wavesplit 从短语音片段中提取辨别说话人表示以帮助表示信号分量，联合学习说话人表示和分量表示。

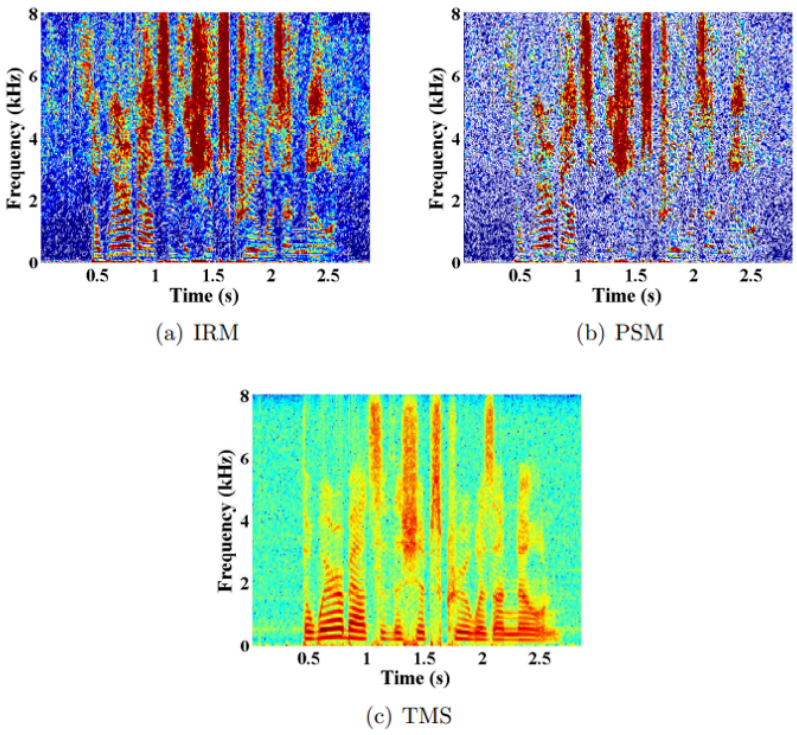

图 2-7: 几种常见的信号分量表示

因为时频域算法通常更易学习，所以在语音信号增强领域这种方法更常见。该类连续信号表示算法重要研究内容在于构造合适的掩膜或信号分量表示，以更好地学习和重建信号的相位，从而进行精确的信号分量表示[104]。如图2-7[104]所示，理想比率掩膜是有监督语音增强中广泛使用的训练目标，它可以被视为

理想二值掩膜的软版本：

$$IRM(t,f)=\sqrt{\frac{|S(t,f)|^2}{|S(t,f)|^2+|N(t,f)|^2}} \tag{2-31}$$

其中 $|S(t,f)|^2$ 和 $|N(t,f)|^2$ 分别表示时间帧 t 和频率帧 f 处时频单元内的干净语音能量和噪声能量。在基于掩膜的语音信号表示算法中，估计的时频掩膜表示被逐元素地乘以带噪语音的幅度谱，以产生增强的语音幅度频谱，该幅度频谱随后与带噪语音的相位一起，通过重叠相加方法重建增强语音的时域波形。相位敏感掩膜（Phase-Sensitive Mask，PSM）则将相位信息融合到时频掩膜表示中，并根据干净语音和噪声语音的 STFT 幅度谱进行定义：

$$PSM(t,f)=\frac{|S(t,f)|}{|X(t,f)|}\cos\theta \tag{2-32}$$

其中，$|S(t,f)|$ 和 $|X(t,f)|$ 分别表示时频单元内的干净语音和带噪语音的频谱幅度，θ 表示该单元内干净语音相位与带噪语音相位之间的差值。也可以通过深度学习直接表示干净语音的目标幅度谱（Target Magnitude Spectrum，TMS），即 $|S(t,f)|$。该方法是一种基于映射的信号表示方法。在基于映射的方法中，估计的幅度谱与噪声相位相结合，直接产生增强的语音表示。

2.3 本章小结

本章主要梳理和介绍了信号表示相关的算法。在传统信号表示算法中，首先介绍了几种典型的、不断发展的基于时频分析的信号表示方法，接着介绍了不同领域的场景信号表示以及对信号分量进行表示的相关工作，如滤波算法和谱分解算法等。在此基础上，为了后文更好地说明基于深度学习编解码理论的信号表示算法特点，我们还分三种类别分别介绍了现有的基于深度学习的信号表示工作：信号纯表示对信号的多义性进行分解，并从多模态信号中获得其表示；脉冲信号解码表示用聚类或者序列预测（分类）的方法融合脉冲信号中的多源信息；连续信号解码表示主要直接对时域连续信号或者在时频分析表示的基础上对连续信号分量进行更好的分解和表示。

第三章　信号表示编解码框架和分类

本章分为四个部分，第一个部分基于编解码理论对传统的信号表示问题进行形式化；第二个部分在第一部分的基础上，形式化基于深度学习的信号表示问题；第三个部分基于信号表示的编解码理论框架，对信号表示问题的常见类别进行划分；第四个部分在三个简单任务上介绍和说明形式化的信号表示编解码框架中的各个组成部分；最后一个部分对本章进行小结。

3.1　传统信号表示形式化

正如第一、二章所介绍的，本论文主要研究信号的表示及应用，既包括信号整体的表示，也包括信号分量的表示。本文的研究面向于具有时间性质的狭义信号，类似于图像、文字等其它信号不包括在内。我们首先从传统的信号表示方法视角对信号表示进行形式化。假设输入信号用 x 表示，基于编解码理论分别定义编码函数 $\mathcal{E}$ 和解码函数 $\mathcal{D}$。$h = \mathcal{E}(x)$ 是对输入信号 x 编码得到的隐向量，它是信号的整体表示或原始表示。f_θ 是对信号表示 h 的变换函数，其参数为 θ。它也是传统信号表示中的信号处理算法，比如各种形式的滤波器，参数量与机器学习算法相比非常少。而在经过 f_θ 处理后，得到的则是信号的分量表示或者信号的高层表示。相比信号原始表示，信号高层表示是经过了进一步处理和映射后的更抽象的信号表示。两者均是我们研究的对象，后续均会统称为信号表示。$\mathcal{D}(f_\theta(h), [x])$ 结合原始信号 x 的信息，对处理后的信号高层表示 $f_\theta(h)$ 进行解码。其中原始信号 x 的信息在有的问题中不需要，是可选项，用"$[x]$" 表示。之后计算解码向量或序列与正样本向量、序列或标签 y 之间的距离 $\mathcal{L}$，作为目标函数。最终，我们希望能够最小化两者间的 $\mathcal{L}$ 距离。这是所有信号表示问题均具有的形式，即：

$$\min \quad \mathcal{L}(\mathcal{D}(f_\theta(h), [x]), y), \tag{3-1}$$

$$s.t. \quad h = \mathcal{E}(x). \tag{3-2}$$

传统的信号处理方法通常是考虑将傅里叶变换函数和任务相关的滤波函数作为编码函数 $\mathcal{E}$。当解码函数 $\mathcal{D}$ 为恒等函数时，目标函数 $\mathcal{L}$ 通常希望滤波信号的频谱 h 接近于某种频谱形式 y，或是希望处理后的 $f_\theta(h)$ 与标签 y 尽可能接近；当解码函数 $\mathcal{D}$ 为傅里叶逆变换函数时，目标函数 $\mathcal{L}$ 希望重构信号 $\mathcal{D}(h)$ 接近于某种类型的时域信号 y。当然，如果编码函数 $\mathcal{E}$ 和解码函数 $\mathcal{D}$ 均为恒等函数，这种情况下是直接在时域对信号本身进行处理。本文研究的是信号表示问题，不考虑直接处理时域信号，即编码函数 $\mathcal{E}$ 不为恒等函数。

3.2 基于深度学习的信号表示形式化

由于本文主要的研究内容是信号的表示，所以重点在于编码函数 $\mathcal{E}$ 和优化目标距离函数 $\mathcal{L}$ 的构造。编码函数 $\mathcal{E}$ 考虑的是将一个信号 x 映射到什么样的隐空间。目标函数 $\mathcal{L}$ 考虑的则是解码的信号表示 $\mathcal{D}(f_\theta(h), [x])$ 与期望形式或标签 y 之间的关系，一般来说是最小化两者间的某种距离。解码函数 $\mathcal{D}$ 通常选择恒等函数或者转置卷积函数。

传统信号表示方法通常能够直接指定和设计合适的编码函数 $\mathcal{E}$，以及与任务相关的变换函数 f_θ。编码函数通常使用时频分析表示，比如傅里叶变换、小波变换和直方图变换等等。而表示变换函数 f_θ 通常也是传统信号处理算法，它们函数形式通常是指定的或者人为设计的，其参数相对于深度学习几乎可以忽略。在解决问题时，部分传统方法会依靠设计合适的滤波器或者寻找合适的算法逻辑来作为变换函数 f_θ，从而得到信号高层表示。

基于深度学习编解码理论的信号表示方法则不同，它一般不指定编码器的函数形式，而是直接将编码函数和信号处理函数参数化为 $\mathcal{E}_\theta$ 和 f_θ，有的任务中也会参数化解码函数 $\mathcal{D}_\theta$。同样的，深度学习中的 f_θ 通常也有着大量的参数。也正是因为没有约束编码解码函数和变换函数的形式，深度学习编解码理论缺少先验知识，通常需要大量数据，也比较容易过拟合。所以为了更好地求解优化问题，通常会加入一些与问题和任务相关的约束函数以简化问题表达。基于编解码理论的信号表示问题可以被形式化为以下优化问题：

$$\min \quad \mathcal{L}(\mathcal{D}_\theta(f_\theta(h), [x]), y) \tag{3-3}$$

$$\begin{aligned} s.t. \quad & h = \mathcal{E}_\theta(x) \\ & N(\theta) \leq k, \\ & G_1(g^{(1)}_{\theta'}(h, [x]), y_1) \leq \delta_1, \\ & \cdots, \\ & G_n(g^{(n)}_{\theta'}(h, [x]), y_n) \leq \delta_n. \end{aligned} \tag{3-4}$$

其中 $N(\theta)$ 表示参数量，k 是参数量的限制。这是因为现实存在的资源限制和运行效率限制而引入的约束。而 $G_i(g^{(i)}_{\theta'}(h, [x]), y_i) \leq \delta_i, i = 1, \ldots, n$ 则是与信号类型和任务特点强相关的约束条件，即先验假设。不同的信号类型和任务通常会对信号的表示 h 做出不同的先验假设。

3.3 信号表示问题分类

在研究信号表示问题时，其形式会因两个因素而产生较大的差别：是否存在解码函数 $\mathcal{D}_\theta$ 以及信号 x 的类型。依据解码函数 $\mathcal{D}_\theta$ 的形式可以将信号表示问题分为两种类型：信号纯表示和信号解码表示。在恒等解码函数下（等价于不使用解码函数），我们只关注信号原始表示 h 本身，经过 f_θ 处理后，信号原始表示将会成为高层表示。特殊的是，在纯表示问题中，高层表示仍然是信号的整体表示，而不是分量表示。并且，此时我们不需要将信号表示再次映射回与信号 x 相同的空间中。所以这个问题被称为纯表示问题。而在卷积函数等其它的解码函数的处理的情况下，f_θ 通常会从信号整体 h 中分解出所需的分量，此时的信号高层表示也被称为分量表示。接着，将信号的分量表示 $f_\theta(h)$ 从隐空间映射回信号所在原空间，这种情况被称为解码表示问题。

在纯表示问题下，无论输入的是什么类型的信号，均可以通过类似的编码函数对其进行建模和优化。因为我们不需要将信号表示 h 映射回原信号空间并进行优化，所以无论原信号 x 是什么类型，表示问题的形式差别不大。但同样因为是在隐空间优化，为了更好地建模隐变量 h，信号表示问题所需的约束条件 G_i 通常比较多。而在解码表示问题下，由于需要将原始表示 h 或者是分量表示

$f_\theta(h)$ 映射回原信号所在的时域空间，此时连续信号和脉冲信号的建模和优化方法会有较大不同，这主要体现在目标函数 $\mathcal{L}$ 的选择上。对连续信号来说，我们会将整个序列视为一个整体，最常使用回归和重构算法进行优化；而对脉冲信号来说，我们会将序列中的脉冲点视为多个有时序关系的样本，最常使用分类聚类算法进行优化。另一方面，在解码表示问题下，通常还会考虑模型复杂度和实时性约束。

因此，整个信号表示问题的研究被划分为了三种类别：（脉冲信号或连续信号）纯表示问题，脉冲信号解码表示问题，连续信号解码表示问题。本文中，我们将对这三种问题及其应用场景分别进行研究。

3.4 传统信号表示实验

为了更清晰地介绍信号表示研究问题，我们将在传统信号表示的三种类别上做了三个简单的实验。本实验中不涉及到基于学习的算法，仅仅只使用传统信号处理算法。第一个实验研究信号纯表示，任务目标是区分三角波信号和正弦信号，对两种信号表示进行分类；第二个实验研究脉冲信号解码表示，任务目标是分选两个发射源的混合脉冲序列；第三个实验研究连续信号解码表示，任务目标是对被高斯白噪声污染的正弦信号进行降噪。

3.4.1 信号纯表示实验

在信号纯表示实验中，输入信号 x 是理想的正弦信号或三角波信号，信号表示 h 是频谱幅度谱。时域的两种信号虽具有显著特征，但不够直观。通过表示信号 x，可以更容易地对两种信号进行分类。在本实验中，编码函数 $\mathcal{E}$ 是傅里叶变换：

$$h(w) = \mathcal{E}(x(t)) = \int_{-\infty}^{\infty} x(t) e^{-jwt} dt. \tag{3-5}$$

其中 t 是时间变量，w 是频率变量。

为了对正弦信号和三角波信号分类，我们定义判别函数 f：

$$f_\theta(h) = \mathcal{I}\left(\frac{b(h)}{a(h)} > \theta\right). \tag{3-6}$$

其中 $\mathcal{I}$ 是指示函数,a 和 b 两个函数分别从信号表示 h 中提取基波的幅度和三次谐波的幅度。参数 θ 是三次谐波与基波幅度比率的阈值。

定义三角波信号的标签 y 为 1，正弦信号标签 y 则为 0. 在该实验中，不需要进行梯度计算和更新，优化目标 $\mathcal{L}$ 可以直接表示为:

$$\mathcal{L}(f_\theta(h), y) = \mathcal{I}(f_\theta(h) = y). \tag{3-7}$$

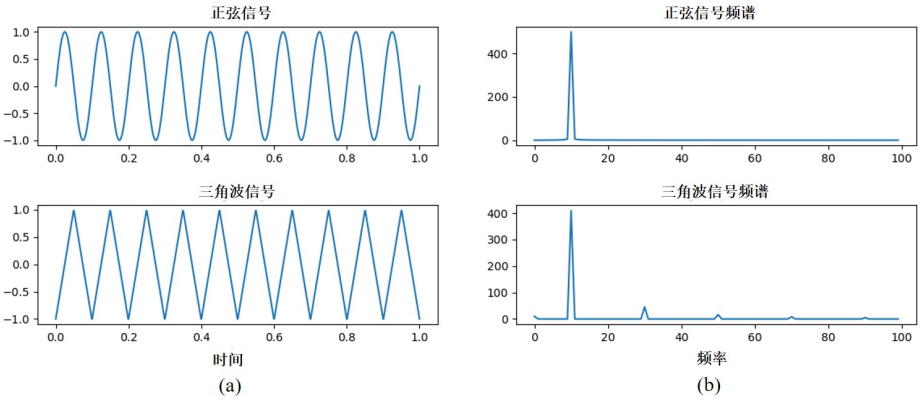

图 3-1: (a) 10HZ 正弦波和三角波; (b) 10HZ 正弦波频谱和三角波频谱

如图3-1所示，直接在时域上对正弦信号和三角波信号进行区分相比频域是困难的，因为信号的频域表示具有良好的稀疏性质。在频域上，我们能够找到幅度最大的频率点，从而确定信号的基频。再通过基频确定三次倍频，从而获得三次倍频幅度。而三角波频谱的三次倍频与基频强度之比远远大于正弦信号频谱的三次倍频与基频强度之比。在确定了阈值参数 θ 后，该算法能够在信号频域表示 h 上完美区分两种理想信号，并且也有一定的抗噪能力。在本实验中，$\theta = 0.01$。

3.4.2 脉冲信号解码表示实验

在脉冲信号解码表示实验中，输入信号 x 是交错的两发射源脉冲信号，而信号表示 h 则是 k 阶差分直方图。因为两列脉冲信号混杂在一起，所以从时域上无法轻易将其区分开来。通过累计差值直方图算法（Cumulative Difference Histogram, CDIF), 能够更容易地对固定脉冲重复周期 (Pulse Repeat Interval, PRI)

的交错脉冲序列进行分选。在本实验中，编码函数 $\mathcal{E}$ 是直方图差分变换：

$$h^{(T)}[k] = \mathcal{E}(x[m]) = C_k(x[m+T] - x[m]). \tag{3-8}$$

其中 T 是差分的阶数，k 是直方图的差分值，m 是脉冲序列时间下标。C 是直方图累积函数，计算差分序列中 k 值的数量。为了对交错的脉冲信号进行分选，我们要在直方图估计信号的 PRI，并进行序列搜索。这个任务中的变换函数 f_θ 是 PRI 估计算法：

$$P = f_\theta(h). \tag{3-9}$$

参数 θ 则是直方图差分值的阈值。经过 f_θ 的处理，我们能够从直方图表示中计算出两个脉冲序列的脉冲重复周期 P。而解码函数 $\mathcal{D}$ 则是序列搜索算法。通过脉冲重复周期从原始信号 x 中确定原始序列的点 S 和脉冲点数 N 从而还原出脉冲序列：

$$\hat{x} = \mathcal{D}(P, x). \tag{3-10}$$

在这个实验中，y 表示原始的两个脉冲信号，而优化目标的距离函数 $\mathcal{L}$ 是汉明距离：

$$\mathcal{L}(\mathcal{D}(f_\theta(h), y)) = \sum \mathcal{I}(\hat{x} \oplus y = 1). \tag{3-11}$$

$\hat{x} \oplus y$ 表示分选脉冲信号与原始脉冲信号的异或。

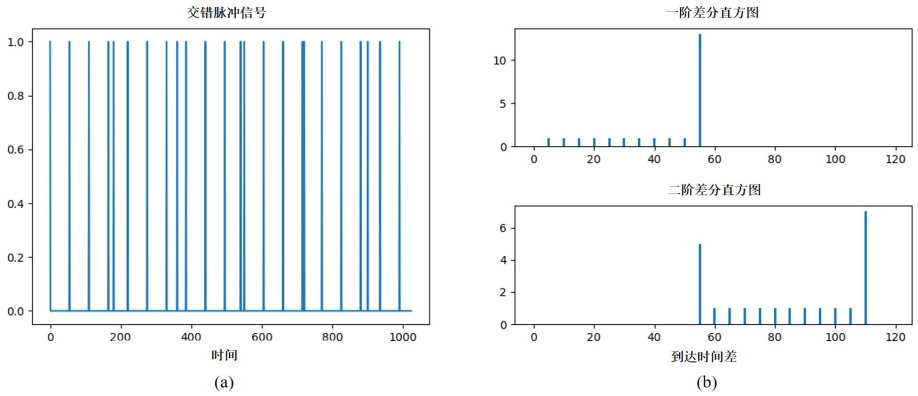

图 3-2: (a) PRI=55 和 PRI=180 的两列交错脉冲信号; (b) 交错脉冲信号一阶和二阶差分直方图

图3-2表示了两列交错的脉冲信号，它们的 PRI 分别为 55 和 180，单位是微秒。图 (b) 中上面的图是一阶差分的直方图，可以看出在 55 微秒处的累积值远

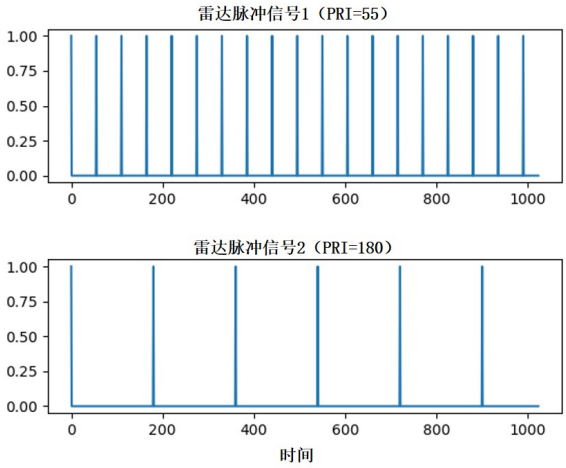

图 3-3: 分选的两列独立脉冲信号

远大于其它差分值。下面的图则是二阶差分直方图，在 55 微秒和 110 微秒处均出现了较大的脉冲。在本实验中，如果在某一间隔的统计值及其二倍处均超过了阈值 $\theta = 4$，那么 CDIF 算法会以该间隔作为潜在 PRI 进行序列搜索。于是在交错序列 x 上以 55 微秒为间隔进行序列搜索，得到起始时间点为 0，脉冲数目为 19，从而分选得到 PRI 为 55 微秒和 180 微秒的脉冲序列。在本实验中，分选算法能够完美地将成功地将两个交错脉冲序列分开，分开的两列脉冲信号如图3-3所示。

3.4.3 连续信号解码表示实验

在连续信号解码表示实验中，输入信号 x 是被白噪声污染的正弦信号，信号表示 h 和编码函数 $\mathcal{E}$ 与信号纯表示实验一致。直接在时域对带噪的正弦信号进行降噪是很困难的。而在频域上信号的特点则十分明显。频域降噪的函数可以被定义为:

$$h' = f_\theta(h) = R_{\theta_1,\theta_2} \cdot h. \tag{3-12}$$

其中 R_{θ_1,θ_2} 是参数为 θ_1 和 θ_2 的矩形窗理想带通滤波器。滤波结束后，用解码函数 $\mathcal{D}$ 将处理后的信号表示重新映射回时域:

$$\hat{x}(t) = \mathcal{D}(h'(w)) = \frac{1}{2\pi}\int_{-\infty}^{\infty} h'(w)e^{jwt}dt. \tag{3-13}$$

在这个实验中，y 是干净的理想正弦信号。在距离函数用均方误差时（Mean Squared Error，MSE），优化目标函数被定义为：

$$\mathcal{L}(\mathcal{D}(f_\theta(h), y)) = \sum(\hat{x} - y)^2. \tag{3-14}$$

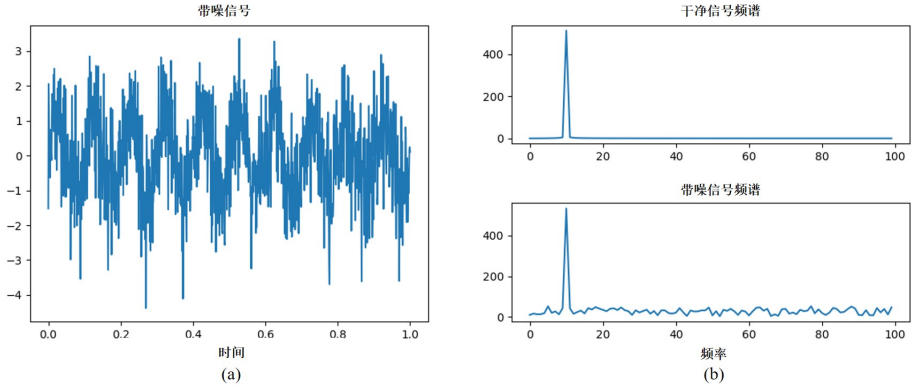

图 3-4: (a) 带噪的正弦信号; (b) 干净正弦信号和带噪正弦信号的频谱图

图3-4中 (a) 展示了带噪正弦信号的波形，是比较杂乱无章的。而在频域上，信号表示的特征相当明显。正如 (b) 中展示的那样，正弦信号只有单个频率。在找到该频率后，将其余频率全部置为 0，再将信号变换为时域，就能够完美地降噪。图3-5展示了原始信号和降噪信号的波形。当 $\theta_1 = \theta_2 = 10$ 时，能够完美还原理想的 10HZ 正弦信号。当 $\theta_1 = 5, \theta_2 = 15$ 时，降噪得到的波形仍有一定的变形。

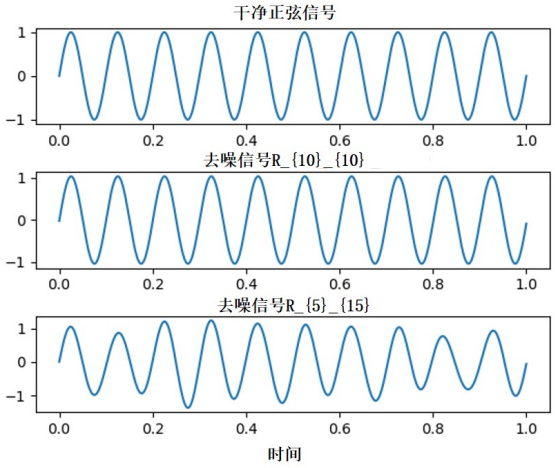

图 3-5: 10HZ 正弦信号及使用两组参数进行滤波后的去噪正弦信号

3.4.4 实验总结

本节通过三个简单的信号表示实验，介绍了传统信号表示编解码框架。虽然本实验中的问题比较简单，但仍能够发现解决信号表示优化问题的几个关键：编解码函数，信号表示变换函数，目标函数。从实验中能够发现，在传统信号表示问题中参数很少，通常能够直接人工设计。但同时，它能够解决的问题也相对简单。当信号本身或者应用场景变得复杂时，无论是信号分类、分选还是降噪都会变得困难。为了解决这些更加复杂的问题，在本论文的后面三章，我们将会基于深度学习研究如何解决三类信号表示问题。

3.5 本章小结

本章基于编解码框架形式化了信号表示问题，并从传统信号表示问题的形式化推广到基于深度学习的信号表示问题的形式化。基于解码函数和信号类型，将信号表示分为了信号纯表示、脉冲信号解码表示和连续信号解码表示问题。通过三个简单实验分别说明了三种信号表示问题的具体形式，并预期通过后续章节的深度学习算法研究和解决更复杂的问题。

第四章　信号纯表示研究

本章分为八个部分，第一个部分描述总体介绍信号纯表示框架 SPRF 及其核心思想，并介绍音乐信号推荐任务；第二个部分介绍 SPRF 中基于内容表示的子模块组成, 并简略介绍音乐共艺人关系与艺人表示；第三个部分介绍 SPRF 的解纠缠的子模块组成，并介绍先验关系；第四个部分介绍 SPRF 的关系表示子模块和图采样和聚合算法 (Graph SAmple and aggreGatE, GraphSAGE)；第五个部分介绍用于 SPRF 各个模块训练的，统一在度量学习框架下的相似性损失，特别是多重关系损失函数；第六个部分将信号纯表示框架纳入到信号表示优化的理论中，说明信号纯表示理论在音乐推荐问题上的具体形式；第七个部分在音乐推荐和艺人推荐两个实际应用任务上使用 AllMusic 数据集评价音乐信号表示的精确性、有效性和鲁棒性；最后一个部分对本章进行小结。

4.1　信号的纯表示

在本章中，我们主要关注信号纯表示问题，并提出了信号纯表示框架。它将不同影响因素拆分到不同正交子空间中，通过多关系损失融合信号间的关系，更好地对其信号的多义性进行建模，保证信号纯表示的准确性和鲁棒性。在实验和分析部分，我们会在音乐推荐任务及其下游艺人推荐任务上验证 SPRF 的有效性，所以本章的信号纯表示研究主要关注音乐信号，其它信号仍然可以使用类似方法建模。

4.1.1　信号纯表示的两种范式

在前面的相关工作介绍中，我们提到了常用的基于信号处理的信号表示方法，也说明了这些算法是通用但却不够专用的信号表征方法。分类或者聚类等基于编解码理论的信号表示算法能够从数据中学习到相对专用的信号表示，但却不能够解决信号的多义性问题，也不能很好地表达信号与信号之间某些确定性的关系。

目前的信号纯表示研究主要包括两种范式，一种是基于内容的方法，以音乐信号为例，主要基于音乐的曲风、旋律、节奏等定义音乐的相似性；另一种是基于关系的方法，以音乐信号为例，主要是基于共同艺人关系、歌单共现关系等对音乐的相似性进行定义。在本章中，不会单独考虑内容表示或者关系表示，而是通过度量学习将两者有机地结合在一起，构建统一的框架。信号纯表示的优劣一般通过相似推荐任务来进行衡量。它的目标是给定任意信号，输出 K 个与之最为相似的信号。对于音乐来说，研究相似音乐推荐有利于了解音乐作品的创作与传播，也能让用户对感兴趣的音乐进行更加方便的检索。

如果将信号关系看作一个无向图，那么信号内容表征就是图上的节点，而关系则表示为图上的边。为了得到信号的表示，首先需要合适的初始化节点表示，以尽可能地将信号解纠缠到不同子空间，从而从不同的角度定义信号，并构建约束 G_i。这保证了信号表示的鲁棒性。接着要结合信号之间的关系信息，沿着图的边进行信息传播，使相似的信号表示更加相似。这强化了表示的表达能力。

4.1.2 信号子空间表示方法

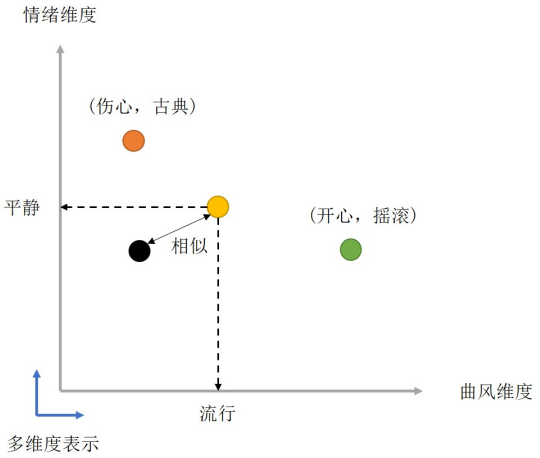

图 4-1: 音乐信号多义性示意图

如图4-1所示，音乐信号作为信号的一种，同样具有多义性。而信号子空间表示方法就是为了从多义的信号中提取出关注的部分表示。这个提取音乐纯表示的过程涉及到子空间的解纠缠和训练两个关键步骤。

在本章中，如图4-2所示，我们将音乐信号表示定义在四个正交子空间上，分别表示艺人、语言、曲风和人气。假设音乐内容表征为 E_C，并且预先人为定

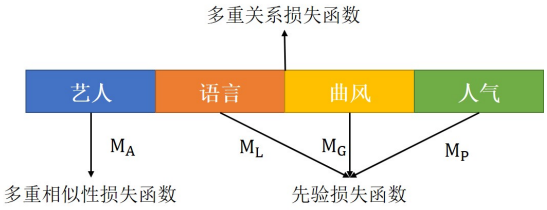

图 4-2: 音乐的子空间表示

义的艺人、语言、曲风和人气的子空间掩膜为 M_A, M_L, M_G, M_P, 从而有:

$$E_A = M_A \circ E_C, \tag{4-1}$$

$$E_L = M_L \circ E_C, \tag{4-2}$$

$$E_G = M_G \circ E_C, \tag{4-3}$$

$$E_P = M_P \circ E_C. \tag{4-4}$$

其中 $\circ$ 表示逐元素乘积。子空间掩膜都是 $0-1$ 向量，其意义是从表征向量 E_C 中提取出表示音乐信号特点的正交的向量，并在不同意义的表征中融入不同的信息，使用不同的损失函数进行训练。在嵌入空间，每个掩码对应着不同的语义维度，这使得解纠缠的音乐表示不仅可以在总体上对音乐相似性进行解释，也同样能够实现某个语义维度的表示和检索。

信号子空间的训练范式可以分为三种，度量学习、聚类和分类[45]。这三种范式都可以通过先融合信号信息，再对信号的隐空间解纠缠的方式，一定程度上解决信号的多义性问题。结合这种解纠缠的思想，把分类、聚类和度量学习统一到相似性损失的通用框架下，可以更好地训练得到信号的表示，解决纯表示问题。

4.1.3 信号纯表示框架

在以前的研究中，信号相似性维度的定义是单一和片面的。比如，音乐的表示一般会从音乐曲风这一维度进行定义。而我们将会从多个角度考虑影响信号表示的因素，并对其进行相应的建模。图4-3展示了 SPRF 的整个流程。SPRF 包括三个模块：内容表示模块、解纠缠模块和关系表示模块。

在本章的信号纯表示研究中，内容表示模块以信号的关键部分或者整段信

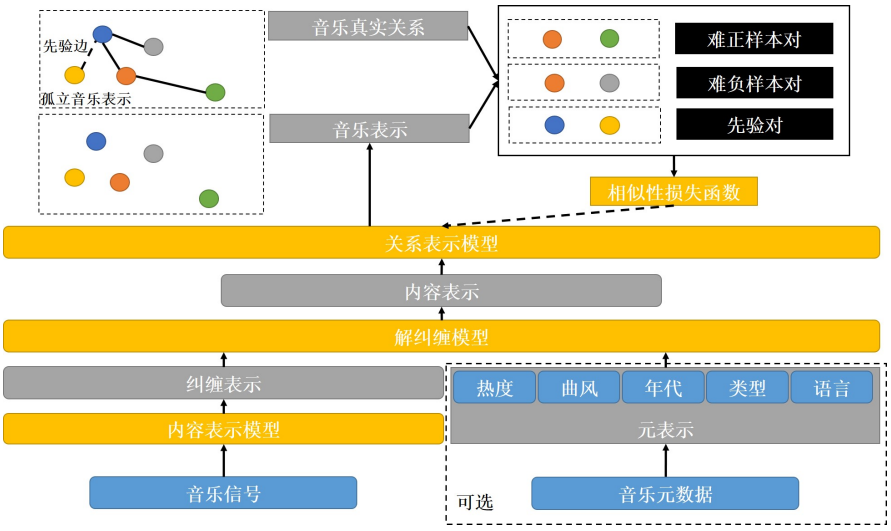

图 4-3: 信号纯表示框架图

号作为输入，并通过内容信息建模。比如音乐信号会以高潮部分作为输入，并通过音乐的共艺人关系和弱监督度量学习来建模音乐的艺人表示 E_A 它也能建立音乐表征和艺人表征之间的联系。解纠缠模块能够将信号的元信息编码为强化的内容表示。比如将音乐信号的热度、曲风、年代、类型和语言等信号信息编码为隐向量，并结合 E_A 进行特征加权和选择，融合不同维度的信息，最后将不同种类的信息准确提取并映射到不同的子空间，用先验损失函数和音乐相似性损失函数进行监督学习并实现解纠缠。在没有可用的音乐信号元数据的时候，这两个模块也可以合并为一个整体，直接从音乐信号中提取所需信息并进行解纠缠，得到音乐内容表征 E_C. 在关系表示模块中，考虑将信号的相似关系视为一个关系图。解纠缠的信号表示被作为图的节点，而信号的相似关系和从知识图谱中挖掘出的其它相似关系会被作为图的边, 比如音乐信号的歌单共现关系。通过图神经网络将信号的内容表示在图上进行传播和汇总，得到最终的信号表示 E_M。在训练时，结合特定的距离函数，从所有信号表示中挖掘出困难的正负样本对和先验样本对，结合相似性损失函数进行有监督训练。

4.2 内容表示模块

内容表示模块的性能表现直接决定了信号表示效果的下限。它输出的纠缠内容表示衡量了信号多个方面的特性。对音乐信号来说，包括歌曲音色、音乐风格、音乐艺人关系等特性。

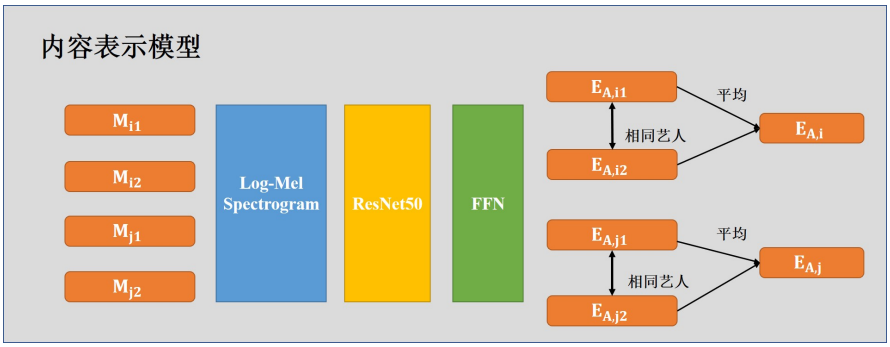

图 4-4: SPRF 的内容表示模块

内容表示模块如图4-4所示，在音乐信号表示建模中，它能够通过共艺人关系 R_A 来衡量音乐是否相关。音乐与演唱（演奏）的艺人之间是多对一的关系，显然同一个艺人的音乐之间具有显著的共性。这天然地定义了音乐信号之间的关系。假设总共有 I 个艺人，第 i 个艺人有 K_i 首音乐作品，那么 M_{ik} 表示第 i 个艺人的第 k 首音乐作品，这是内容表示模型的输入。

如果存在可用的音乐元数据，优先使用元数据对语言、曲风和人气进行编码。相对于从信号中提取特征，元数据的表征能够显著减少多义性，得到更可靠表征 E_L, E_G 和 E_P。此时内容表示模块仅从音乐信号中提取共艺人特征 E_A。否则，该模块将会从音乐信号中提取音乐表征 E_C，并通过后续的解纠缠模块进行特征解纠缠。

4.2.1 信号的时频谱

音乐信号的长度通常为 3 到 5 分钟，如果直接将其分段并输入内容表示模型，再取平均作为音乐的内容表示，可能会极大损失音乐信号本身的特点。一般的音乐前奏、主歌、副歌、过渡句和间奏等部分组成的。因为音乐的前奏、间奏等，相对来说并不能很好地体现艺人的特点。所以在实际处理时，会通过音乐高潮提取模型提取每首音乐最具特点的副歌部分。我们主要通过衡量重复率和旋律感来识别副歌，这部分不是本论文的重点，细节略过。

对低频声音比较敏感是人类听觉的特性。当声音频率线性增大时，频率越高，越难听出其中存在的差别。因此相对于直接使用时域音乐信号，梅尔对数谱是更容易表达信号特点的特征。梅尔谱包含三大特性：时域-频域信息，感知相关的振幅信息，感知相关的频域信息。正如许多音乐标注和情感识别任务所做

的那样，提取的音乐高潮片段会被变换到梅尔域，得到梅尔谱。接着通过对数变换，将梅尔谱的幅度单位转换为为分贝，从而计算出高潮部分的 Log-Mel 频谱 S_{ik}，即：

$$S_{ik} = logMel(M_{ik}). \tag{4-5}$$

4.2.2 骨干网络

内容表征模块用 50 层的残差网络 (ResNet50)[105] 作为骨干网络。在 ResNet 提出之前，大多数网络是通过卷积层和池化层的所组成的。理论上，当卷积层和池化层的层数越多，提取的信息越全面，学习效果也就越好。但是实际上，随着卷积层和池化层的增加，反而会出现梯度消失和梯度爆炸的问题。ResNet 是多个残差模块的串联，拟合的是上层网络的残差，很大程度上上解决了梯度消失和梯度爆炸的问题。

梅尔谱如同二维图像一般，也具有类似的 “纹理” 特征。为了融合局部的时域和频域信息，使用预训练的 ResNet 提取特征是比较合理的。在内容表征模块中，ResNet50 骨干网络负责提取梅尔频谱特征，接着通过前向传播网络（Feed Forward Network, FFN) 得到最终的内容表征。将音乐作品映射为音乐表征过程表示为:

$$E_{A,ik} = FFN(ResNet50(M_{ik})). \tag{4-6}$$

4.2.3 音乐信号的共艺人关系定义和艺人表征

为了得到音乐信号的共艺人表征 E_A，需要定义音乐关于艺人的相似关系，在这里我们将其定义为共艺人关系 R_A。定义第 i 个艺人的所有音乐之间具有相似关系，即当 $Y_{A,i} = Y_{A,j}$ 时，有 $Y_{M,i1} = ... = Y_{M,ik_i} = Y_{M,j1} = ... = Y_{M,jk_j}$。并且认为不同艺人的音乐是不相似的，即如果 $Y_{A,i} \neq Y_{A,j}$，那么 $Y_{M,ik_i} \neq Y_{M,jk_j}$。有了以上定义，就可以只使用音乐数据本身，结合度量学习采样困难的正负样本，进行自监督的度量学习，这部分将会在4.5节详细介绍。

另外，除了音乐表征以外，还能够得到艺人的表征。艺人表征也能对音乐信号的下游任务（比如相似艺人推荐）有所帮助。通过将每个艺人的所有音乐信号

表示 $E_{A,ik}$ 进行平均，从而将音乐表示聚合成艺人表示：

$$E_{A,i} = \frac{1}{K_i} \sum_{k=1}^{K_i} E_{A,ik}. \tag{4-7}$$

更好的办法是选择艺人最有代表性的作品进行聚合得到 E_A。在已知音乐热度的情况下，可以将热度最高的 K 首歌曲表征进行平均，即：

$$E_{A,i} = \frac{1}{K} \sum_{M_{ik} \in pop_i} E_{A,ik}. \tag{4-8}$$

其中 pop_i 是艺人 i 热度最高的 K 首歌曲的集合。

4.3 解纠缠模块

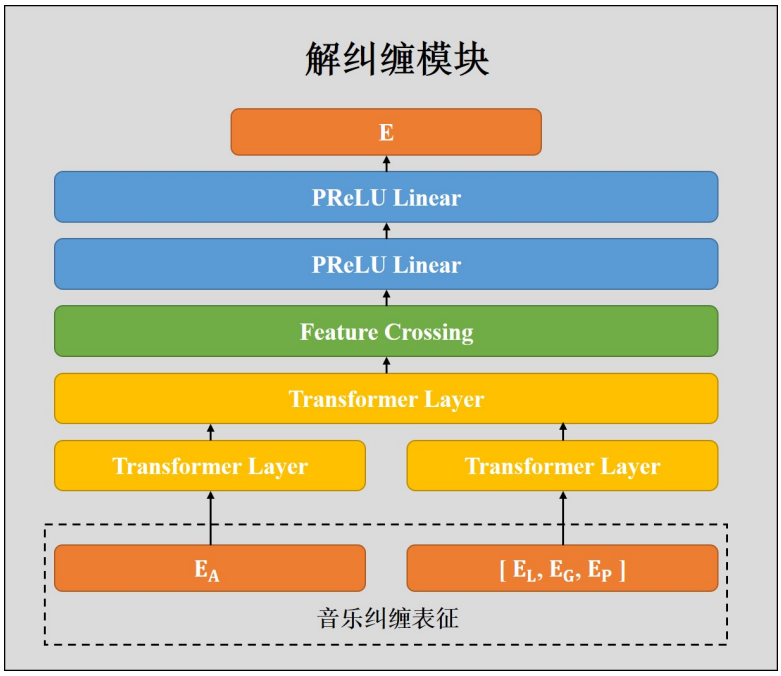

图 4-5: SPRF 的解纠缠模块

解纠缠模块是 SPRF 的重要组成部分。如图4-5所示，其输入是信号的纠缠表征，对音乐信号来说，其中包括音乐信号共艺人子空间表示 E_A 和未经训练的其它子空间表示 $[E_L, E_G, E_P]$. 内容表示模块从音乐信号中抽象出了音乐共艺人表示 E_A, 并从音乐信号或者音乐元数据得到了 $[E_L, E_G, E_P]$. 它们组合起来构成了音乐信号的初步表示，但这个表示存在一定的局限。如果 $[E_L, E_G, E_P]$ 是从

音乐信号中提取出来的，那么在内容表示模型中是通过共艺人关系 R_A 去训练它们的，此时它们的意义实际与 E_A 是耦合在一起的，表达的都是音乐的共艺人关系，而不是期望的语言、曲风和人气关系。如果 $[E_L, E_G, E_P]$ 是由元数据得到的，那么它们还只是初始化的、未经训练的表征，同样不能够很好地表示语言、曲风和人气关系。解纠缠模块正是为了解决以上问题而构造的，它能够先将信号的每个子空间表示进行融合、筛选和组合，最后通过合适的损失函数将信号不同方面的表示投影到相应的子空间中，以实现解纠缠的目的。

4.3.1 注意力机制

信号表示的交互能够为解纠缠做准备，在本章中，不同特征表示之间的注意力机制能够实现这个目的。注意力机制很像人类关注某件事物的逻辑。我们通常并不会仔细观察事物的每个细节，而是将注意力集中在某个部分。比如图片中人的脸部，文章中的标题以及或首句等位置。对于音乐信号也是一样的，它的副歌，或者一些能够体现曲风和人声特点的特征是更重要的，这反应的是人脑对音乐信号中关键特征的捕捉。

从另一个角度看，注意力机制也类似于计算机的软寻址，这个过程通过键、查询和值三个变量实现。将键视为地址，将值视为存储的数据，当需要进行查询的时候，通过键与查询的匹配，得到特定的值。与硬寻址不一样的地方在于，注意力机制不是直接寻找查询与键相等的位置，而是通过计算键与查询的相似度间接完成寻址，一般通过内积进行计算。Transformer 中使用的是自注意力机制，它与普通注意力机制的不同在于输入的查询与键和值均是由同一个特征进行线性变换得到的。

4.3.2 信息的加权和组合

信号多个子空间的内容信息均被输入到了解纠缠模块中，所以我们需要考虑如何衡量每种信息的重要性,从而实现特征的加权和筛选。如果把所有的特征都拆分到更细粒度的子空间中，再对这些子空间进行重新加权，就能实现信号特征的融合。所以，我们将所有子空间特征向量的维度 D_A, D_L, D_G 和 D_P 都定义为细粒度子空间大小 F 的整数倍，从而通过 Transformer 层计算每个细粒度子

空间之间注意力。

具体地，使用多头注意力机制将纠缠的信号内容表示拆分并映射到不同的子空间。给定某个子空间特征 $E^{(1)}$，我们先将其分为 N 个维度为 F 细粒度子空间，即 $E^{(1)} = [E_1, E_2, \cdots, E_N] \in \mathbb{R}^{N \times F}$，并通过自注意力机制对各个子空间特征重要性进行加权，从而整合信号的多个特征：

$$K, Q, V = Linear(E^{(1)}) \in \mathbb{R}^{N \times F}, \tag{4-9}$$

$$E^{(2)} = Softmax(\frac{QK^T}{\sqrt{F}})V. \tag{4-10}$$

接着再对 $E^{(2)}$ 进行非线性变换和批归一化 (Batch Normalization) 变换, 这里同样存在着残差连接操作:

$$E^{(3)} = BN(FFN(E^{(2)})) + E^{(1)}. \tag{4-11}$$

其中，BN 表示批归一化, FFN 是非线性变换。经过 Transformer 层的信号表示加权和融合后，还需要对多个维度的信号表示进行组合，以增加特征的表达力。组合特征也叫特征交叉，即让不同维度特征之间的交叉组合，主要目的是为了将融合的信号表示进行进一步组合交互，从而为解纠缠提供进一步的信息。

我们通过 Feature Crossing 层对多个细粒度子空间特征进行组合, 从而得到二阶特征:

$$E_{cf} = [\cdots, E_i^{(3)} \cdot E_j^{(3)}, \cdots], \forall i \leq j. \tag{4-12}$$

接着，再将其与原始特征拼接: $E^{(4)} = [E^{(3)}, E_{cf}]$. 最后经过参数化线性整流单元（Parametric Rectified Linear Unit，PReLU）激活的非线性变换后，得到最终的信号内容表征 E_C.

4.3.3 音乐信号的先验关系

对于音乐信号来说，在训练时，解纠缠模块依赖上一节所介绍的共艺人关系 R_A 来强化艺人表征 E_A. 除此以外，音乐先验关系 R_L, R_G 和 R_P 会将音乐特征中的语言、曲风和人气映射到合适的子空间。这里的先验关系是通过少量的有监督标签 Y_L, Y_G 和 Y_P 所定义的，可以将其视为子空间上的分类或聚类问题。

由此，使用共艺人关系定义的的多重相似性损失和不同的先验损失分别训练不同的音乐信号子空间特征，就能够在融合整体信息的前提下对音乐纠缠内容表示实现解耦。具体的损失函数同样会在4.5节详细介绍。

4.4 关系表示模块

解纠缠模块得到了信号的内容表示。正如第一节所介绍的，信号表示有两种范式，内容表示只是其中的一种，另一种是基于关系的表示。信号关系从两个角度建模，首先是训练目标，其次是显式表达。音乐信号的关系有着多种来源，比如从音乐知识图谱和用户交互数据中进行挖掘，也有专家标注的相似音乐关系。它们共同组成了真实的音乐相似关系 R，是音乐相似性图上的主要的信息边。在所有的相似关系 R 中，最为常用的音乐信号关系是共歌单相似关系 R_M，描述的是两首音乐是否同时出现在一个或多个歌单中。不过，这些音乐信号关系一般只能覆盖到少量音乐信号，所以音乐内容表示也同样是关键的，它们提供了大量的先验知识。前面描述的内容相关特征 R_A, R_L, R_G 和 R_P 也可以作为图上的先验边，此时这个图是一个异构的有向图。图神经网络能够有力地表达音乐之间的关系，所以关系表示模块的主体是图卷积算法。

4.4.1 图关系表示算法

图卷积又分为谱图卷积和空域卷积两大类型。谱图卷积是对图的拉普拉斯矩阵进行特征值分解，从而更好的找到图中的簇，代表性方法是谱图卷积 (Graph Convolution Network, GCN)。这种方法的缺点是时间复杂度较高，在数据量较大的情况下很难使用；它利用了图的整个邻接矩阵来融合相邻节点的信息，因此一般只能够用于直推学习任务，不能用于处理归纳学习任务，泛化性比较差。而空域卷积作用于节点的邻居节点，用一定数目的邻居节点来计算当前节点的属性。在关系表示模块中，我们使用的就是基于空域卷积的图采样和聚合算法。

基于 GraphSAGE 的图关系表示算法如4.1所示，其输入为解纠缠模块的信号内容表示 E_C，输出则是信号表示 E_M。该算法主要包括三个步骤：在信号关系图上对邻居进行随机采样，聚合邻居信号节点的表示和生成目标节点的表示。当然，在实际使用中会将 GraphSAGE 重复多次，使得当前节点能够融合高阶的邻

Algorithm 4.1 基于 GraphSAGE 的图关系表示算法提取信号表示 E_M

输入: 信号相似关系图 $\mathcal{G}$,
信号内容表示 E_C,
非线性函数 σ,
聚合跳数 K,
采样函数 $\mathcal{N}_k, \forall k \in 1, \ldots, K$,
聚合函数 $\mathcal{A}_k, \forall k \in 1, \ldots, K$,
模型参数 $\mathcal{W}_k, \forall k \in 1, \ldots, K$.
输出: 信号表示 E_M.

$E^K \leftarrow E_C$
for $k = K, \ldots, 1$ **do**
  $E^k \leftarrow E^{k-1}$
  for $h \in E^k$ **do**
    $E^k \leftarrow E^{k-1} \bigcup \mathcal{N}_k(h)$
  end for
end for
$h_0 \leftarrow h, \forall h \in E^0$
for $k = 1, \ldots, K$ **do**
  for $h_{k-1} \in E^{k-1}$ **do**
    $h^{\mathcal{N}_k} \leftarrow \mathcal{A}_k(\{h'_{k-1}, \forall h'_{k-1} \in \mathcal{N}_k(h_{k-1})\})$
    $h_k \leftarrow \sigma(\mathcal{W}_k \cdot [h_{k-1}, h^{\mathcal{N}_k}])$
    $h_k \leftarrow \frac{h_k}{\|h_k\|}$
  end for
end for
$E_M \leftarrow E^K$

居节点信息。生成目标节点一般通过神经网络进行实现即可，接下来将从邻居采样和聚合两个方面来介绍 GraphSAGE 算法。

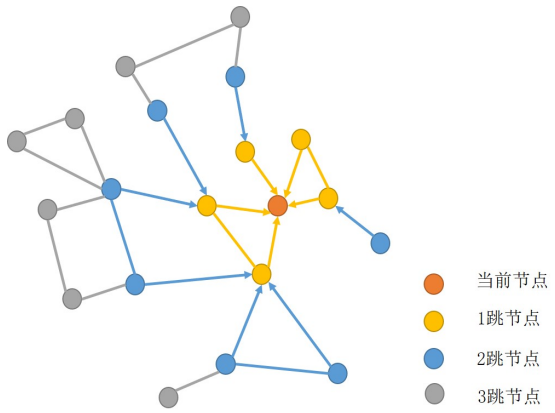

图 4-6: GraphSAGE 采样和聚合示意图

4.4.2 邻居采样

在 GCN 算法中，需要建立整个信号关系图的拉普拉斯矩阵，并在关系图上进行信息传播。但这意味着，每次迭代都要对整个图的所有节点进行更新。GraphSAGE 每次训练从所有内容表征 E_C 中采样部分节点作为一个小批量，后续在这些节点及其邻居构成的子图上进行更新，这对应着算法4.1中的 1 到 7 行。

邻居采样函数 N_k 负责从整个信号关系图中采样出所需的邻居节点。采样时主要关注两个参数，一是采样的跳数，跳数越多，当前节点关注的邻居阶数就越高，聚合的信息就越远；二是每跳采样的节点个数，通常对于不同跳数的邻居有不同的个数限制 N_k。设定不同的邻居阶数和采样跳数，GraphSAGE 算法可以采样整个信号关系图，也可以完全不聚合任何邻居的信息。图4-6中的 1 跳的黄色节点和 2 跳蓝色节点就是从整个图中采样得到的当前节点的邻居，其中 $K = 2, N_1 = 5, N_2 = 7$. 显然，随着 K 和 N_k 的增大，当前节点关注到的节点更多，但也同样会稀释最邻近节点的信息，实际中需要权衡两者的关系。另一方面也要选择合适的聚合函数，尽可能反应任务的本质，以获得更具表达能力的信号表示。

4.4.3 邻居聚合

邻居采样起到了减少计算复杂度的作用，同时也引入了随机性使得 GraphSAGE 算法具有一定的泛化性能。但如何将邻居的内容特征与当前节点的内容特征进行聚合，从而得到更具特点信号表示，是邻居聚合需要解决的问题, 其核心是邻居聚合函数 $\mathcal{A}_k$。

为了进行聚合，使用消息传递机制，由外而内地依次聚合信号内容表示，最终得到当前节点的信号表示。以图4-6为例，具体来说，先将 2 跳节点的内容特征 h_0 按照箭头方向聚合到 1 跳节点上，得到 h_1，再将 1 跳节点更新后的特征 h_1 聚合到当前节点上，得到 h_2, 也就是当前节点的信号表示 E_M. 这个过程的其核心代码是算法4.1中的 8 到 15 行。聚合的特征 h 与当前节点原始特征拼接，并通过非线性变换和归一化对当前节点特征进行更新。其中影响性能最关键的因素是聚合函数 $\mathcal{A}_k$ 的选择。

第一种常用的聚合函数形式类似于卷积神经网络中常用的池化方法，比如

对邻居节点的特征求平均:

$$h^{\mathcal{N}_k} \leftarrow Mean\left(\left\{h'_{k-1}, \forall h'_{k-1} \in \mathcal{N}_k(h_{k-1})\right\}\right). \tag{4-13}$$

或者是通过非线性变换后再进行最大值池化:

$$h^{\mathcal{N}_k} \leftarrow Max\left(\left\{\sigma(W \cdot h'_{k-1}), \forall h'_{k-1} \in \mathcal{N}_k(h_{k-1})\right\}\right). \tag{4-14}$$

第二种函数依然使用 GCN 的思想进行聚合, 依然会考虑当前节点原始特征:

$$h^{\mathcal{N}_k} \leftarrow Mean\left(W \cdot Mean(\{h_{k-1}\} \cup \left\{h'_{k-1}, \forall h'_{k-1} \in \mathcal{N}_k(h_{k-1})\right\})\right). \tag{4-15}$$

第三种函数是基于 LSTM 或者注意力机制的聚合，如:

$$h^{\mathcal{N}_k} \leftarrow LSTM\left(\left\{h'_{k-1}, \forall h'_{k-1} \in \mathcal{N}_k(h_{k-1})\right\}\right), \tag{4-16}$$

$$h^{\mathcal{N}_k} \leftarrow Attention\left(\left\{h'_{k-1}, \forall h'_{k-1} \in \mathcal{N}_k(h_{k-1})\right\}\right). \tag{4-17}$$

其中 LSTM 网络设计具有时序上的先验，也就是邻居输入 LSTM 的顺序会影响得到的特征。所以在训练时需要输入尽可能多的不同顺序的邻居节点集合，才能保证具有不变性。而注意力机制聚合函数则不存在这个问题，另一方面，它也能够考虑不同节点的重要性，给不同邻居加权，所以在关系表示模块中我们最终使用的是注意力机制聚合函数。

4.5 相似性损失

在前面几节中，我们说明了信号纯表示框架的各个模块的结构和功能，并且定义了多种关系 R_A, R_L, R_G, R_P 和 R_M, 分别表示音乐信号的共艺人、语言、曲风、人气和共歌单相似关系。不过尚未说明如何利用这些关系作为弱监督或者有监督的目标训练内容表示模块、解纠缠模块和关系表示模块。本节将基于度量学习中的多重相似度损失将这些关系统一起来，得到适用于所有模块的多重关系损失。

4.5.1 分类聚类与度量学习的统一

信号表示的三种训练范式在各个任务中都被广泛使用，其中分类和聚类作为有监督学习的典型方法，最常被用来进行表征学习。共艺人表示和解纠缠的整体信号表示通常是通过度量学习进行训练的，而语言、曲风和人气表征一般情况下会通过多标签分类的方法训练。这导致各个模块的训练目标并不是统一的，并且不同方法之间各有优劣。度量学习的方法太过依赖信号之间的关系信息，很容易导致过拟合；分类聚类的方法建模平滑，但却也比较粗糙，会有信息上的缺失，加剧不同维度表征之间的纠缠。所以，在 SPRF 下，它们均被纳入到度量学习的框架之下，相互补充。

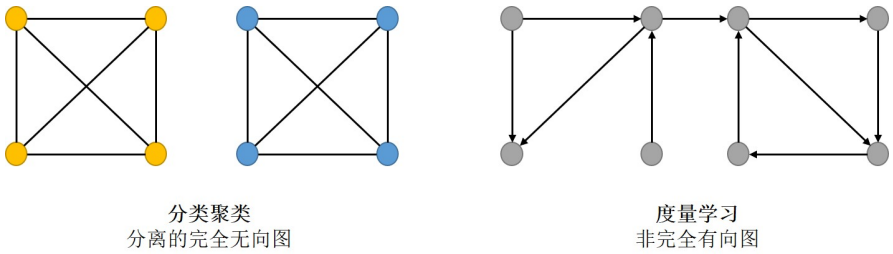

图 4-7: 基于图表示的表征学习范式

如图4-7所示，分类聚类问题实际上也可以被纳入到度量学习的范畴之中，作为它的一种特殊情况。分类聚类通常会的样本都会对应某个中心，聚类是簇的中心，分类则是以标签为中心。在同一个类别内的样本在这个类别子空间内，都具有完全相同的相似关系。即同一类的样本构成全连接的无向子图，而不同类别之间的样本完全没有连接。这实际上是特殊的图结构。而度量学习中，也可能会会存在这样的切图，实际上也可以看作是谱聚类的推广。此时度量学习的非完全的有向图，并且“簇”与“簇”之间可能也会有一些连接，“簇”内部的节点也很少会是全连接的。这里面表示了相似性的非对称结构，也是更加复杂的。所以这三种范式天然是可以统一在一起的。分类聚类完全图的特殊性导致其建模相对简单，度量学习的一般性导致其容易过拟合。

自然的，可以考虑将三种范式融合在一起并相互补充。基于这种思考，在下文中，我们将先验损失、多重相似性损失统一到度量学习框架下，并通过多重关系损失进行建模。多重相似度损失函数[106] 是一种对样本对关系进行建模的通用性度量学习框架，损失计算主要分为两个步骤：困难样本采样和样本加权。

4.5.2 困难样本采样

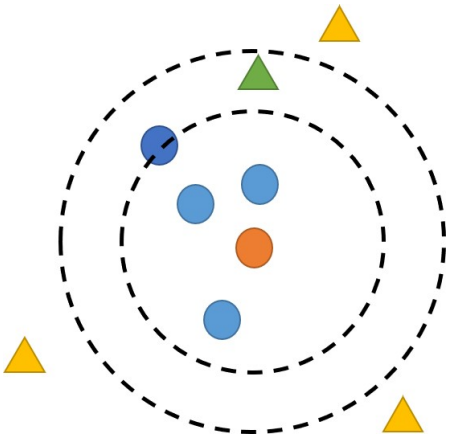

图 4-8: SPRF 的关系表示模块困难负样本采样

令 $X = \{A, M\}$，并将 N 首信号基于关系 R_X 表示 E_X 的相似性矩阵定义为 $S_X \in \mathcal{R}^{N\times N}$。其中 $S_{X,ij} = f(E_{X,i}, E_{X,j})$ 表示信号 i 的表示 $E_{X,i}$ 与信号 j 的表示 $E_{X,j}$ 在相似性度量 f 下的相似度。在计算关系 R_X 的度量损失时，我们使用多重相似度损失，它是度量学习最经典的方法是三元组损失的推广。它将当前需要更新的信号表示作为锚点，分别提取相对于锚点的困难正样本（即具有最低相似度的正样本对）和负样本（即具有最高相似度的负样本对），作为困难正负样本对。如图4-8所示，橙色圆点表示当前要更新的锚点，浅蓝色和深蓝色是相对于锚点的正样本，而黄色和绿色是负样本。为了挖掘困难负样本，我们以在相似性度量 f 下与锚点最不相似的深蓝色正样本的相似性为界限，认为在距离其一定范围以内的负样本均为困难负样本，例如示意图中的绿色负样本。具体地，我们结合相似性矩阵 S_X 和关系 R_X 来挖掘有信息的样本对。Y_X 是由关系 R_X 定义的标签。给定阈值参数 δ, 困难正样本对 $(E_{X,i}, E_{X,j})$ 的下标集合被表示为:

$$\mathcal{P}_{X,i} = \left\{ j \middle| S_{X,ij} < \min_{Y_{X,k}\neq Y_{X,i}} S_{X,ik} + \delta \right\}. \tag{4-18}$$

类似的，困难负样本对 $(E_{X,i}, E_{X,j})$ 的下标集合被表示为:

$$\mathcal{N}_{X,i} = \left\{ j \middle| S_{X,ij} > \min_{Y_{X,k}=Y_{X,i}} S_{X,ik} - \delta \right\}. \tag{4-19}$$

从而得到了困难正负样本集合 $\mathcal{P}_{X,i}$ 和 $\mathcal{N}_{X,i}$。

4.5.3 样本加权

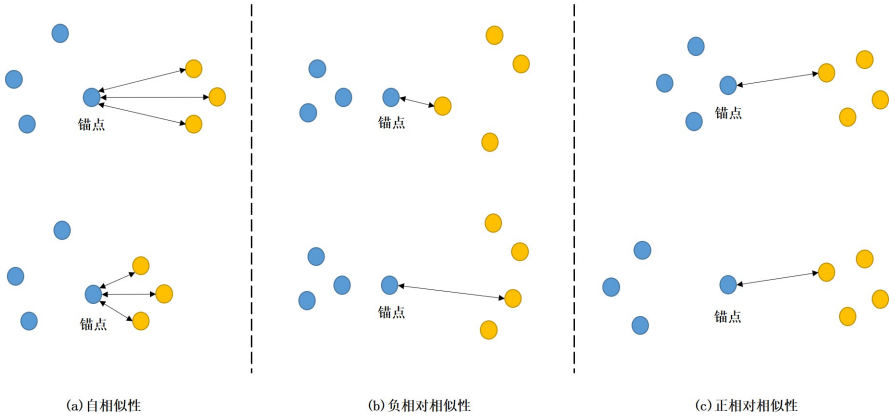

图 4-9: SPRF 度量学习中三种类型的样本对相似性

在经过困难样本采样过程之后，我们将抛弃对模型的训练没有较大帮助的，信息不足的样本。接着，需要考虑的是每个样本对于当前样本表征学习的误差权重。通过样本加权的方法，将不同样本分布下相似性关系纳入到考虑之中。多重相似度损失将样本相似性分为三类：自相似性，负相对相似性和正相对相似性。在图4-9中，从左到右展示了这三类样本相似性。其中，自相似性是锚点与正负样本之间的相似性，通过直接的计算得到。锚点与正样本的自相似性越大，与负样本的的自相似性越小，样本对越难被区分，从而携带有更多的信息和意义。负相对相似性考虑的是自身的相似度和其它负样本对相似度之间的差别。正相对相似性考虑的是自身的相似度与其它正样本对相似度之间的差别。依据以上三种相似性对挖掘的困难样本加权：

$$w_{\mathcal{P},ij} = \frac{1}{e^{-\alpha_1(\gamma_1 - S_{X,ij})} + \sum_{k\in\mathcal{P}_{X,i}} e^{-\alpha_1(S_{X,ik} - S_{X,ij})}}, \tag{4-20}$$

$$w_{\mathcal{N},ij} = \frac{1}{e^{\beta_1(\gamma_1 - S_{X,ij})} + \sum_{k\in\mathcal{N}_{X,i}} e^{\beta_1(S_{X,ik} - S_{X,ij})}}. \tag{4-21}$$

其中 α_1, β_1 和 γ_1 均为超参数。$w_{\mathcal{P},ij}$ 和 $w_{\mathcal{N},ij}$ 分母的第一项体现了自相似性，第二项分别体现了正相对相似性和负相对相似性。

4.5.4 多重关系损失

通过困难样本挖掘和样本加权，多重相似度损失被定义为：

$$\begin{aligned}\mathcal{L}_{X,MS} = \frac{1}{N}\sum_{i=1}^{N}\Big\{\frac{1}{\alpha_1}log\Big[1+\sum_{j\in\mathcal{P}_{X,i}}e^{-\alpha_1(S_{X,ij}-\gamma_1)}\Big] \\ +\frac{1}{\beta_1}log\Big[1+\sum_{j\in\mathcal{N}_{X,i}}e^{\beta_1(S_{X,ij}-\gamma_1)}\Big]\Big\},\end{aligned} \tag{4-22}$$

其中 α_1, β_1 和 γ_1 为超参数。它描述的是 R_A 与 R_C 所定义的度量学习表示，这将被用来训练信号内容表示模块和信号关系表示模块。但这两种关系通常是比较稀疏的，只能比较充分地建模其中一部分信号的表示。

为了更好地表示其他信息，令 $Z = \{L, G, P\}$，并通过 R_Z 导出的标签 Y_Z 定义先验损失。对于一个信号相似关系图，其中没有边的信号占据很大的比例。当没有任何关系信息时，训练出的信号表示会比较不准确。此时可以利用信号的先验信息 R_Z 进行分类聚类，使表征空间更平滑。正如4.5.1所述，我们将分类聚类统一在度量学习的框架下，从而可以得到先验损失：

$$\begin{aligned}\mathcal{L}_{Z,Prior} = \frac{1}{N}\sum_{i=1}^{N}\Big\{\frac{1}{\alpha_2}log\Big[1+\sum_{Y_{Z,i}=Y_{Z,j}}e^{-\alpha_2(S_{Z,ij}-\gamma_2)}\Big] \\ +\frac{1}{\beta_2}log\Big[1+\sum_{Y_{Z,i}\neq Y_{Z,j}}e^{\beta_2(S_{Z,ij}-\gamma_2)}\Big]\Big\},\end{aligned} \tag{4-23}$$

其中 α_2, β_2 和 γ_2 为超参数。先验损失被用来训练解纠缠模块和关系表示模块。其中，关系表示模块中既使用了多重相似度损失，也使用了先验损失，我们将两者的加权和定义为多重关系损失：

$$L_{MR} = L_{M,MS} + \lambda_1 L_{L,Prior} + \lambda_2 L_{G,Prior} + \lambda_3 L_{P,Prior}. \tag{4-24}$$

其中 $\lambda_1, \lambda_2, \lambda_3$ 为平衡多重相似性损失和先验损失的超参数。在其取值合适时，先验损失能够去除信号关系中不可靠的噪声，同时对真实存在的信号相似关系影响较小。多重关系损失将内容和关系表示的训练目标统一在了有监督度量学习的框架下，同时也利用了多源信息，尽可能准确和全面地得到了信号的表示。

4.6 音乐信号纯表示理论

经过以上对信号纯表示框架的介绍，我们能够基于深度学习纯信号表示形式，构建出音乐推荐任务上的优化问题形式：

$$\min \quad \mathcal{L}_{M,MS}(\mathcal{R}(E_C)), Y_M) \tag{4-25}$$

$$\begin{aligned} s.t. \qquad & E_C = C(x) \\ & \mathcal{L}_{A,MS}(M_A \circ E_C, Y_A) \leq \delta_A, \\ & \mathcal{L}_{L,Prior}(M_L \circ E_C, Y_L) \leq \delta_L, \\ & \mathcal{L}_{G,Prior}(M_G \circ E_C, Y_G) \leq \delta_G, \\ & \mathcal{L}_{P,Prior}(M_P \circ E_C, Y_P) \leq \delta_P, \end{aligned} \tag{4-26}$$

其中 C 是内容表示和解纠缠模块，而 $\mathcal{R}$ 则是关系表示模块。目标是通过度量学习融合最关键的共歌单关系，优化音乐信号表示 $E_M = \mathcal{R}(E_C)$。在音乐推荐任务中，我们将内容表示 E_C 视为信号原始表示 h, 并将关系表示模块 $\mathcal{R}$ 视为信号高层表示的变换函数 f_θ。在实际推荐任务中，我们会视情况使用 E_C 或 E_M。

与传统信号纯表示方法对比能够发现，在该任务中，音乐信号表示的约束条件 G 是比较多的，共包括四种约束：共艺人约束、语言先验约束、曲风先验约束和人气先验约束。这些约束均是为了缓解 SPRF 参数量大，容易拟合信号内容和关系中噪声的问题。也因为是纯表示问题，信号不会被映射回音乐信号空间，所以需要这些约束来尽可能地保证信号表示空间的平滑。

4.7 实验和分析

4.7.1 数据集

本章的纯信号表示实验中所使用的多源音乐数据集包括音乐数据、音乐元数据和音乐共歌单关系数据三个部分。音乐和关系数据集均是从 AllMusic 音乐网站收集得到[①]。它是一个关于音乐的元数据数据库，归属于 All Media Guide，也是全球最大的音乐数据库之一。该多元音乐数据集会被应用到 SPRF 的训练

[①]https://www.allmusic.com/

中，结合音乐推荐任务对 SPRF 提取的音乐表示的表达能力和鲁棒性进行测试。同时为了验证音乐推荐下游的艺人推荐任务效果，我们也从 AllMusic 中搜集到了多源艺人数据集，包括艺人元数据和艺人相似关系数据，用来辅助训练音乐和艺人表示。

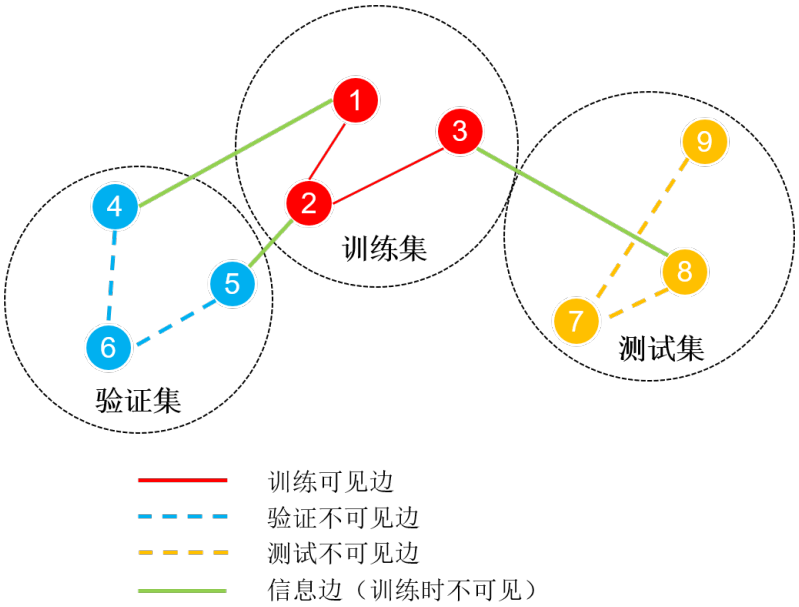

图 4-10: 多源音乐数据集的划分示意图。我们使用红色节点和边进行训练，使用蓝色节点和绿色的边来预测蓝色的边进行验证。在测试时，使用黄色节点和绿色的边来预测黄色的边，并计算 Hitscore@K, Consistent@K, MAP 和 NDCG 指标

多源音乐数据集用于归纳学习的划分如图4-10所示。训练集、验证集和测试集中的音乐信号数据没有交集，但不同集合中的音乐可能存在共歌单或者共艺人相似关系。由于音乐之间的关系是相对静态的，在实际使用中，如果不考虑算法在不同图结构之间的泛化性，直推学习方法能尽可能多地利用音乐关系信息。如果使用这种方法，在模型推断期间，绿色和蓝色虚线将是可见的。不过，为了尽可能地展现我们方法得到的音乐表示的泛化能力，在没有任何特定说明的情况下，下文的实验均是在归纳学习的基础上进行的。

多源音乐数据集中总共包括 9627 位艺人和 61584 首音乐。其中 49266 首音乐用于训练，分别有 6159 首音乐用于验证和测试。音乐共歌单关系数据总共包括 82880 个相似关系，其中 36546 首音乐与其它至少一首音乐有相似关系，即 53.06% 的音乐之间没有任何共歌单相似关系。如图4-11所示，在多源音乐数据中，音乐和艺人关系数目的分布在一定程度上是长尾分布。对于这些长尾的音乐信号，基于音乐内容和解纠缠方法能够对其进行鲁棒建模。而非长尾艺人则

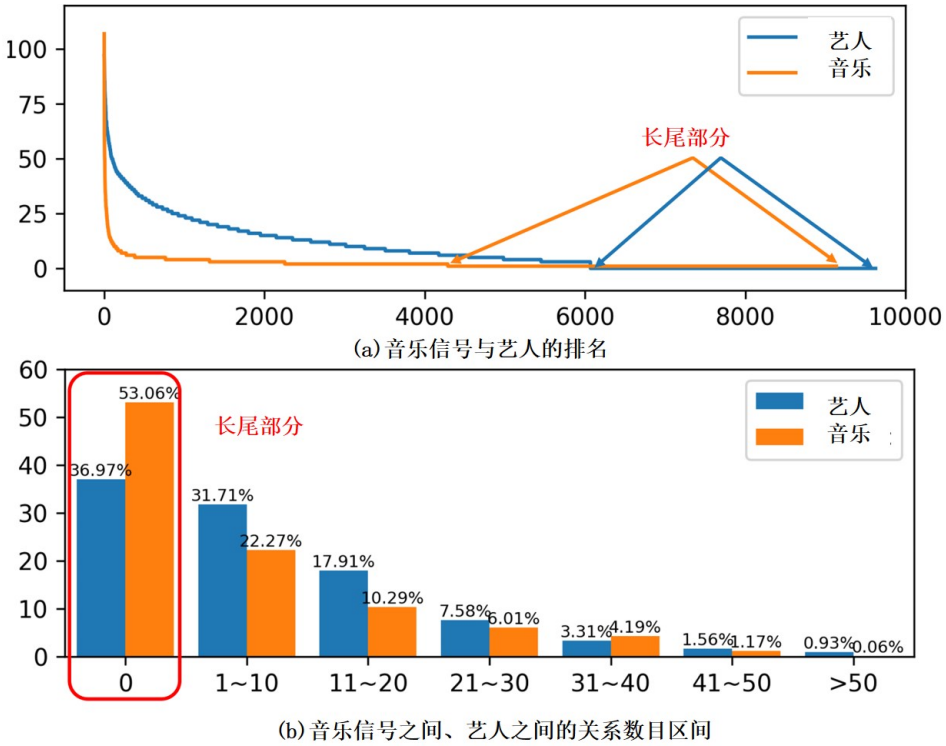

图 4-11: 多源音乐数据集中艺人/音乐的分布。(a) 是每个艺人/音乐的关系数目分布，而 (b) 将关系数数目划分为区间，并计算每个区间中艺人/音乐所占的比例

能够从共歌单关系中得到更精确的表示。这正体现了融合两种范式的多重关系损失及其相应模块的优越性。在下面的实验中我们会分别对这两种情况进行讨论。

4.7.2 实验设置

实验旨在证明信号纯表示框架提取的音乐信号表征 E_M 在音乐推荐任务中有着良好表现，以证明信号理论的可靠性。同样为了证明 SPRF 在融合音乐内容和音乐关系的基础上，其泛化性和鲁棒性有显著提升。我们通过消融实验比较了不同模块提取出的音乐表示在音乐推荐任务中的表现，也比较了我们的方法与其他常见音乐推荐算法之间的推荐效果的差别。此外，为了证明音乐信号表示同样可以对音乐信号相关的其他下游任务产生积极影响，我们还将音乐信号表示和艺人表示结合起来，在相似艺人推荐任务中进行了实验。

用于评估音乐推荐效果的指标包括命中率 (Hitscore@K)，平均精度均值 (Mean Average Precision, MAP)，归一化折损累计增益 (Normalized Discounted Cumulative Gain, NDCG) 和一致性 (Consistent@K)。通过共歌单关系 R_M 和曲风

关系 R_G 导出的标签 Y_M 和 Y_G，这些评价指标被定义为：

$$Hitscore@K = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K_i}\frac{\mathcal{I}(Y_{M,i} = \hat{Y}_{M,j})}{K_i}, \tag{4-27}$$

$$Consistent@K = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}\frac{\mathcal{I}(Y_{G,i} = \hat{Y}_{G,j})}{K}, \tag{4-28}$$

$$MAP = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}\sum_{k=1}^{j}\frac{\mathcal{I}(Y_{M,i} = \hat{Y}_{M,k})}{j}, \tag{4-29}$$

$$NDCG = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}\frac{\mathcal{I}(Y_{M,i} = \hat{Y}_{M,j})}{K_i \cdot log_2(j+1)}, \tag{4-30}$$

其中 $\mathcal{I}(x)$ 是指示函数。前三个指标常用于评估推荐任务的性能，衡量召回效果和排名相似性。由于音乐信号的真实标签集是无序集合，因此 NDCG 的计算与通常情况有所不同。

对于不具有共歌单相似关系的长尾音乐信号，数据集中没有关于它们的相似关系信息。因此我们只能通过间接的指标来客观衡量这些音乐信号的表示和推荐效果。总的来说，同一类型的音乐比不同类型的音乐更相似。所以，Consistent@K 被用来评估这些没有共艺人关系的音乐信号推荐效果的指标。它通过计算当前音乐召回的所有 K 首音乐的曲风是否相似来衡量曲风空间的平滑度。

4.7.3 消融实验

表 4-1: 信号纯表示模型不同子模块得到的音乐信号表示在测试集上的音乐推荐效果对比表

音乐表示方法	Hitscore@10	Consistent@10	MAP	NDCG
随机表示	0.34%	12.95%	0.18%	0.1425
纠缠内容表示	23.23%	33.65%	13.52%	0.3686
音乐内容表示	42.65%	37.82%	27.23%	0.5220
关系表示	25.26%	17.78%	16.21%	0.3825
纠缠关系表示	37.05%	34.96%	24.47%	0.4891
音乐信号表示	**44.58%**	**45.44%**	**29.00%**	**0.5476**

为了验证本章中提出的内容表示模块、解纠缠模块和关系表示模块的作用，我们设计了以下消融实验。表4-1展示了每个模块输出的信号表示在音乐推荐任

务上的性能效果。“随机表示”是基线模型，即每个音乐信号使用随机生成的表示。“纠缠内容表示”和“音乐内容表示”分别由音乐内容表示模块和音乐解纠缠模块生成，均只利用了音乐信号中的内容信息，而没有使用任何的关系信息。“关系表示”指的是用“随机表示”作为关系表示模块的输入，得到的纯音乐关系表征。类似的，“纠缠音乐表示”也是将纠缠的内容表示输入关系模块得到的表征。“音乐信号表示" 指的是整个音乐信号表示框架的提取的最终的音乐表示 E_M。

从表4-1中可以发现，基线模型表示效果很差，即使是纠缠的音乐内容表示，也可以远远超过随机表示。这说明音乐信号内容中存在着大量可挖掘的信息，通过共艺人关系能够从音乐信号中提取出多方面的共性特征。音乐内容表示相对于纠缠内容表示的提升也非常显著，充分说明了音乐信号中的纠缠信息笼统建模并不利于音乐信号表示的准确性。解纠缠模块通过子空间投影和先验损失，使得音乐内容表示更加准确。

将所有的内容表示作为关系表示模块输入，得到的音乐表示对于音乐推荐任务均有提升。其中关系表示和纠缠关系表示的提升最为显著。音乐信号表示相对于音乐内容表示也有一定提升。这是因为随机表示和纠缠内容表示中的音乐之间的关系信息没有被很好地提取出来，导致其表达能力比较弱。而音乐内容表示已经比较充分地融合了音乐相关的信息。关系表示模块能够更进一步，通过共歌单关系建模流行音乐之间的关系信息。音乐内容表示越弱，共歌单关系对音乐信息的补充就越多。从消融实验中可以发现，内容表示模块，解纠缠模块和关系表示模块都是必不可少的。

4.7.4 对比实验

正如4-10小节所述，数据集中存在非长尾音乐与长尾音乐。图4-12展示了不同音乐表示方法在非长尾与长尾音乐测试数据上的效果。从图中可以发现，除了随机生成内容表示的随机表示与关系表示以外，其它情况下长尾音乐的推荐效果都会略逊于非长尾音乐。这是因为非长尾音乐能够比较充分地利用共歌单关系，这本身是有利于音乐推荐的。另一方面，能够发现所有音乐信号推荐效果在整个信号纯表示框架下表现最好。在非长尾音乐上，音乐信号表示的 Consistent@10 比随机表示提升了 32.49%；在长尾音乐上，音乐信号表示的 Consis-

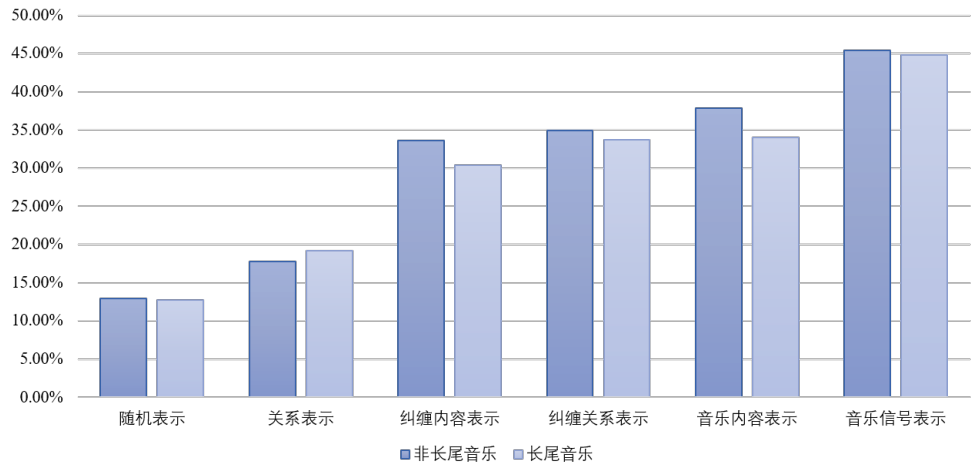

图 4-12: 测试集中长尾和非长尾音乐上的表示和推荐效果 (通过 Consistent@10 衡量)

tent@10 比随机表示提升了 32.13%。音乐信号表示与纠缠内容表示相比，在长尾与非长尾音乐上的 Consistent@10 差距不大，说明它能够对长尾音乐进行同样鲁棒的建模。

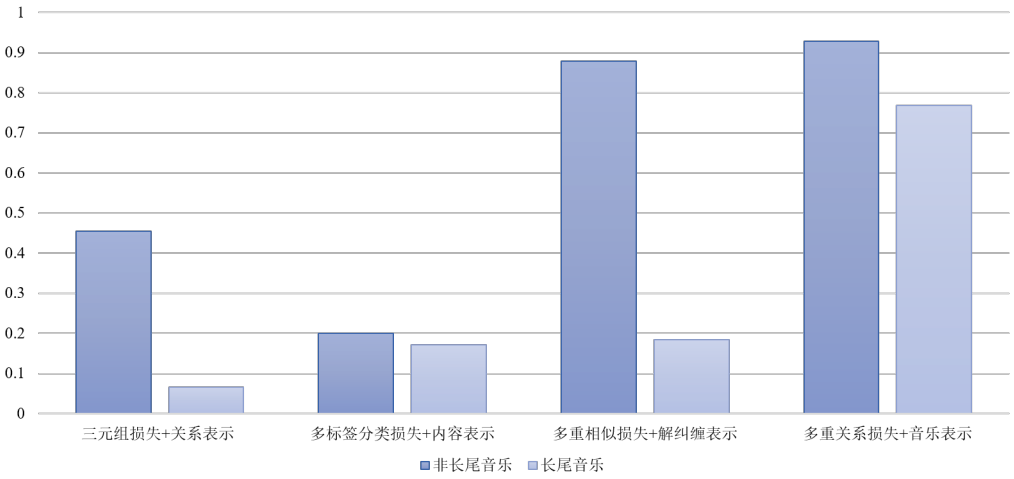

图 4-13: 不同音乐信号表示算法在音乐推荐任务上的 Hitscore@10 对比

我们还通过 Hitscore@10 对比了基于多重关系损失的 SPRF 与其它经典的音乐推荐算法的推荐效果。实验结果如图4-13所示，基于三元组损失的关系表示模型[39], 基于多标签分类损失的内容表示模型[48] 和基于多重相似性损失的解纠缠表示模型[46,50] 均为音乐表示范式中具有代表性的算法。基于三元组损失的关系表示模型仅仅使用了共歌单关系，在音乐相似性图上通过 GraphSAGE 传播节点表示。音乐信号的表征是随机初始化并通过共歌单关系训练得到，所有对非长尾音乐的建模效果较好，而长尾音乐则接近于随机表示。基于多标签分类损失的内容表示模型能够基于音乐内容和元数据得到音乐表示，但受限于音乐信

号的多义性，表示目标不明确，推荐效果不好。基于多重相似性损失的解纠缠表示模型与 SPRF 比较相近，在非长尾艺人上的推荐效果也接近于 SPRF。但因为没有将先验信息统一到度量学习框架下，缺失了音乐曲风、人气和语言等先验约束，在长尾音乐上表现仅与基于多标签分类的内容表示模型相近。

4.7.5 下游相似艺人实验

表 4-2: 音乐信号表示生成的下游艺人表示在测试集上的艺人推荐效果对比表

艺人表示方法	Hitscore@10	Consistent@10	MAP	NDCG
随机表示	1.47%	11.23%	1.08%	0.1803
协同内容表示	4.30%	12.88%	3.83%	0.2144
纠缠内容表示	30.29%	54.88%	17.87%	0.4080
艺人内容表示	39.89%	65.03%	25.68%	0.4846
关系表示	21.90%	28.31%	14.77%	0.3553
协同关系表示	26.91%	42.67%	18.71%	0.4045
纠缠关系表示	34.76%	57.04%	23.60%	0.4618
艺人表示	**39.98%**	**86.19%**	**26.14%**	**0.4904**

为了证明音乐信号表示具有一定的泛化性和普适性，我们不仅在音乐推荐任务上验证了它的效果，也会在下游的艺人推荐任务上验证它的效果。与表4-2类似，表4-2中的艺人表示是由每个艺人对应的音乐表示集合平均得到的。不同的是在艺人推荐任务中增加了“协同内容表示”和“协同关系表示”。“协同内容表示”是通过用户-艺人的协同过滤的方式直接提取艺人表示[107]。“协同关系表示”相比于“协同内容表示”还结合了艺人与艺人之间的关系信息。

从表中能够发现，基于 SPFR 提取的音乐信号表示平均得到的艺人表示同样在所有评价指标上优于其它方法得到的艺人表示。从 Consistent@10 来看，先验知识的加入显著提高了长尾艺人的表达能力。具有相同标签的艺人可以在艺人表征空间中自然地聚集在一起，这些艺人之间相似的可能性将大大提高。而且，在提高长尾艺人表达能力的基础上，正则化项的加入对有关系的人气艺人没有负面影响，这同样可以从第三方抽样的主观评价结果中得到证明。所以，SPRF 获得的音乐表示也同样对下游的艺人推荐任务有帮助。实际上，它也可以应用

于所有与音乐信号有关的下游任务上，这将是我们未来的工作。

图4-14展示了三个艺人推荐的结果样例。从推荐结果中可以发现，推荐的相似艺人无论在曲风、地区还是咖位上都是比较接近的，并且也与我们的认知接近。另外也能发现，尽管并没有显式地约束推荐算法推荐相似的乐队，但 SPRF 得到的下游艺人表示依然能够办到这一点。所以，音乐信号表示框架提取的音乐表示对于下游其它任务也是有帮助的。

图 4-14: 艺人推荐效果展示

4.8 本章小结

本章主要基于信号纯表示理论，提出了一种基于内容和关系的信号纯表示框架。此框架通过内容表示模块提取信号的关键信息，通过解纠缠模块结合信号元数据在信号子空间中优化子空间表示，通过关系表示模块融合信号的关系模式。三个子模块将信号纯表示的两种范式进行了利用，从多个角度对信号表示进行了系统建模。同时，我们将分类聚类和度量学习方法整合到同一框架下，并基于多重关系损失函数对信号表示约束建模，利用多种相似性关系平滑了信号表示空间，缓解了长尾信号表示不准确的问题，提高的信号表示算法的泛化性和鲁棒性。经过实验验证，SPRF 在音乐信号推荐任务和下游的艺人推荐任务上均有着良好的表现，性能好于单纯基于内容或关系的音乐推荐算法。

第五章　脉冲信号解码表示研究

本章分为七个部分，第一个部分简要介绍解决脉冲信号解码表示问题，系统阐述脉冲信号调制模式和表示方法，并介绍深度脉冲信号掩膜（Deep Pulse Signal Mask, DPSM）算法的整体流程；第二个部分介绍 DPSM 算法核心的递归表示网络结构，包括递归掩膜模块对脉冲分量的表示；第三个部分提出脉冲信号解码表示中存在的通道排列问题，并通过 DPSM 算法的损失函数进行置换不变性训练；第四个部分阐述脉冲信号分量表示中存在的歧义性问题，并提出适当的预处理和后处理算法利用脉冲描述字信息提高算法的鲁棒性；第五个部分结合 DPSM 算法说明脉冲信号解码理论在雷达脉冲信号分选问题上的具体形式；第六个部分在多种复杂情况的仿真数据集下验证 DPSM 算法在雷达信号分选任务上的效果，并在多种雷达调制模式上对算法进行了验证；最后一个部分对本章进行小结。

5.1　脉冲信号的解码表示

在本章中，我们主要关注脉冲信号的解码表示和分选问题，并提出了深度脉冲信号掩膜（Deep Pulse Signal Mask, DPSM）算法。DPSM 算法是第一种结合深度可分离空洞卷积和双路径注意力机制的深度学习方法，它构建编码器和解码器，递归地得到特征空间中每个脉冲点分量的掩膜和表示。DPSM 可以提高脉冲信号表示和分选的精度，更好地解决脉冲信号解码中的通道排列和分选问题。该算法不需要关于脉冲信号的先验知识，这更有利于实际的任务。但当关于脉冲信号的其它先验知识可用时，我们也研究了如何将其纳入 DPSM 算法框架中：基于预处理微调和后处理重新聚类方法来充分利用这些额外的信息。它可以自动适应复杂的环境，并且无需手动设置阈值。以雷达脉冲信号分选为例，我们考虑了三种可能的方法，以进一步提高 DPSM 算法在多种脉冲重复间隔（Pulse Repetition Interval, PRI）调制场景中的效果。在已知 PRI 或具有已知调制模式类型但未知 PRI 的情况下，DPSM 算法可以在参差 PRI、滑动 PRI、成组 PRI 调制等复杂调制条件下，以高精度对雷达脉冲信号进行分选。本章的脉冲信号解码

表示研究主要关注雷达脉冲信号的分选，其它的脉冲信号解码表示任务仍然可以使用类似方法建模。

5.1.1 脉冲信号表示

本章主要研究如何对脉冲信号进行解码表示和分选。脉冲信号通常会用序列来表示，是一组按照时间排序的信号特征。对于雷达脉冲信号而言，脉冲描述字就是这样一组特征序列。离散的脉冲描述字序列 $\boldsymbol{D}$ 可以表示为 $\boldsymbol{D} = [\boldsymbol{D_1 D_2} \ldots \boldsymbol{D_T}]$。其中，$D_i$ 是一个多维向量，包含所有的脉冲描述字参数。为了表示方便，在本章中我们用脉冲描述字来表示所有种类脉冲信号的特征序列。而到达时间是从脉冲描述字序列中选择出来的，所有脉冲信号均具有的普适特征，它表示脉冲信号出现的每个时间点。我们从脉冲描述字序列 $\boldsymbol{D}$ 中选择出到达时间序列 $\boldsymbol{S}$，$\boldsymbol{S}$ 是一个稠密的时间序列向量，每个元素 S_i 表示脉冲序列的到达时间点。

在雷达脉冲信号分选任务中，$\boldsymbol{S}$ 是由多个脉冲信号所构成的到达时间序列。依靠一组脉冲信号来预测多组脉冲信号的值是一个欠定问题。幸运的是，可以通过将原始到达时间序列 $\boldsymbol{S}$ 转换为稀疏的脉冲序列 $\boldsymbol{x}$，利用 $\boldsymbol{x}$ 的稀疏性实现脉冲信号的表示和分选。将 $\boldsymbol{S}$ 视为一个按 ToA 升序排列的有序集合，即 $\boldsymbol{S} = \{ToA_1, ToA_2, \ldots, ToA_n\}$。对于每个 $ToA_i \in \boldsymbol{S}$，不妨定义 $ToA_i \triangleq ToA_i - ToA_1 + 1$。其中，$\triangleq$ 表示赋值符号。我们希望将 $\boldsymbol{S}$ 的值缩放到 $[1, T]$ 范围内。缩放后，$ToA_1 = 1$，而 $ToA_n = T$。对于雷达脉冲信号而言，到达时间序列主要由 PRI 决定。如果到达时间序列已知，则 PRI 可通过以下方式估算：

$$PRI(i) = ToA_{i+1} - ToA_i. \tag{5-1}$$

我们可以将集合 $\boldsymbol{S}$ 转换为时域中的脉冲信号 $\boldsymbol{x} = [x_1, x_2, \ldots, x_n, \ldots, x_T]$，如下所示：

$$\boldsymbol{x}(n) = \sum_{toa \in \boldsymbol{S}} \delta(n - toa). \tag{5-2}$$

其中 $\delta(n - toa)$ 表示 toa 处的移位单位脉冲函数，n 是脉冲序列 $\boldsymbol{x}$ 的时间索引。

经过上述转换之后，$\boldsymbol{x}$ 是一个脉冲信号。对于一列多分量混合的脉冲信号，

假设 $\boldsymbol{x}$ 是通过混合 K 个独立的脉冲信号混合获得的，我们将 y 定义为脉冲信号分量编号 $[1,K]$，并定义映射 $f:\boldsymbol{x}\rightarrow y$。其中 K 通常是事先未知的。将 $\boldsymbol{x_k}$ 定义为第 k 个分量脉冲信号，并将 $\hat{\boldsymbol{x}}_k$ 定义为通过脉冲信号解码表示算法分选得到的的脉冲信号序列，那么：

$$\boldsymbol{x_k}(n)=\sum_{toa_k\in \boldsymbol{S}\wedge f(\boldsymbol{x}(n))=k}\delta(n-toa_k). \tag{5-3}$$

5.1.2 雷达脉冲信号 PRI 调制

在脉冲多普勒雷达中，为了消除距离模糊、速度模糊或消除目标覆盖，通常会对雷达的脉冲重复周期进行调制。常见的 PRI 调制方法包括抖动 PRI、参差 PRI、滑动 PRI 和成组 PRI。将脉冲信号的 PRI 序列的期望表示为 $\mu_{PRI}=\frac{1}{T-1}\sum_{i=1}^{T-1}PRI(i)$。连同固定 PRI 在内，五种 PRI 调制类型表示如下：

$$PRI_{fixed}(i)=\mu_{pri}+\delta_F(i), \tag{5-4}$$

$$PRI_{jitter}(i)=\mu_{pri}+\delta_J(i), \tag{5-5}$$

$$PRI_{stagger}(i)=\mu_{pri}+A(i), \tag{5-6}$$

$$PRI_{sliding}(i)=\mu_{pri}+B(i), \tag{5-7}$$

$$PRI_{group}(i)=\mu_{pri}+C(i). \tag{5-8}$$

δ_F 和 δ_J 都是高斯分布噪声，但 δ_F 的标准差通常在 μ_{pri} 的 5% 以内。δ_J 的标准差能够达到 μ_{pri} 的 30%。设 β_A，β_B 和 β_C 为给定的限制因子。那么对于交错 PRI，PRI 参数集合满足以下条件：

$$A=\left\{A_\sigma\middle||A_\sigma|\le\beta_A,0\le\sigma<K_A,\sum_{\sigma=0}^{K_A-1}A_\sigma=0\right\}. \tag{5-9}$$

设 $mod(i,K_A)=\sigma$，K_A 表示参差参数集合的大小。那么 $A(i)=A_\sigma$。对于线性滑动 PRI，我们则可以定义：

$$B(i)=\beta_B(\frac{\sigma}{K_B}-\frac{1}{2})\mu_{pri},\quad mod(i,K_B)=\sigma. \tag{5-10}$$

类似地，如果存在 K_C 个 PRI 参数组，则：

$$C = \left\{ C_\sigma \middle| |C_\sigma| \le \beta_C, 0 \le \sigma < K_C, \sum_{\sigma=0}^{K_C-1} C_\sigma = 0 \right\}. \tag{5-11}$$

每组内部有 G 个 PRI。当 $\lfloor mod(i, G \cdot K_C)/G \rfloor = \sigma$ 时，$C(i) = C_\sigma$。其中 $mod(x/y)$ 和 $\lfloor x \rfloor$ 分别代表 x/y 的模和对 x 向下取整。

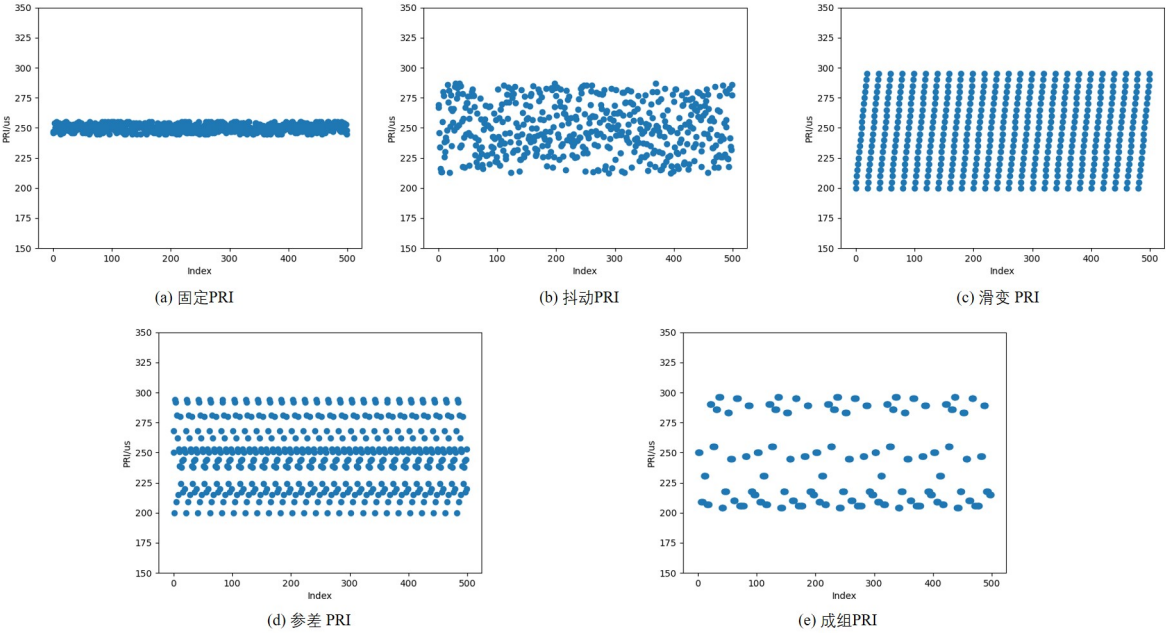

图 5-1: PRI 调制类型示意图 ($\mu_{pri} = 250\mu s$)

图5-1是上述五种 PRI 调制类型的示意图。根据雷达系统和功能，雷达脉冲信号 PRI 调制类型也不同。固定 PRI 的典型应用是常规搜索和跟踪雷达、脉冲多普勒雷达。抖动 PRI 用于抵消预测的 ToA 干扰并减少特定干扰效应。参差 PRI 用于消除 MTI 系统中的盲速。滑动 PRI 在俯仰扫描中提供恒定的高度覆盖，避免阴影效应。成组 PRI 用于解决速度或距离模糊，尤其是脉冲多普勒雷达[108]。

在 PRI 未知、雷达辐射源数量未知、具有环境噪声和调制类型未知的实际情况下，解决雷达脉冲信号解码表示分选问题极其困难。我们将提出三种可能的方法（使用三种不同的额外信息）来解决它。只要事先知道每个发射器的调制类型，那么 DPSM 算法就可以扩展到多模调制的脉冲信号解码表示问题下，而不需要考虑脉冲的具体调制参数。

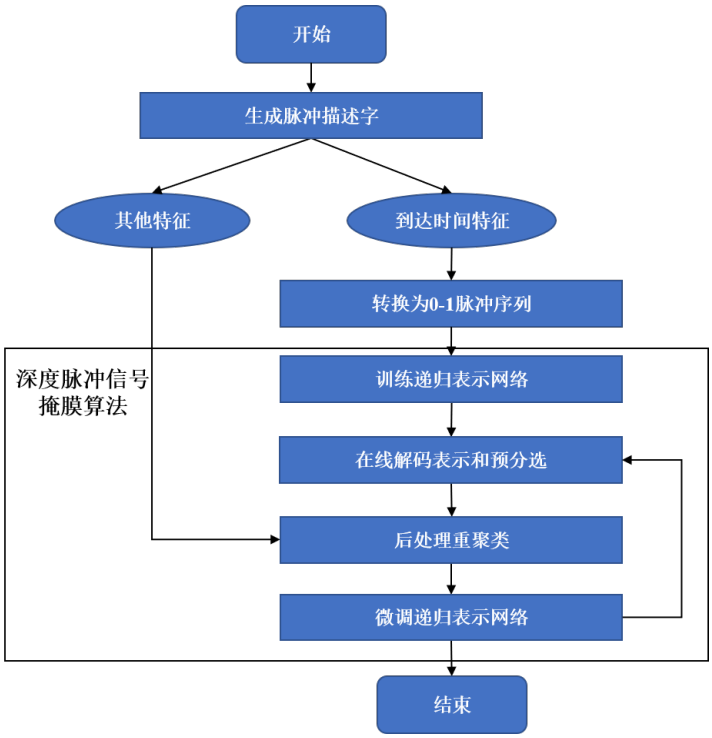

图 5-2: DPSM 处理流程图

5.1.3 DPSM 算法流程

在固定 PRI 的情况下，虽然精度不高，但可以使用直方图方法来进行脉冲信号的解码表示，使用序列搜索的方法进行进一步的雷达脉冲信号分选。当已知脉冲信号分量的数目时，基于聚类的算法还可以对未知 PRI 的脉冲信号进行解码表示。然而，对于未知 PRI、噪声抖动 PRI 脉冲信号和未知脉冲信号分量的脉冲信号，即使现有的深度学习方法也无法解决这一问题，在多个未知 PRI 调制模式的情况下对多分量脉冲信号进行解码表示则是更加困难的问题。但本文提出的 DPSM 算法可以在一定条件下解决这一问题。

本章提出的深度脉冲信号掩膜算法主要基于盲源信号分离理论[109-110]，灵感来自独立分量分析[111] 和非负矩阵分解（NMF)[97] 等方法。递归表示网络（Recursive Representation Network, RRN）是其核心。在电子战中，交错的雷达脉冲信号通常是一列连续到达接收机的流数据，它们不能立即被处理。整个脉冲信号解码表示的处理框架如图5-2所示。在将接收到的 PDW 数据 $\boldsymbol{D}$ 的到达时间序列 $\boldsymbol{S}$ 形式化为 $\boldsymbol{x}(n)$ 之后，除了 ToA 之外的所有可用 PDW 参数都保存为可选的后处理算法额外特征向量序列 $\boldsymbol{E}$，即 $\boldsymbol{D} = [\boldsymbol{E}; \boldsymbol{S}]$。在雷达脉冲信号分选任务中，可以通过模拟数据或通过收集离线数据来获得脉冲信号序列 $\boldsymbol{x}$ 和独立的脉冲信

号序列 $\boldsymbol{x}_k$。我们将长脉冲信号序列 $\boldsymbol{x}$ 分成 M 帧。每帧脉冲信号包含 N 个脉冲采样点数据，$\boldsymbol{s} = [x_i, x_{i+1}, \ldots, x_{i+N-1}]$，$i = 1, 2, \ldots, M$。$\boldsymbol{s}_k$ 表示脉冲信号分量 k 的脉冲信号帧，$\boldsymbol{y}_s$ 是 $\boldsymbol{s}$ 每个脉冲点的分量标签，通过它可以很容易地推断出 $\boldsymbol{s}_k$ 的值。这些帧形成了一个经过处理的脉冲数据集 $\{\boldsymbol{s}^{(j)}, \boldsymbol{y}_s^{(j)}\}$，$j = 1, 2, \ldots, J$，包括总共 J 对数据。

之后将每帧脉冲信号 $\boldsymbol{s}$ 输入 RRN 模型，并通过 DPSM 的损失函数离线训练得到脉冲信号的表示，从而解码预测分量的标签。在训练阶段完成后，我们执行在线脉冲信号分选以评估 DPSM 算法。我们可以在线进行预处理，以自适应地调整 RRN 参数。同时，在雷达脉冲信号解码表示和分选任务中，RRN 能够预测脉冲信号分量的数量，并基于信号表示掩膜对雷达脉冲信号帧进行分选。脉冲信号表示掩膜是在基于 ToA 的脉冲信号特征空间中，对应每个信号分量的系数矩阵。它表示基于特征空间的每个单独脉冲信号分量特征与脉冲信号特征的权重或比例。之后，我们使用带噪声的参数 $\boldsymbol{E}$ 作为后处理的输入并执行重新聚类，以获得更稳健的结果。其结果能够作为在线微调的标签集。RRN 能够在后台进行动态的微调，以避免影响在线分选的实时性能。微调完成后，我们更新 RRN 参数。之后对后续脉冲信号帧重复上述过程，直到处理完所有帧。最后，标记脉冲信号中每个脉冲点属于哪个信号分量，从而完成分选。DPSM 算法包括 RRN 训练阶段和整个在线脉冲信号解码表示和分选的过程。

5.2 递归表示网络结构

5.2.1 模型结构和特点

表 5-1: RRN 子模块的输入形状

子模块	形状	仿真实验中的形状
编码器	(B, T)	$(4, 60000)$
递归掩膜模块	(B, F, L)	$(4, 256, 3749)$
解码器	(B, K, F, L)	$(4, 6, 256, 3749)$

DPSM 算法的关键部分是 RRN 模型，如图5-3所示。它是一个将空洞卷积块与帧内-帧间注意力机制相结合的网络。这种结构有利于整合特征信息并提取脉

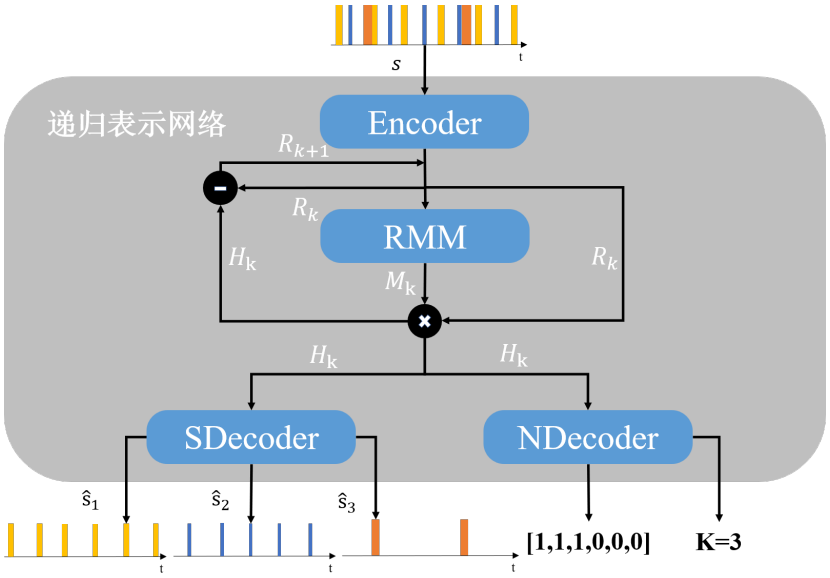

图 5-3: DPSM 的递归表示网络的结构示意图

冲信号的模式。表5-1的第二列显示了 RRN 的每个模块的输入数据的形状，第三列显示在实际仿真实验时 RRN 各个模块的输入形状。第一个维度 B 表示训练数据的批量大小。解码器的输出与编码器的输入形状相同。RRN 的思想与有监督非负矩阵分解（非负矩阵分解参考第二章的相关介绍）类似，与有监督非负矩阵分解相比有四个比较显著的特点。

首先，RRN 包括可学习的自适应卷积编码器。它将脉冲信号帧 $\boldsymbol{s}$ 直接从时域映射为系数矩阵 $\boldsymbol{H}$，隐式地将雷达脉冲序列编码到适当的特征空间从而得到其表示，而不是通过短时傅里叶变换将其转换到时频域。解码器卷积核类似于非负矩阵分解的基矩阵 $\boldsymbol{W}_k$。解码器用于将脉冲信号表示解码为脉冲分量标签序列 $\boldsymbol{y}_s$，而不是短时傅里叶变换频谱 $\boldsymbol{X}_k$。

其次，RRN 通过空洞卷积，利用脉冲上下文信息计算每个脉冲信号分量的表示掩膜。空洞卷积是指在其卷积核中具有额外空洞的卷积运算。与通道的卷积运算相比，空洞卷积具有一个特别的超参数，称为空洞率，这是指卷积核中两个值之间的步长（通常卷积运算的空洞率为 1）[112]。而 PRI 的短期模式和长期模式信息会通过双路径注意力机制融合到脉冲信号分量的掩膜表示中。换言之，RRN 并不是简单且直接地估计每个脉冲信号分量的表示 $\boldsymbol{H}_k$。

第三，RRN 计算脉冲信号分量掩膜系数矩阵的过程是递归的。在每次迭代过程中，通过递归掩膜模块（Recursive Mask Module, RMM）从脉冲信号表示的残差系数矩阵 $\boldsymbol{R}_k$ 中提取信号分量 k 的脉冲信号表示掩膜 $\boldsymbol{M}_k$。由掩膜 $\boldsymbol{M}_k$ 过

滤后的残差系数矩阵 $\boldsymbol{R}_{k+1}$ 被用于下一次迭代以估计下一个脉冲信号表示掩膜 $\boldsymbol{M}_{k+1}$。也就是说，在每次迭代中，分量脉冲信号的系数矩阵 $\boldsymbol{H}_k$ 都会被从原始脉冲信号表示 $\boldsymbol{H}$ 中删除以简化问题。

第四，在雷达脉冲信号分选任务中，RRN 还能够基于脉冲信号表示自动地预测脉冲分量的数目，并较好地解决分选任务中的通道置换问题。

5.2.2 编码器和解码器

RRN 的编码器包括卷积核 $\boldsymbol{B}$ 和 ReLU 激活函数。它确保了编码器的系数矩阵 $\boldsymbol{H}$ 为非负的。即使在短时间内，脉冲信号也是非平稳的信号。因此很难用线性函数对脉冲信号序列建模。非线性激活使特征的表示能力更强。编码器表示如下：

$$\boldsymbol{H} = Encoder(\boldsymbol{B}, \boldsymbol{s}) = ReLU(Conv(\boldsymbol{B}, \boldsymbol{s})), \boldsymbol{H} \in \mathcal{R}^{F \times L}. \tag{5-12}$$

其中，F 是 $\boldsymbol{H}$ 的特征维度，L 是时间维度。如果解码系数矩阵 $\boldsymbol{H}$，则可以获得重构的脉冲信号帧 $\hat{\boldsymbol{s}}$。而若对通过 RRN 处理的 $\boldsymbol{H_k}$ 进行解码，我们将获得分量脉冲信号帧 $\hat{\boldsymbol{s}}_{\boldsymbol{k}}$:

$$P^{(s)} = Sigmoid(DeConv(\boldsymbol{W}, \boldsymbol{H})), \tag{5-13}$$

$$\hat{\boldsymbol{s}} = SDecoder(\boldsymbol{W}, \boldsymbol{H}) = I(P^{(s)} > t_s) \tag{5-14}$$

$$h_k = DeConv(\boldsymbol{W}, \boldsymbol{H_k}) \tag{5-15}$$

$$P_k^{(s)} = Sigmoid(h_k), \tag{5-16}$$

$$\hat{\boldsymbol{s}}_{\boldsymbol{k}} = SDecoder(\boldsymbol{W}, \boldsymbol{H_k}) = I(P_k^{(s)} > t_s). \tag{5-17}$$

其中，$Sigmoid(x) = \frac{1}{1+e^{-x}}$ 是激活函数，$DeConv$ 表示反卷积操作。h_k 可以看作第 k 个脉冲信号分量的解码表示。

SDecoder 是脉冲信号解码器。它可以通过全连接层或反卷积层实现。当然，反卷积层的实现更为有效。脉冲点始终都是 0-1 的二进制变量，所以可以通过 SDecoder 最后的 Sigmoid 函数来估计脉冲点出现的置信度 $P_k^{(s)}$。$I(x)$ 是指示函数。当脉冲信号帧属于第 k 雷达的置信度大于阈值 t_s 时，指示函数 $I(P_k^{(s)} > t_s)1$；否则，$I(P_k^{(s)} > t_s) = 0$。

这里不适合选择 Softmax 进行概率归一化，因为脉冲信号帧可能会出现重叠。此时，某一脉冲信号帧可能属于多个脉冲分量。不同的脉冲信号分量之间具有弱相关性，该特征是判断不同模式的脉冲信号序列是否属于同一分量的基础。由于现实环境中存在噪声，重建的 $\hat{\boldsymbol{s}}$ 和真实的 $\boldsymbol{s}$ 之间通常存在一定的差距。值得注意的是，我们在 DPSM 算法中假设基矩阵 $\boldsymbol{W}_k = \boldsymbol{W}$。由于神经网络有足够的多的参数，这种简化可以降低其复杂度。

目前的脉冲解码表示算法都不能很好地确定噪声环境中脉冲信号分量的数量。RRN 能够基于分量系数矩阵和残差系数矩阵来比较精确地预测脉冲点的存在概率：

$$P_k^{(n)} = NDecoder(\boldsymbol{H}_k) = Sigmoid(Linear(\boldsymbol{H}_k)) \tag{5-18}$$

$$K = \sum_{k=1}^{K_{max}} I(P_k^{(n)} > t_n). \tag{5-19}$$

$Linear(\cdot)$ 表示线性变换。与 SDecoder 类似，NDecoder 是脉冲分量数目解码器，用于预测脉冲信号分量的数目。t_n 是信号分量存在的置信度阈值。经过 NDecoder 分支的解码后，RRN 能够预测脉冲信号分量是否存在，从而很容易地计算得到分量数目 K。

5.2.3 递归掩膜模块

递归掩膜模块的主要组成部分是两个深度可分离的空洞卷积块和双路径注意力块。如图5-4所示，它在融合局部和全局信息后对 $\boldsymbol{H}$ 进行分解，从而得到脉冲信号的分量表示。局部信息是指脉冲点之间的局部关系，反映脉冲信号在短时间内的变化（实际中其变化通常在 2000 μs 内）。全局信息则是指一段脉冲信号帧甚至整个脉冲信号所呈现出的模式。该模式更为宏观（实际中其变化通常在 60000 μs 左右）。之所以使用双路径注意力机制，是为了更好地建模多种脉冲信号调制模式。以成组 PRI 调制为例，组内 PRI 的变化是局部信息，组间 PRI 的变化则属于全局信息。通过该模块，RRN 从整体的脉冲信号表示系数矩阵递归地预测分量表示掩码 $\boldsymbol{M}_k$，从而进一步得到分量脉冲信号表示 $\boldsymbol{H}_k$。图中的 BottleNeck 是指神经元前、后两层的维度高于中间层的神经网络结构。这种结构看起来像一个瓶颈。不过，这里的瓶颈仅包括前两层，主要起到降维的作用。

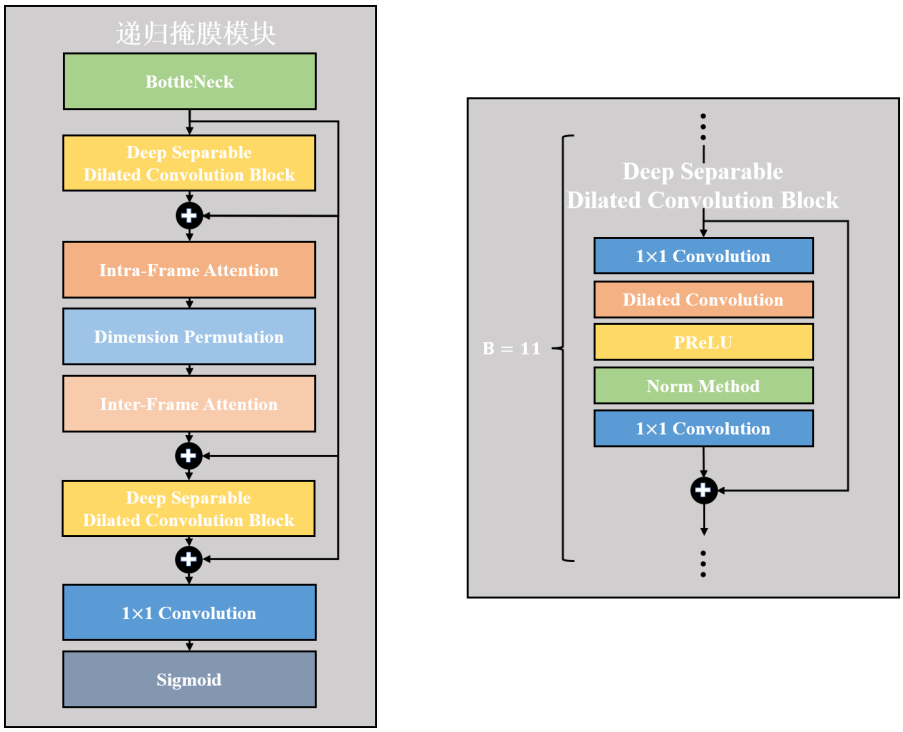

图 5-4: RRN 的递归掩膜模块的结构示意图（左）; RRN 的深度可分离卷积块的细节（右）

在瓶颈层之后，输入脉冲信号的维度 F 被降低到 F'。这里的激活函数 PReLU 定义为:

$$PReLU(y_i) = \begin{cases} y_i, & y_i > 0 \\ \gamma_i y_i, & y_i \leq 0. \end{cases} \tag{5-20}$$

其中 γ_i 是可学习的参数，是激活函数的负斜率。y_i 是第 i 个通道中非线性激活函数的输入。

假设 $\boldsymbol{R_k}$ 是第 k 次迭代前的残差系数矩阵表示，在第一次迭代前，$\boldsymbol{R_1 = H}$，则递归表示过程可以被描述如下:

$$\boldsymbol{M_k} = RMM(\boldsymbol{R_k}) \tag{5-21}$$

$$\boldsymbol{H_k} = \boldsymbol{M_k} \odot \boldsymbol{R_k} \tag{5-22}$$

$$\boldsymbol{R_{k+1}} = \boldsymbol{R_k} - \boldsymbol{H_k}, \tag{5-23}$$

其中 $k = 1, 2, \ldots, K_{max}$，$K_{max}$ 是脉冲信号最大可能分量数目。

在 RMM 中，我们使用了空洞率为 2^b 的空洞卷积层，其中 $b = 1, 2, \ldots, B$。脉冲信号变化迅速，脉冲调制模式的识别也同样需要长期的信息。因此，在考

虑时间分辨率的同时，也必须尽可能地扩大感受野。感受野被定义为每个层的输出特征图上的值被映射到输入特征图后，输入特征图上的区域大小。空洞卷积层减少了具有深度可分离卷积结构的参数量。深度可分离卷积则将普通卷积分解为两个连续的操作，即逐点卷积和通道卷积，提高了网络的推理计算效率。被编码器下采样 16 次后的脉冲信号特征将被输入到包含 $B = 11$ 个深度可分离空洞卷积块的序贯结构中。其中每个块的感受野为 8191。因此，它已经足够提取脉冲信号的 PRI 特性了。

我们将第一个深度可分离空洞卷积块处理后的特征划分为 S 帧，表示为 $\boldsymbol{Z} \in \mathcal{R}^{F' \times S \times L'}$。此时，脉冲信号的局部模式已经被编码在每一帧特征中，而全局模式则体现在帧与帧之间。注意力操作本质上则是在为特征矩阵 $\boldsymbol{Z}$ 中每个时间步的特征向量分配权重。在 RMM 中，所有的脉冲信号表示系数矩阵都是连续值，此时的注意力操作可以被理解为软寻址。如果特征序列的每个向量都以 $\boldsymbol{K}, \boldsymbol{V}$ 的形式存储，则注意力操作是通过计算查询向量 $\boldsymbol{Q}$ 和键向量 $\boldsymbol{K}$ 之间的相似度来完成寻址的。$\boldsymbol{Q}$ 和 $\boldsymbol{K}$ 计算的相似性反映了提取的值向量 $\boldsymbol{V}$ 的重要性。我们首先通过帧内注意力对帧内的脉冲信号点的相关性进行建模，然后再通过帧间注意力对帧间脉冲信号点的相关性进行建模。双路径注意机制有利于脉冲调制方式的识别和表示。同样以成组 PRI 调制为例，帧内注意能够提取每个组内的恒定 PRI 信息，帧间注意则负责分析组间 PRI 的变化。帧内注意机制表示如下：

$$\boldsymbol{K_1}, \boldsymbol{Q_1}, \boldsymbol{V_1} = Linear(\boldsymbol{Z}) \tag{5-24}$$

$$IntraAttention(\boldsymbol{K_1}, \boldsymbol{Q_1}, \boldsymbol{V_1}) = Softmax\left(\frac{\boldsymbol{Q_1}\boldsymbol{K_1^T}}{\sqrt{F'S}}\right)\boldsymbol{V_1}. \tag{5-25}$$

类似地，在维度变换后，对 $\boldsymbol{Z^T} \in \mathcal{R}^{F' \times L' \times S}$，帧间注意力机制表示如下：

$$\boldsymbol{K_2}, \boldsymbol{Q_2}, \boldsymbol{V_2} = Linear(\boldsymbol{Z^T}) \tag{5-26}$$

$$InterAttention(\boldsymbol{K_2}, \boldsymbol{Q_2}, \boldsymbol{V_2}) = Softmax\left(\frac{\boldsymbol{Q_2}\boldsymbol{K_2^T}}{\sqrt{F'L'}}\right)\boldsymbol{V_2}. \tag{5-27}$$

在维度变换前，通过帧内注意力获得局部信息。这相当于将相邻的 ToA 脉冲点特征堆叠在一起，然后作为特征向量输入帧内注意力层。维度变换后，再将一定

间隔的脉冲点特征堆叠并作为帧间注意力层的输入，以获得全局信息。

归一化方法和残差连接是 RMM 取得良好结果的重要保证。层归一化方法同时在通道和时间两个维度上中对脉冲信号系数矩阵进行归一化。在 RMM 中，层与层之间以及块与块之间添加了许多残余连接，以防止出现梯度消失问题。

RMM 根据脉冲信号表示计算分量表示掩膜 $\boldsymbol{M}_k$，然后将第 k 个脉冲信号分量系数矩阵与残差系数矩阵 $\boldsymbol{R}_k$ 分离。在这里，因为 $\boldsymbol{W} = \boldsymbol{W}_k$，所以无论 $\boldsymbol{W}$ 是否为非负矩阵，递归表示的过程均等价于有监督非负矩阵分解的迭代过程：

$$\boldsymbol{M}_k \prod_{i=1}^{k-1}(1 - \boldsymbol{M}_i) = \frac{\boldsymbol{W}_k \boldsymbol{H}_k}{\boldsymbol{W}\boldsymbol{H}} = \frac{\boldsymbol{H}_k}{\boldsymbol{H}}. \tag{5-28}$$

实际上，$\boldsymbol{W}_k$ 可以被显示约束为非负矩阵，但我们发现不对其进行特别约束，脉冲信号解码表示的效果更好。$\boldsymbol{W}_k$ 是否为非负并不影响非负矩阵分解算法所导出的 RMM，而只会影响编码器-解码器将脉冲信号映射到哪种潜在的特征空间。

5.3 置换不变性训练和损失函数

5.3.1 训练方法

对于短时平稳的连续信号，通常会使用信噪比相关的函数或是最小平方损失函数来作为解码表示的损失函数。然而，在脉冲信号的解码表示任务中，真实的信号分量 $\boldsymbol{s}_k$ 高度稀疏，但是解码得到的脉冲信号分量 $\hat{\boldsymbol{s}}_k$ 是概率序列，在舍入前并不具有稀疏性。在训练阶段，概率序列 $\boldsymbol{P}_k^{(s)}$ 在阈值 t_s 附近的微小变化可能会导致脉冲信号分量 $\hat{\boldsymbol{s}}_k$ 从 0 变为 1 或从 1 变为 0。这将导致信噪比或最小平方损失的计算剧烈波动，网络参数收敛困难。此外，在训练过程中也并不容易估计阈值 t_s。因此，基于信噪比的损失函数不能够用于脉冲信号的解码表示，这也是连续信号解码表示和脉冲信号解码表示问题最大的不同之一。在这里，由脉冲分布距离定义的序列交叉熵损失函数将会是更好的选择。

在雷达脉冲信号分选任务中，还存在通道排列的问题。通道排列问题指的是：由于脉冲信号的每个分量在不同的输出通道上的任何排列都是等价的，导致相同的脉冲信号分量可以被分配给不同的通道。在解码表示和分选过程中，我们需要定义脉冲信号分量的排序关系，以避免出现通道排列问题。指定脉冲信号

分量系数矩阵 $\boldsymbol{H}_k$ 的范数按照迭代的轮次进行降序排列，即 $\|\boldsymbol{H}_k\| \leq \|\boldsymbol{H}_k - \mathbf{1}\|$，则能够确定输出通道的排列关系。在上述输出通道排列规则下，PRIμ_{PRI} 的期望随着脉冲信号分量标签 k 的增大而增加，即 $\mu_{PRI,k} \geq \mu_{PRI,k-1}$，这也是更自然的。换句话说，随着脉冲信号分量标签 k 的增大，脉冲信号中脉冲点的密度会减少。

当多个脉冲信号分量具有相似的 PRI，或者当脉冲信号受到 PRI 调制模式和环境噪声的影响时，上述方法可能是无效的。这是因为脉冲信号分量表示的范数 $\|\boldsymbol{H}_\boldsymbol{k}\|$ 和 $\|\boldsymbol{H}_{\boldsymbol{k-1}}\|$ 在这种情况下会比较相似。另一种更普适的方法是，在每次迭代过程中对脉冲信号分量 $\hat{s}_k$ 进行排序，接着计算哪一个真实的脉冲信号分量 s_k 最接近预测 $\hat{s}_k$，并将 $\hat{s}_k$ 分配给相应的分量标签 k，从而确定输出通道。这种训练方法被称为置换不变性训练。在测试阶段，该训练策略确保了预测的脉冲信号每个通道仅有单一分量的脉冲点。另外，脉冲信号帧之间也存在着一定的重叠。通过不同帧之间的重叠部分能够计算当前帧与先前帧重叠部分的相似度，从而确保当前的通道分配与先前一致。这确保了脉冲信号帧与帧之间输出通道顺序的一致性。通过这种方法，脉冲信号解码表示和分选的精度将不会被脉冲信号分量表示范数 $\|\boldsymbol{H}_\boldsymbol{k}\|$ 的模糊性所影响。

5.3.2 损失函数

基于上述分析，可以设计一种特殊的交叉熵损失函数来度量预测的脉冲信号分量和真实脉冲信号分量之间的距离。我们仅关心当 $\boldsymbol{s}(n) = 1$ 时，$\hat{\boldsymbol{s}}_k(n) = \boldsymbol{s}_k(n)$ 是否成立。而当 $\boldsymbol{s}(n) = 0$ 时则不会对 DPSM 算法的解码表示性能做进一步的要求。因为存在诸如 PRI 抖动调制或脉冲点丢失之类的环境噪声，当 $\boldsymbol{s}(n) = 0$ 时 $\hat{\boldsymbol{s}}(n) > 0$ 常常成立。而环境中的噪声等级越强，脉冲点被错误预测为出现的概率越高。如果概率分布 $P_k^{(s)}$ 相对平坦，则意味着脉冲帧出现的概率 $\boldsymbol{s}_k(n) = 1$ 被分配给了相邻的脉冲时间点。因此，当 $\boldsymbol{s}(n) = 0$ 时，将预测的脉冲信号帧强制约束为满足 $\hat{\boldsymbol{s}}_k(n) = 0$ 是不合适的，容易让训练过程不稳定。

假设 $\boldsymbol{ps}_k = \{n|\boldsymbol{s}(n) = 1 \wedge \boldsymbol{s}_k(n) = 1\}$ 和 $\boldsymbol{ns}_k = \{n|\boldsymbol{s}(n) = 1 \wedge \boldsymbol{s}_k(n) = 0\}$ 分别表示脉冲信号点的正样本集和负样本集，则使用范数排序训练的脉冲信号解码

表示交叉熵损失如下：

$$\mathcal{L}_S = -\sum_{k=1}^{K}\left[\sum_{n\in ps_k} \boldsymbol{s_k}(n)\log P_k^{(s)}(n) + \sum_{n\in ns_k}(1-\boldsymbol{s_k}(n))(1-\log P_k^{(s)}(n))\right]. \quad (5\text{-}29)$$

而如果使用置换不变性训练的脉冲信号解码表示交叉熵损失则表示为：

$$\mathcal{L}_S = -\sum_{k=1}^{K}\left[\sum_{n\in ps_k} \boldsymbol{s_k}(n)\log P_{\phi^*(k)}^{(s)}(n) + \sum_{n\in ns_k}(1-\boldsymbol{s_k}(n))(1-\log P_{\phi^*(k)}^{(s)}(n))\right]. \quad (5\text{-}30)$$

设 $\mathcal{P}$ 是所有可能的脉冲信号分量输出通道排列方式，那么 ϕ^* 是能够最小化交叉熵损失的通道排列方式，即：

$$\phi^* = \arg\max_{\phi\in\mathcal{P}} \sum_{k=1}^{K}\left[\sum_{n\in ps_k} \boldsymbol{s_k}(n)\log P_{\phi(k)}^{(s)}(n) + \sum_{n\in ns_k}(1-\boldsymbol{s_k}(n))(1-\log P_{\phi(k)}^{(s)}(n))\right]. \quad (5\text{-}31)$$

另外，为了估计脉冲信号分量的数目，还需要计算脉冲信号存在概率的交叉熵损失来训练 NDecoder：

$$\mathcal{L}_N = -\sum_{k=1}^{K}\log P_k^{(n)} - \sum_{k=K+1}^{K_{max}}\log(1-P_k^{(n)}). \quad (5\text{-}32)$$

我们默认当脉冲信号分量存在时，它会始终被优先分配给最靠前的通道。仅当 $K = K_{max}$ 时，最后一个通道才是有意义的脉冲信号概率序列表示。最终用于训练 RRN 的损失函数为：

$$\mathcal{L} = \mathcal{L}_S + \lambda\mathcal{L}_N. \quad (5\text{-}33)$$

λ 是加权两个损失函数 $\mathcal{L}_S$ 和 $\mathcal{L}_N$ 的超参数。

5.4 算法预处理和后处理聚类

5.4.1 脉冲信号分量歧义性

在雷达脉冲信号分选任务中，仅依靠 ToA 信息对具有复杂调制模式的雷达脉冲信号进行解码表示可能存在问题。如果既不知道 PRI 信息或脉冲信号分量的数量，也不知道调制模式的类型，那么仅依靠 ToA 信息在解码表示中容易引

起歧义。其中，参差 PRI 调制的分量歧义性是最典型的情况。仅使用 ToA 信息的脉冲信号解码表示算法很可能会将信号中的每个 PRI 分量 $\mu_{pri} + A_k$ 视为独立的脉冲信号分量。如图5-5所示，同一脉冲信号可以被解释为参差 PRI 调制的单分量信号或固定 PRI 调制的三分量信号。当然，在这个简单的例子中，我们可以分析出，如果认为他是固定 PRI 调制的脉冲信号，那么每个分量的 PRI 均相同，这是不合理的。然而，一旦环境的脉冲丢失率上升，或者同时出现多个调制类型的脉冲信号，仅通过 ToA 信息将不能完全地处理脉冲信号解码表示和分选问题。通过分析脉冲表示的帧间信息，RRN 中的双路径注意力可以在一定程度上缓解这一问题。然而，为了减少网络参数的数量和所需的训练样本，以使 DPSM 算法更可用，如果能够引入其他信息来帮助减少模糊性则会更好。所以用合适的预处理和后处理方法辅助 RRN 处理脉冲信号也是 DPSM 算法的关键之一。

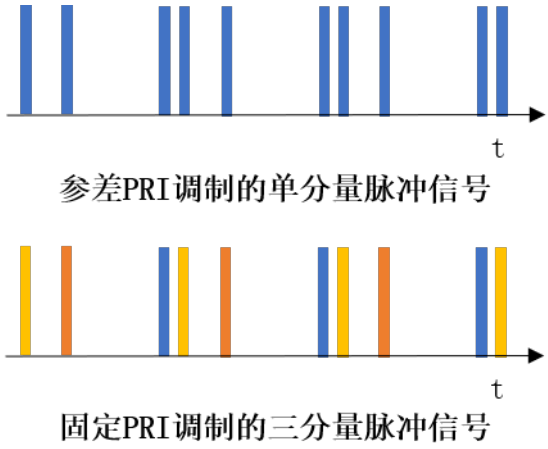

图 5-5: 参差 PRI 单分量脉冲信号和固定 PRI 多分量脉冲信号歧义性例子

我们提出了三种可能的方法使 DPSM 算法能更好地应用于具有多种调制模式的脉冲信号解码表示任务中。第一种方法是引入 PRI 信息来解决歧义。但实际应用中很难提前知道具体的脉冲信号 PRI 信息。第二种方法是引入 PRI 调制类型信息，相对于 PRI 信息这是比较容易获得的。因为 DPSM 算法只需要知道 PRI 的调制类型，而不需要知道具体的调制参数。第三种方法是将参差 PRI 和成组 PRI 的每个 PRI 分量视为单个 PRI 序列。该方法忽略了脉冲信号的组间的关系，并认为不同 PRI 的脉冲信号分量之间彼此独立。因此，在 RRN 处理结束之后，使用带噪声的 PDW 信息来帮助确定这些分量属于哪个脉冲信号分量。在本章我们基于相同脉冲信号分量的相关性对 RRN 解码表示和分选的结果进行后处理重聚类。

5.4.2 预处理微调

对于雷达脉冲信号分选任务而言，在训练结束后，RRN 可以直接应用于现实的电子战环境，并准确地对多个 PRI 调制的雷达脉冲信号进行分选。然而，在复杂环境和没有除了 ToA 以外的其它 PDW 先验知识的情况下，DPSM 算法不够鲁棒。雷达发射器经常连续发射脉冲信号，并且脉冲变化很快。在本章中，以 1μs 的时间分辨率进行脉冲信号处理，雷达发射器可以在 1 秒内生成 10^6 个脉冲采样点。特定的环境中，雷达接收机的处理器可能具有特定的先验知识，例如雷达发射器的类型、PRI 调制类型、噪声水平估计等。在应用之前，可以通过构建仿真数据或使用过去收集的雷达脉冲信号数据，基于先验知识对 RRN 进行微调。即使是从头开始训练模型，也只需要大约十分钟的脉冲信号数据。而几秒钟的数据已经足够用来微调网络参数。

具体地，在接收机接收到雷达脉冲信号之后，RRN 首先对脉冲信号的帧执行预分选。这些雷达脉冲信号帧和相应的分选结果可用于离线或在线微调 RRN。为了对模型进行微调以适应真实环境，可以收集一些实际中存在的分选好的脉冲信号分量，并通过去除异常点和手动校正来简单地处理它们。之后，它们可以被用于离线微调 RRN。在线脉冲信号分选期间微调 RRN 则应该首先确保预分选结果的准确性。通过使用序列搜索和后处理方法获得更准确的结果。DPSM 算法利用这些比较精确的脉冲信号数据在后端对 RRN 进行微调。经过一段时间的训练后，用更新的模型替换当前的老模型。在线脉冲信号分选和预处理过程互不干扰。该方法将低频的模型更新与高频的脉冲信号解码表示相结合，不仅能实时分选雷达脉冲信号，而且也保证了结果的准确性。以往基于 LSTM 的算法需要预热，即在使用前用一定数量的真实有监督数据重新训练 LSTM 模型，否则其表现将会非常不稳定。相对而言，它更接近于离线微调，这是该算法的必要步骤。在本章中，用于 DPSM 算法的在线微调则不需要通过手动标记脉冲信号分量来获得监督数据，而是通过 DPSM 算法流程，在后台自动地标记它们。DPSM 算法之所以能做到这一点，主要得益于它对脉冲信号解码表示和分选的高精度以及适当的后处理聚类方法。即使没有对 RRN 的微调和预处理，它也足以精确地表示脉冲信号和对其进行分选，这与基于 LSTM 的算法是不同的。

5.4.3 后处理重聚类

Algorithm 5.1 GMM 脉冲信号后处理重聚类算法

输入: RRN 预测的脉冲信号分量数目 K,
脉冲分量解码表示 $h_k, k = 1, \ldots, K$,
RRN 输出的脉冲点概率 $P_k^{(s)}, k = 1, \ldots, K$,
PDW 特征 $E_k, k = 1, \ldots, K$,
迭代次数 N.
输出: 重聚类的脉冲点概率 $\gamma_k, k = 1, \ldots, K$

for $k = 1, \ldots, K$ **do**
  $\gamma_k \leftarrow P_k^{(s)}$
  $\boldsymbol{e}_k \leftarrow [E_k; h_k]$
end for
for $n = 1, \ldots, N$ **do**
  for $k = 1, \ldots, K$ **do**
    $\boldsymbol{\mu}_k \leftarrow \frac{\gamma_k \cdot \boldsymbol{e}_k}{\gamma_k \cdot \mathbf{1}}$
    $\boldsymbol{\Sigma}_k \leftarrow \frac{\gamma_k \cdot (\boldsymbol{e}_k - \boldsymbol{\mu}_k)(\boldsymbol{e}_k - \boldsymbol{\mu}_k)^T}{\gamma_k \cdot \mathbf{1}}$
    $\alpha_k \leftarrow \frac{\gamma_k \cdot \mathbf{1}}{\mathbf{1} \cdot \mathbf{1}}$
    $\gamma_k \leftarrow \frac{\alpha_k \cdot p(\boldsymbol{s}_k | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{k=1}^{K} \alpha_k \cdot p(\boldsymbol{s}_k | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$
  end for
end for

通常有许多后处理算法可用于进一步提高 DPSM 算法的表示精度。基于脉冲信号分量跟踪的算法就是其中之一。此外，基于聚类的算法通常也是一个不错的选择。聚类算法利用 PDW 特征 $\boldsymbol{E}$ 对脉冲点进行聚类利用了充分多的雷达信息，但缺点是容易受到 PDW 噪声的影响，并且没有将时序关系利用到脉冲点的表示当中。但作为后处理算法，RRN 已经基于脉冲信号时序关系进行了解码表示，得到了可靠的脉冲分量概率序列。用概率序列对聚类算法进行初始化能够显著改善算法的准确性，减少 PDW 中的噪声影响，较好地利用 PDW 中的其它信息。

PDW 向量 $\boldsymbol{E}$ 以及 RRN 的脉冲信号解码表示序列 $P_k^{(s)}$ 作为后处理重聚类算法的输入。后处理算法的关键是如何更好地将 RRN 预分选结果与其它信息 $\boldsymbol{E}$ 相融合。一种可行的方法是使用可靠的解码表示脉冲信号 $\hat{\boldsymbol{s}}_k$ 作为脉冲信号分量点的标签，并初始化每个 PDW 分量的聚类中心。此外，在计算特征相似度时，RRN 预测的概率序列 $P_k^{(s)}$ 能够作为脉冲样本点的权重，将其作为高斯混合模型

（Gaussian Mixture Model, GMM）算法的初始化后验概率是一种后处理良好选择。RRN 还非常准确地预测出了脉冲信号分量的数目。利用该信息也能在后处理中直接确定聚类算法中簇的数量。因此，与直接执行聚类算法相比，后处理重聚类的准确性会大大提高。

正如算法5.1所示，后处理算法 GMM 利用了 RRN 输出的脉冲信号分量解码表示，脉冲点概率和预测的分量数目。算法中的 $p(\boldsymbol{s}_k|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)$ 表示高斯分布：

$$p(\boldsymbol{s}_k|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)=\frac{1}{(2\pi)^{n/2}\left|\boldsymbol{\Sigma}_k\right|^{1/2}}e^{-\frac{1}{2}(\boldsymbol{s}_k-\boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{s}_k-\boldsymbol{\mu}_k)} \tag{5-34}$$

利用 RRN 输出的脉冲点概率作为 GMM 初始化的后验概率，并将脉冲信号分量的解码表示 $\boldsymbol{h}_k$ 与每个分量脉冲点的 PDW 特征 $\boldsymbol{E}_k$ 结合起来作为 GMM 算法的输入，能够得到更加鲁棒的重聚类脉冲信号分选结果 $\boldsymbol{\gamma}_k$。

5.5 雷达脉冲信号解码表示理论

在传统脉冲信号解码表示的基础上，基于深度学习构建出雷达脉冲信号分选任务上的优化形式：

$$\min \quad \mathcal{L}_S(\mathcal{D}_S(\mathcal{G}(\boldsymbol{H})),\boldsymbol{y}_s) \tag{5-35}$$

$$\begin{aligned} s.t. \quad & \boldsymbol{H}=\mathcal{E}(\boldsymbol{s}) \\ & \mathcal{L}_N(\mathcal{D}_N(\mathcal{G}(\boldsymbol{H})),K)\leq\delta_n, \\ & \mathcal{L}_{cluster}(g(\boldsymbol{H},\boldsymbol{E}),\boldsymbol{c})\leq\delta_c. \end{aligned} \tag{5-36}$$

$\mathcal{E}$ 和 $\mathcal{D}_S$ 分别表示 RRN 的编码器和脉冲信号解码器，而 $\mathcal{G}$ 表示递归掩膜模块。编码器将脉冲信号帧 $\boldsymbol{s}$ 编码得到脉冲信号整体表示 $\boldsymbol{H}$，再经过 RMM 处理后分解为脉冲分量掩膜 $\boldsymbol{M}_k$，从而进一步得到脉冲信号分量表示 $\boldsymbol{H}_k$。目标是最小化预测的脉冲信号分量 $\hat{\boldsymbol{s}}_k$ 和真实脉冲信号分量 $\boldsymbol{s}_k$ 之间的置换不变的交叉熵损失函数 $\mathcal{L}_S$。真实脉冲信号分量 $\boldsymbol{s}_k$ 通过脉冲点分量标签 $\boldsymbol{y}_s$ 推断。在雷达脉冲信号解码表示和分选任务中，还需要预测雷达脉冲信号分量数目。$\mathcal{D}_S$ 表示脉冲分量数目解码器，通过 $\mathcal{L}_N$ 约束实际分量数目 K 和解码预测的分量数目从而完成网络的问题的优化。另外，g 表示对脉冲信号表示和脉冲描述字特征的变换函数，

$\mathcal{L}_{cluster}$ 则是特定的后处理聚类算法，在本章中使用的是 GMM 算法。$\boldsymbol{c}$ 指的是聚类中心向量，而 $\boldsymbol{s}_k$ 是真实的雷达脉冲信号分量。

在脉冲语音信号解码表示任务种，通过 DPSM 算法的 RRN 预处理，在线微调和后处理重聚类三个步骤能够很好地表示和分选雷达脉冲信号。该算法通过递归表示的方法结合双路径注意力机制，能够很好地将脉冲信号分量从信号表示中逐个分离出来，并以此为基础预测脉冲信号点的存在概率。通过预处理微调和后处理聚类，DPSM 算法还能利用先验信息应对可能存在的环境噪声和多变的脉冲信号调制模式。

5.6 实验和分析

5.6.1 数据集

表 5-2: 模拟实验中的雷达参数信息

参数	取值范围
时间分辨率	$1\mu s$
分量间最小间隔差	$20\mu s$
脉冲到达时间	$1\mu s - 60000\mu s$
脉冲重复间隔	$100\mu s - 600\mu s$
脉冲宽度	$1\mu s - 5\mu s$
脉冲到达角	$30° - 90°$
脉冲载频	5.0 MHz-15.0 MHz
分量数目 (K)	1-6
抖动率 (θ_J)	0-0.3
丢失率 (θ_M)	0-0.3
参差 PRI 集合大小 (K_A)	2-7
参差限制因子 (β_A)	50
滑动 PRI 集合大小 (K_B)	4-10
滑动限制因子 (β_B)	100
成组 PRI 集合大小 (K_C)	2-5
成组限制因子 (β_C)	100
组内 PRI 数量 (G)	2-8

雷达脉冲信号分选任务没有标准数据集，因此本实验中使用的数据都是模拟仿真生成的带噪雷达脉冲描述字。仿真的雷达参数如表5-2所示。雷达脉冲信

号数据集分为训练集、验证集和测试集。每个数据集由不同的随机种子独立生成。一帧脉冲信号 s 包含长度为 60 毫秒的 PDW。首先会将到达时间序列转换为脉冲信号的形式。脉冲信号帧的下标总是从 1 开始，到 60000 结束。一帧混合的脉冲信号帧 s 包括多个脉冲信号分量 s_k。训练数据集中存在由单分量脉冲信号组成的脉冲信号，它可以帮助网络学习单个雷达脉冲信号分量的特征。实验中使用的雷达参数或其平均值从表5-2中随机选择。脉冲重复间隔的平均值表示为 δ_{pri}，标准差为 $\delta_{pri} = 0.05\mu_{pri}$。脉冲宽度的平均值表示为 δ_{pw}，而标准差为 $\delta_{pw} = 0.5\mu_{pw}$。由于模型的时间分辨率仅为 $1\mu s$，因此脉冲重复间隔和脉冲宽度均向上取整。脉冲信号第一个脉冲点的到达时间从 1 到 μ_{pri} 中随机选择。构成交错脉冲信号的每个分量间的最小间隔差由参数“分量间最小间隔差”确定。在本模拟实验中，该参数为 $20\mu s$。

首先生成抖动 PRI 调制的雷达脉冲信号数据集。训练集包括总共 8000 条具有随机雷达参数的脉冲信号，长度之和为 8 分钟。验证集和测试集各包括独立生成的 4000 条脉冲信号，每条脉冲信号由 2 到 6 个单雷达脉冲信号分量混合而成。为了模拟复杂电子战环境中动态变化的信号，在生成过程中，根据抖动率 θ_J 和丢失率 θ_M 在脉冲信号中添加不同程度的高斯分布噪声。θ_J 会导致雷达脉冲信号的 PRI 在均值附近 δ_J 范围内随机变化，并且 $\delta_J = \theta_J \cdot \mu_{pri}$。然后再生成具有多个 PRI 调制模式和可变脉冲分量数目的雷达脉冲信号数据集。PRI 调制模式包括前文介绍的抖动 PRI、参差 PRI、滑动 PRI 和成组 PRI。雷达调制参数同样如表5-2所示，每个参数的具体含义可在本章第一节中找到。PRI 调制模式和参数在一定范围内随机选择，以更好地模拟实际环境中雷达的不同类型和工作模式，而非某一固定情况。除了 PRI 调制模式的改变，其他设置与抖动 PRI 调制的雷达脉冲信号数据集类似。

如图5-6所示，它表示一帧抖动 PRI 调制的六分量的雷达脉冲信号，图中不同颜色表示不同脉冲信号分量。其生成参数 $\theta_J = \theta_M = 0.1$。可以发现，尽管单个雷达脉冲信号分量是稀疏的，但混合的脉冲点出现得相对密集。在 $ToA = 50\mu s$ 附近，出现了多个脉冲分量点的重叠。在 $ToA = 850\mu s$ 附近，蓝色的雷达脉冲点明显出现了丢失。所以，当雷达脉冲分量数目较多时，解码表示和分选任务会变得更加困难。受环境噪声和调制模式的影响，传统的雷达信号分选算法将难以发挥作用。

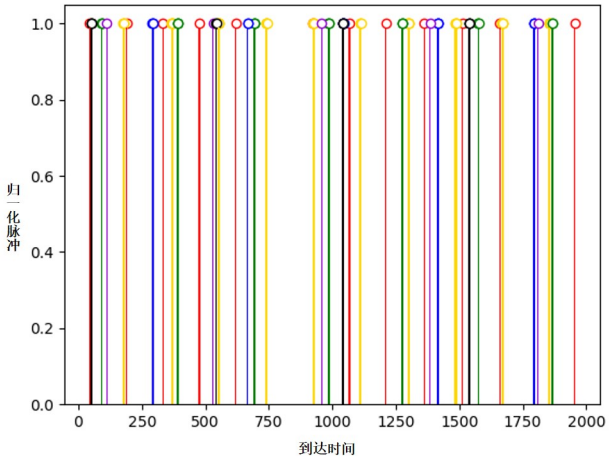

图 5-6: 六分量雷达脉冲信号序列

5.6.2 评价指标

雷达脉冲信号分选任务主要评估属于每个分量的雷达脉冲点是否被正确预测。评估主要包括两个方面。对分选算法最直接的评估标准是评估到达时间序列中每个脉冲点 ToA_i 预测的准确度。设从到达时间序列到脉冲分量标签的预测为 $g : ToA \rightarrow y$，那么预测的准确度表示如下：

$$Accuracy = \frac{1}{T}\sum_{i=1}^{T} I(g(ToA_i) = y_i). \tag{5-37}$$

然而，DPSM 算法实际上是直接处理雷达脉冲信号帧 $\boldsymbol{s}$，而不是到达时间序列，因此 ToA 预测的准确度不能直观地反映算法的性能。基于算法的损失函数形式，不妨用多分类问题的精确率和召回率来评估其性能。有四个相关指标：真阳性（TP）、真阴性（TN）、假阳性（FP）和假阴性（FN）。TP 表示集合 $\{n|\boldsymbol{s}(n) = 1 \wedge \boldsymbol{s}_k(n) = 1 \wedge \hat{\boldsymbol{s}}_k(n) = 1\}$ 中的元素数量，TN 表示集合 $\{n|\boldsymbol{s}(n) = 1 \wedge \boldsymbol{s}_k(n) = 0 \wedge \hat{\boldsymbol{s}}_k(n) = 0\}$ 的元素数量。FP 和 FN 的定义与前两者相似。因此，精确率 P 和召回率 R 如下：

$$P = \frac{TP}{TP + FP} \tag{5-38}$$

$$R = \frac{TP}{TP + FN}. \tag{5-39}$$

F1 则是精确率和召回率的调和平均值：

$$F1 = \frac{2PR}{P+R}. \tag{5-40}$$

P，R 和 F1 可以更准确、全面和直观地评估 DPSM 算法的优劣。如果雷达在某一时刻发出了脉冲信号，但 DPSM 算法没有正确预测出来，那么召回率将降低。如果该算法预测的虚假脉冲点太多，则精确率将降低。

5.6.3 仿真实验设置

仿真实验包括两个阶段，训练和测试。在将雷达脉冲信号数据输入 RRN 之前，首先会对其进行分帧，每帧长度为 60 毫秒。在训练阶段，编码器和解码器卷积的步长 step 为 32，卷积核大小为 $k = 16$。RMM 包括 2 个深度可分离空洞卷积块和双路径注意力块。每个卷积块包含 $B = 11$ 空洞卷积结构。空洞率从 2^0 增加到 2^{10}，卷积核的大小为 3。帧 $\boldsymbol{Z} \in \mathcal{R}^{F' \times S \times L'}$ 的长度 L' 为 300。编码后脉冲信号表示的特征维度为 $F = 256$，通过瓶颈层压缩后特征维度为 $F' = 64$。瓶颈层由一个 $F \times F'$ 线性层和一个带有 PReLU 激活的 $F \times F'$ 线性层组成。它将数据维度映射到子空间，从而压缩数据。

训练过程遵循课程学习的范式[113]。课程学习根据样本的难度为不同难度的训练样本分配不同的权重。在初始阶段，简单样本的权重最高。随着训练过程的继续，更难的样本的权重将逐渐增加。这种为样本动态分配权重的过程被称为课程。在雷达脉冲信号分选任务中，固定 PRI 调制和无环境噪声的两种任务是最简单的，而在未知多个分量的多 PRI 调制模式下的带噪雷达脉冲信号分选任务是最困难的。所以，训练过程共包括三个阶段。首先使用固定和抖动 PRI 调制模式，六雷达脉冲信号分量的无丢失数据进行预训练。然后用多 PRI 调制模式的六雷达脉冲信号分量数据来训练 RRN。此时抖动 PRI 调制的脉冲信号的权重相对较低，因为它相对简单。最后，我们使用随机 1 到 6 个分量的雷达脉冲信号，随机选择多种 PRI 调制模式，设定环境噪声强度为 $\theta_M = \theta_J = 0.3$ 进行脉冲信号解码表示训练。通过这种方法能从易到难地训练 RRN。它将训练所需的雷达脉冲信号总长度缩短到数十分钟。在训练数据中包括单分量雷达脉冲信号（$K = 1$），允许网络更容易地学习单个雷达脉冲信号本身的特性。在第一个

阶段，学习率 $lr_1 = 0.001$。而在第二和第三阶段，$lr_2 = lr_3 = 0.001$。每个阶段的训练轮数 epoch 均为 40。训练中使用的优化器是 Adam。当 F1 连续两个轮次没有上升时，将学习率降低到前一个轮次的 0.7 倍。阈值 t_s 和 t_n 属于算法的超参数，通过网格搜索来选择这两个阈值。我们将验证数据集上 F1 得分最高的阈值作为 DPSM 算法的阈值。最后，我们设置阈值围为 $t_s = 0.8$，$t_n = 0.5$。

在仿真实验中，我们在不适用任何预处理和后处理算法的情况下，评估 DPSM 算法在抖动 PRI 调制的雷达脉冲信号数据集中的性能。首先在不同的雷达发射器和不同的噪声水平下对其进行评估，然后再加入预处理和后处理算法观察其性能。在预处理期间，将假设存在特定的先验知识，并使用生成的数据来微调网络，具体与上文介绍的微调方法一致。在后处理过程中，首先通过 RRN 的粗预测来初始化 K-Means 算法和 GMM 算法。然后利用额外的 PDW 参数 $\boldsymbol{V}$ 进行重聚类。此外，还将 DPSM 算法与一些常用的脉冲信号分选算法进行了比较，以证明 DPSM 算法的准确性和鲁棒性。最后，我们扩展了 DPSM 算法，在多个 PRI 调制数据集上进行雷达信号分选任务，分选不同 PRI 调制模式的雷达脉冲信号。

在没有特别说明的情况下，仿真实验均在未知脉冲分量数目，未知具体 PRI 和未知 PRI 调制模式的配置下测试 DPSM 算法。

5.6.4 分选结果和可视化

RRN 的编码器将雷达脉冲信号帧编码为表示系数矩阵 $\boldsymbol{H}$，然后递归地分选出每个分量。图5-7中的脉冲信号表示掩膜 $\boldsymbol{M}$ 可视化地展示了脉冲信号表示 $\boldsymbol{H}$ 中哪个分量表示 $\boldsymbol{H_k}$ 占主导地位。脉冲信号表示掩膜点 $M(i,j)$ 取 $-10, 1, 2, 3, 4, 5$ 的离散值，表示不同的脉冲分量 k 分别支配 $\boldsymbol{H}$ 中表示矩阵中的 (i,j) 位置。当所有 $H_k(i,j)$ 的值都很小时，$M(i,j) = -10$。而当 $H_k(i,j) > H_{-k}(i,j)$ 并且 $H_k(i,j)$ 的值足够时，$M(i,j) = k$。其中 $\boldsymbol{H_{-k}}$ 表示除去 $\boldsymbol{H_k}$ 以外其它脉冲分量表示的总和。如上文所述，$\boldsymbol{H} \in \mathcal{R}^{F \times L}$，所以同样的，$\boldsymbol{M} \in \mathcal{R}^{F \times L}$。$i$ 是 $\boldsymbol{M}$ 的时间维度，也就是 L 维度的下标。j 是 $\boldsymbol{M}$ 的频率维度，也就是 F 维度的下标。从中可以发现，每个脉冲表示点在不同的位置被不同分量所主导，这种特定的表示模式是帮助网络识别每个脉冲信号分量的基础。时间维度中的掩膜点呈现出特定的周期性，并且与特定的脉冲分量相关联。

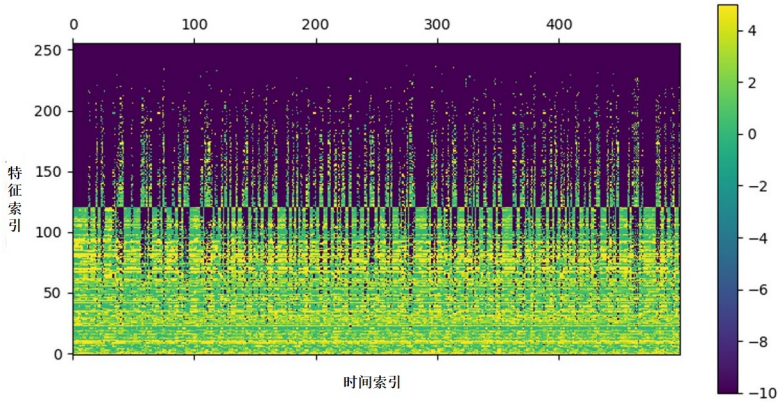

图 5-7: DPSM 的脉冲信号表示掩膜 $\boldsymbol{M}$

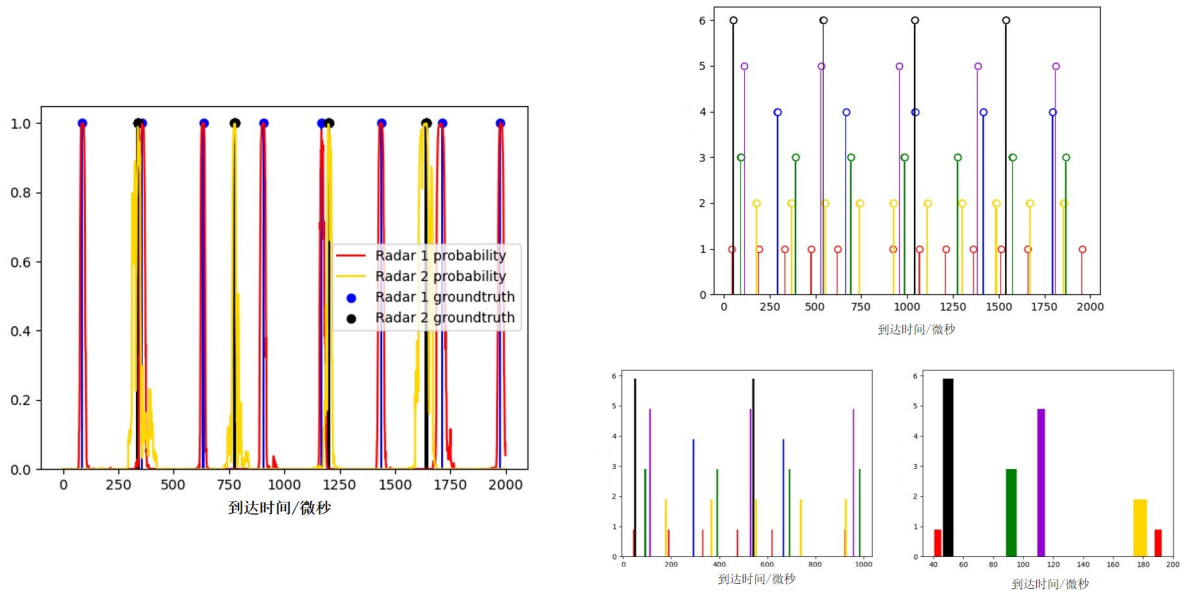

图 5-8: (a) 左图：脉冲点概率 $P_1^{(s)}$，$P_2^{(s)}$ 和对应的真实脉冲；(b) 右图：脉冲信号分选结果 (下面两图展示了更多细节)

RRN 处理后每个时间点的脉冲分量出现概率 $P^{(s)}$ 和两个分量的真实脉冲点如图5-8所示。这与先前的假设一致，RRN 预测的脉冲概率在真实脉冲点处最大，但有部分概率被分配给了真实脉冲点附近的时间点。从图中可以发现，即使不同脉冲分量点之间的时间间隔很小，算法也可以正确地对它们进行分选。分选过程中，可以通过计算预测的每个雷达分量脉冲 PRI 的平均值和标准差，并去除超过标准差范围三倍的 PRI 脉冲点来执行异常检测和修正。六个雷达脉冲分量的分选结果如图5-8（b）所示。从中能够发现，其预测结果与真实情况几乎相同，这证明了 DPSM 算法可以在多分量噪声环境中精确地进行脉冲信号解码表示和分选。

5.6.5 环境敏感度实验

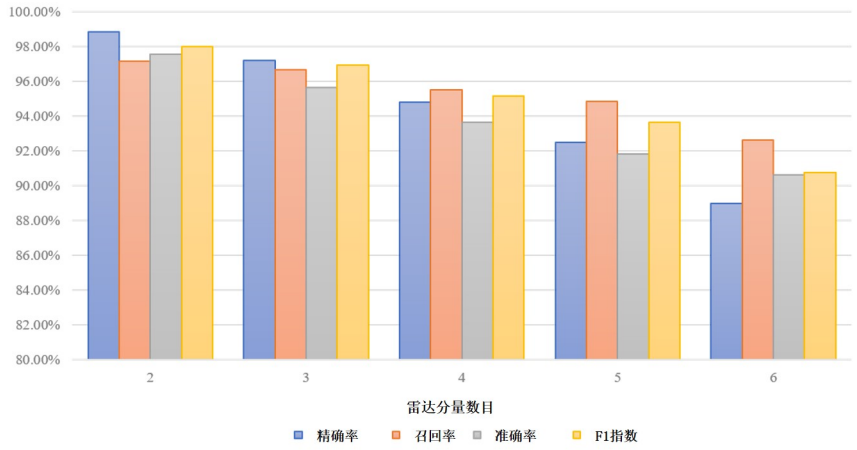

图 5-9: 不同最大脉冲分量数目下雷达脉冲信号分选效果图

在实际的电子战环境中，通常有多个雷达信号发射器，从而有多个脉冲信号分量。它们的数量会随着时间的变化而变化。这要求算法能够动态预测特定环境中的脉冲信号分量数目，并且预测得应该足够准确。假设脉冲信号分量的最大数目为 $K_{max} = 6$。图5-9展示了不同分量数目下 DPSM 算法的分选结果。当分量数目持续增加时，分选效果将逐渐变差。但是，即使存在六个脉冲信号分量，DPSM 算法的解码表示准确率也能达到 90.6%。在这样的准确率下，算法可以比较容易地估计每个分量的实际 PRI，并校正预测的脉冲到达时间序列。此外，RRN 预测脉冲分量存在概率的置信度超过 99%，并且分量数目的预测准确率为 100%。换句话说，当雷达脉冲信号分量 k 存在时，$P_k^{(n)} > 0.99$。RRN 总是能够正确预测雷达脉冲信号的数目。这表明该算法确实对脉冲分量数目的变化做出了鲁棒的响应。应该注意的是，准确预测分量数目的意义不仅在于提高 RRN 解码表示的准确性。更重要的是，它还为后处理方法提供了重要信息。

图5-10中的曲线是不同缺失率和抖动率下排序的精确召回（Precision-Recall，P-R）曲线。横轴是召回率，纵轴是精确率。随着阈值 t_s 从 0 增加到 1，精确率逐渐增加，召回率逐渐降低。仅仅 P-R 曲线中的某一点不能完全衡量模型的性能，只有 P-R 曲线整体才可以更全面地评估该模型。P-R 曲线下的面积越大，模型性能越好。从图中可以看出，噪声的存在会影响模型的性能。由于 PRI 未知，当脉冲点丢失或因抖动而偏离 PRI 平均值太多时，模型无法确定哪个脉冲信号分量出现了异常。有时甚至会出现多个脉冲点相互重叠的情况，此时更难判断。

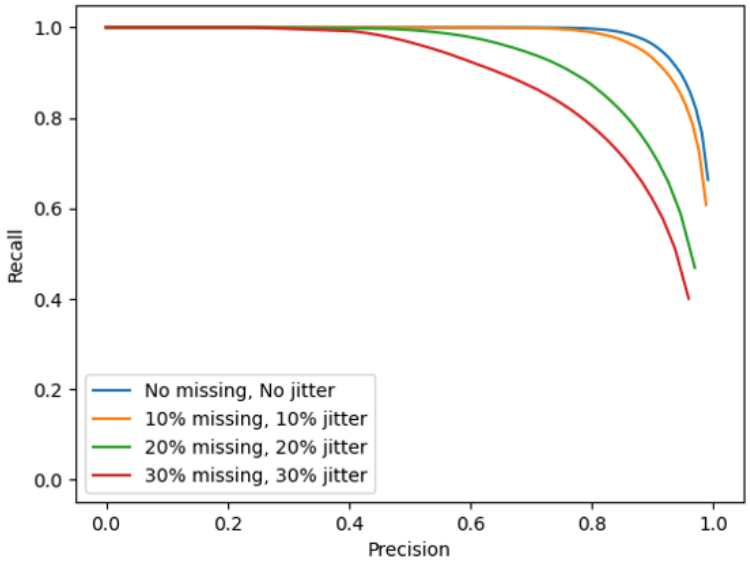

图 5-10: 不同丢失率和抖动率下的精确率-召回率曲线

表 5-3: 不同环境噪声下未知分量数目雷达脉冲信号分选效果表。

噪声等级	Precision	Recall	Accuracy	F1
$\theta_M = 0.0, \theta_J = 0.0$	0.9205	0.9409	0.9496	0.9306
$\theta_M = 0.0, \theta_J = 0.1$	0.9104	0.9213	0.9323	0.9158
$\theta_M = 0.1, \theta_J = 0.0$	0.9129	0.9421	0.9468	0.9273
$\theta_M = 0.1, \theta_J = 0.1$	0.9063	0.9240	0.9303	0.9151
$\theta_M = 0.2, \theta_J = 0.2$	0.8312	0.8377	0.8617	0.8344
$\theta_M = 0.3, \theta_J = 0.3$	0.8146	0.7646	0.8172	0.7888

但即使当 $\theta_M = \theta_J = 0.3$ 时，测试集上的 RRN 准确率仍能达到约 80%。在这样的苛刻条件下，假设丢失和抖动彼此独立，则脉冲点受到影响的概率为 51%。对于这样的脉冲点，可以正确预测的概率约为 2/3。表5-3中显示了随丢失率和抖动率更具体的统计数据的变化。当丢失率和抖动率小于 0.1 时，噪声对 RRN 的性能几乎没有影响。相对而言，PRI 抖动对 DPSM 算法的影响大于脉冲点的丢失。上述仿真实验证明，即使没有任何预处理和后处理，DPSM 算法也仍然可以对电子战环境中受噪声严重影响的未知数量雷达脉冲信号进行鲁棒分选。

5.6.6 对比试验

表5-4中比较了 DPSM 算法与其他常用的雷达脉冲信号解码表示和分选算法的性能。需要注意的是，由于雷达脉冲信号分选任务没有通用的数据集，因此原始论文中的数据可能会不具有可比性。然而，由于在我们的数据集上环境噪声等级总是与其他论文中相等或更强的，并且本实验中脉冲信号分量数目未知，特定的抖动 PRI 调制参数也未知，所以在本实验中的解码表示和分选总是比对比论文中更困难的。为了确保比较的公平性，表中对比的算法结果是实现的和原论文数据两者间的较大值。在表5-4中，“固定 PRI”表示 PRI 的平均值 μ_{pri} 在模拟实验中是固定的。“2, w/o noise”意味着脉冲信号分量数目为固定的两个，并且没有脉冲的丢失和抖动。在预处理过程中会缓存处理后的雷达脉冲信号片段，然后使用准确的标签（或具有高置信度的后处理标签）对网络进行 10 秒的微调，然后更新 RRN 的参数，最后对后续脉冲信号帧进行预测。在后处理过程中，我们分别使用 K-Mean 和 GMM 算法进行重聚类。聚类中簇的数量由 RRN 预测的分量数目 K 确定，簇的初始中心则由置信度大于 0.8 的 PDW 参数 $\boldsymbol{E}$ 的加权平均得到。模拟实验中生成的 PDW 变量 $\boldsymbol{E}$ 同样具有大量的噪声和丢失。只有当基于 ToA 的脉冲信号解码表示和分选算法得到足够准确的结果时，后处理聚类算法才能对分选产生积极影响。整个 DPSM 算法在测试阶段的实时率小于 0.5，即 60 毫秒的雷达脉冲信号可以在 30 毫秒内处理完毕。因此，DPSM 算法能够实现实时处理。

基于聚类的分选算法 K-means，GMM 和 NN-cluster 均直接使用了额外的 PDW 参数 $\boldsymbol{E}$，并且只能将聚类数量设置为固定的 $K_{max} = 6$。这些算法的性能不太好，因为未知且不固定数目的脉冲信号分量对聚类分选算法的性能有很大影响。然而，如果聚类算法被用作 DPSM 算法中的后处理模块，则能帮助 RRN 做出更准确的预测。因为此时聚类的数量是已知的，并且存在初始化的聚类中心。当信噪比较低且脉冲信号分量数目较多时，SDIF 和 IHM 算法不能很好地工作。随着雷达辐射源数量的增加和环境噪声的增加，基于序列搜索的方法的效果也会急剧下降。去噪编码器（Denoise Autoencoder, DA）只能处理固定 PRI 调制的情况，但它在这种情况下工作得很好。除了使用 ToA 信息，该算法还使用脉冲幅度以及通过到达时间和脉冲幅度构建的一些其他特征作为输入。LSTM 算法对

表 5-4: 不同算法雷达脉冲信号分选结果对比表

算法	Precision	Recall	Accuracy
K-Means[56]	/	/	0.3721
GMM[61]	/	/	0.7151
NN-cluster[62]	0.7021	0.7442	0.7573
SDIF[69,114]	/	0.3587	/
IHM[55]	/	0.5120	/
DA[67] (fixed PRI)	0.8356	0.8452	0.8598
LSTM[64] (2, w/o noise)	0.9672	0.9836	0.9749
RRN (fixed PRI)	**0.9307**	**0.9642**	**0.9242**
RRN (2, w/o noise)	**0.9921**	**0.9791**	**0.9847**
RRN	**0.8146**	**0.7646**	**0.8172**
RRN + post. K-means	/	/	0.9441
RRN + post. GMM	/	/	0.9522
pre. + RRN	0.9166	0.8838	0.8907
pre. + RRN + post. GMM	/	/	**0.9764**

脉冲信号的起始点非常敏感，通常实际使用时需要预热。经过调研，现有的深度学习方法在当前设置下都无法取得良好的效果。RRN 本身仅使用 ToA 信息进行脉冲信号解码表示，精度能够达到 81.72%。仿真实验表明，适当的预处理和后处理可以在不同程度上提高 DPSM 算法的预测精度。DPSM 算法在所有对比算法中实现了 97.64% 的最优准确率。在两个脉冲分量和无噪声的情况下，DPSM 的性能也优于 LSTM 算法。在固定 PRI 调制的条件下也优于去噪自编码器算法。

5.6.7 雷达脉冲信号调制模式实验

最后，我们基于多 PRI 调制模式的雷达脉冲信号数据集进行了仿真实验，如表5-5所示。首先测试了在脉冲分量数目未知，PRI 未知并且 PRI 调制模式未知的情况下，抖动 PRI、滑动 PRI、参差 PRI 和成组 PRI 的分选准确率。所有的雷达调制模型参数都是从表5-2的参数取值范围中随机选择的。每个雷达脉冲分量的调制类型从四种调制类型中随机选择，多个脉冲分量可能会具有相同的调制

类型。从前文对抖动 PRI 调制的脉冲信号模拟实验中可以发现，当仅使用 ToA 信息时，也能比较好地对固定 PRI 调制和抖动 PRI 调制的脉冲信号实现鲁棒和准确的分选。但如果脉冲信号是由多个 PRI 调制类型的数个脉冲分量组成的，那么只使用 ToA 信息进行分选不能得到最好的结果。相对来说，滑动 PRI 和抖动 PRI 调制模式对精度的影响较小，仅使用 ToA 信息也可以实现约 90% 的排序精度。这是因为这两种 PRI 调制类型基本上属于单模调制，受歧义性影响较小。参差 PRI 和成组 PRI 则都属于多模调制，当仅使用 ToA 信息进行分选时，准确度较低。所以，当没有额外信息时，多模调制的雷达脉冲信号分选精度会大大降低。在这种情况下，分选的结果是不明确的。尽管存在歧义性性，DPSM 算法仍然可以利用诸如空洞卷积和双路径注意力机制等结构对脉冲信号进行解码。与之前的模拟实验相比，因为设置的丢失率和抖动率相对较小（$\theta_M = \theta_J = 0.1$），因此 RRN 对抖动 PRI 调制的分选准确率有一定的提高。滑动 PRI 调制的分选准确率高于前者，这是因为其 PRI 变化模式更易于预测。参差 PRI 和成组 PRI 调制的分选准确率约为 70%，远低于抖动 PRI 和滑动 PRI 调制的脉冲信号。这说明了歧义性对这两种 PRI 调制模式分选有比较大的负面影响。

表 5-5: DPSM 算法在具有不同信息的多个 PRI 调制模式下的准确率

PRI 调制模式	无额外信息	调制类型信息	PRI 信息	带噪 PDW 信息
抖动 PRI 调制	0.8969	**0.9881**	0.9510	0.9298
滑动 PRI 调制	0.9010	**0.9783**	0.9486	0.9262
参差 PRI 调制	0.6848	**0.9415**	0.9210	0.8079
成组 PRI 调制	0.6954	0.8241	**0.9485**	0.8530
All	0.8495	0.9444	**0.9491**	0.9083

在本实验中考虑了三种额外信息来解决歧义性：调制类型、PRI 和带噪 PDW 信息。“调制类型信息”指的是预先知道每个雷达脉冲分量的调制类型，但不知道具体的调制参数。在这种情况下，有关于发射器调制类型的额外信息是可以获得或者是可以推断的。也就是说，即使调制类型未知，也可以通过深度学习算法来准确预测。“PRI 信息”是指我们提前知道每个脉冲分量的 PRI，这是一个非常强的额外信息，在实际情况下不很少出现。“噪声 PDW 信息”是最容易获得的额外信息，也经常被用于脉冲信号分选任务。在仿真实验中，将“噪声 PDW

信息”视为额外信息，并通过后处理聚类算法解决各种脉冲调制类型条件下的雷达脉冲信号分选问题。PRI 调制信息可以将 DPSM 算法的分选准确率提高到 90% 以上。成组 PRI 调制也增加了十多个百分点。当事先知道雷达脉冲信号的 PRI 信息时，DPSM 算法对于任何调制模式都具有很高的分选准确率，准确率达到 94.91%。唯一的缺点是，当只知道 PRI 信息而不知道调制类型时，网络的微调需要数分钟的样本。这也间接表明调制类型对于具有多种调制模式的雷达脉冲信号解码表示是非常重要的信息。还可以考虑将参差 PRI 的参数组 A_k 和成组 PRI 的参数组 C_k 视为单个脉冲分量，然后使用带噪 PDW 信息来辅助分选。经过 RRN 预分选后，再根据每个脉冲分量之间的相关性或 PDW 参数之间的相关性对分选后的脉冲点进行重新聚类，以间接实现参差 PRI 和成组 PRI 调制信号的分选。由于 RRN 对具有单模 PRI 变化模式（如抖动 PRI 和滑动 PRI）的脉冲信号具有较高的分选准确率，在将参差 PRI 和成组 PRI 转换为多个固定 PRI 后，分选准确率主要取决于脉冲分量其他 PDW 参数的噪声水平。其他 PDW 参数有噪声或缺失，所以参差 PRI 和组 PRI 调制的分选准确率低于具有调制类型信息或 PRI 信息的情况。但其优点在于，除了 PDW 参数之外，该方法不需要任何其它的先验知识。此时，参差 PRI 和成组 PRI 调制的分选准确率能达到 80% 以上。如果存在调制类型或 PRI 信息，多模调制类型的分选结果与单模调制类型没有太大差异。调制类型信息能够比较容易地从到达时间序列中获得或推断。此时，DPSM 算法可以精确地表示和分选任何 PRI 调制类型的雷达脉冲信号。

5.7 本章小结

本章受到基于非负矩阵分解的盲源信号分离算法的启发，基于脉冲信号解码表示理论，提出了深度脉冲信号掩膜算法，其核心是递归表示网络。递归表示网络的编码器-解码器结构和设计的损失函数解决了非平稳脉冲信号的表示问题。递归掩膜模块的空洞卷积块增加了感受野，深度可分离卷积结构提高了网络的解码表示速度。双路径注意机制整合了帧内和帧间的 PRI 信息，更有利于不同 PRI 调制模式的识别和分选。基于脉冲信号分量表示掩膜的方法使递归表示网络在高缺失率和重叠率时也能获得更好的结果。DPSM 算法还能够处理未知 PRI 情况，并预测脉冲信号分量数目。预处理微调和基于高斯混合模型的后

处理聚类算法使 DPSM 能够利用 PDW 中的带噪信息，并获得更准确的表示和分选结果。通过仿真实验，证明了 DPSM 算法可以对具有未知脉冲信号分量数目和未知 PRI 的抖动 PRI 调制雷达脉冲信号进行准确分选。它对噪声具有很强的鲁棒性，即使仅使用 ToA 信息进行分选，分选准确率也比较高。此外，仿真实验还证明了在已知 PRI 或未知 PRI 但已知 PRI 调制类型的情况下，DPSM 算法可以高精度地分选多种 PRI 调制的雷达脉冲信号。

第六章　连续信号解码表示研究

本章分为七个部分，第一个部分介绍解决基于深度学习的连续信号解码表示问题的关键，并整体介绍实时频域卷积时序网络 RTFCRN；第二个部分介绍如何设计 RTFCRN 的模型结构以实现实时流式运算和多通道信号处理；第三个部分以语音增强任务为例，结合人耳主观感知指标，介绍 RTFCRN 基于复数掩膜的预测目标和损失函数；第四个部分介绍用于 RTFCRN 模型轻量化算法，包括基于敏感度的剪枝算法、基于特征的蒸馏算法和参数动态量化方法；第五个部分结合 RTFCRN 算法说明连续信号解码理论在语音信号增强问题上的具体形式；第六个部分在多通道实时语音增强应用任务上用多个开源中文数据集评价语音信号增强的效果，并通过参数量和实时率衡量 RTFCRN 模型压缩的效果；最后一个部分对本章进行小结。

6.1　连续信号的解码表示

在本章中，我们主要关注连续信号解码表示问题，并提出了实时频域卷积时序网络（Real-Time Frequency Convolutional Recurrent Network, RTFCRN）用以建模连续信号解码表示并主要解决信号增强任务。它能够更好地融合频率维度和空间通道维度的信息，对输入的多通道混响信号进行鲁棒建模，并解码得到符合干净的信号。在输入长度为 40 毫秒信号，并进行流式信号表示和增强时，RTFCRN 仍然能保持与离线信号增强一致的效果。我们还对 RTFCRN 分别使用剪枝算法和知识蒸馏算法进行了网络压缩，并进行了动态量化，减少网络的参数量和存储量。在实验和分析部分，我们会在语音信号增强任务中，融合多个开源语音数据集及噪声数据集，动态地生成房间冲激响应构造混响噪声数据集。接着在这个统一的数据集上将 RTFCRN 与其它语音增强算法进行对比，验证连续信号解码表示方法指导的 RTFCRN 在语音增强性能、网络参数量和降噪时延上均优于其它算法。本章的连续信号解码表示研究主要关注语音信号，优化语音信号对人耳听感的效果，其它的连续信号也可以使用类似方法建模。相对于第四章和第五章，本章的工程性较强。

6.1.1 连续信号表示

连续信号的表示研究一直都是信号处理领域的热门话题。常用的最常用傅里叶变换能够将连续信号的频率分量抽取出来，并在频率域直观展示。然而，傅里叶变换只适合于处理平稳信号，对于非平稳信号，因为信号的频率特性会随时间变化，为了捕获这一时变特性，需要对信号进行时频分析，从而获得非平稳信号的时频表示。

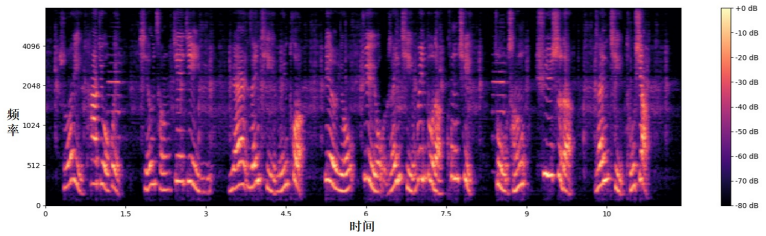

图 6-1: 短时傅里叶变换时频图

短时傅里叶变换、小波变换和希尔伯波特黄变换等算法，均是为了解决傅里叶变换对非平稳信号表示存在问题而提出的。图6-1展示了短时傅里叶变换到频谱图。在本章中，我们在短时傅里叶变换时频图基础上对连续信号进行表示和处理，从频率维度和时间维度两个方面进一步提取时间和频率信息，得到带噪信号的隐空间表示。并且，本章中处理的语音信号是多通道的信号，其中还存在的因为麦克风位置不同而导出的空域信息。所以我们还会通过相位角差和后置波束形成等方法进一步提取空间信息。从而，最终期望通过深度学习方法从带噪信号中得到融合了时间、频率、空间三个维度信息的鲁棒表示，并从信号表示中分离出去噪的成分，解码回信号所在的时域空间。

另一方面，连续信号中也依然存在着多义性，比如带噪的语音信号中存在着说话人信息、语音信息和噪声信息。在进行语音增强任务时，如果能够在语音信号表示中识别和提取出说话人身份，将其与语音和噪声信号表示解耦，并利用说话人身份信息辅助降噪，语音增强任务通常能够取得更好的效果。

6.1.2 实时性和参数量的制约

连续信号的解码表示问题一般会被应用在信号增强、信号分离和语音识别等任务中。其中信号增强和分离等任务属于信号处理的前端部分，通常是为后续

的通话、录制、会议等场景所服务的。所以，对于连续信号的解码表示问题，我们不仅仅要关注信号表示质量本本身，也要考虑模型和算法的复杂度 $N(\theta)$。这主要体现在两个方面：模型的存储量和模型的实时性。这两者是制约连续信号解码表示算法是否能够实际应用的重要因素。

大部分基于深度学习的信号增强算法，特别是已有的多通道语音信号增强算法网络参数量较多，无法满足实时性要求。有的算法虽然理论上能够实现实时，但在测试时如果流式地输入信号，信号表示的效果不尽如人意。这些问题产生的主要原因有以下几点：（1）连续信号解码表示模型本身设计的不够轻量。常见的语音信号增强算法中，大部分属于单通道语音信号增强，主要关注离线语音信号的增强效果，并且有的会使用参数量比较多的复数网络结构。而使用了神经网络波束形成算法的多通道语音信号增强算法参数量更多，速度也会更慢；（2）神经网络结构本身不支持实时连续信号解码表示。比如普通的注意力结构和卷积结构通常并不满足因果性；（3）神经网络训练范式以及训练目标与任务目标不匹配。离线连续信号解码表示处理得到的信号信噪比较高，但实际流式处理时效果却不够好；（4）连续信号表示时，主要关注时间维度信息的提取，频率维度和空间信息提取不足；（5）没有使用系统的网络轻量化方法对神经网络进行冗余信息的过滤，使得网络更加轻量化。

6.1.3 实时频域卷积时序网络

在本文中，我们提出了一种轻量级的多通道连续信号解码表示算法，其核心是实时频域卷积时序网络模型（Real-Time Frequency Convolution Recurrent Network，RTFCRN）。该模型在对各种场景噪声和混响具有良好适应性的前提下，参数量也较少。此外，它可以实现实时性，满足流式信号处理的要求。RTFCRN的总体架构如图6-2所示。网络的输入是 M 个通道的带噪连续语音信号 x，其中第 m 个通道的带噪连续语音信号可以表示为：

$$x_m = s \otimes h_{s,m} + n \otimes h_{n,m}. \tag{6-1}$$

其中 s 是干净的语音信号，n 是噪声信号。而 $h_{s,m}$ 和 $h_{n,m}$ 分别对应于第 m 个麦克风的语音混响脉冲响应和噪声混响脉冲响应。$\otimes$ 是卷积运算符。在本章中，我

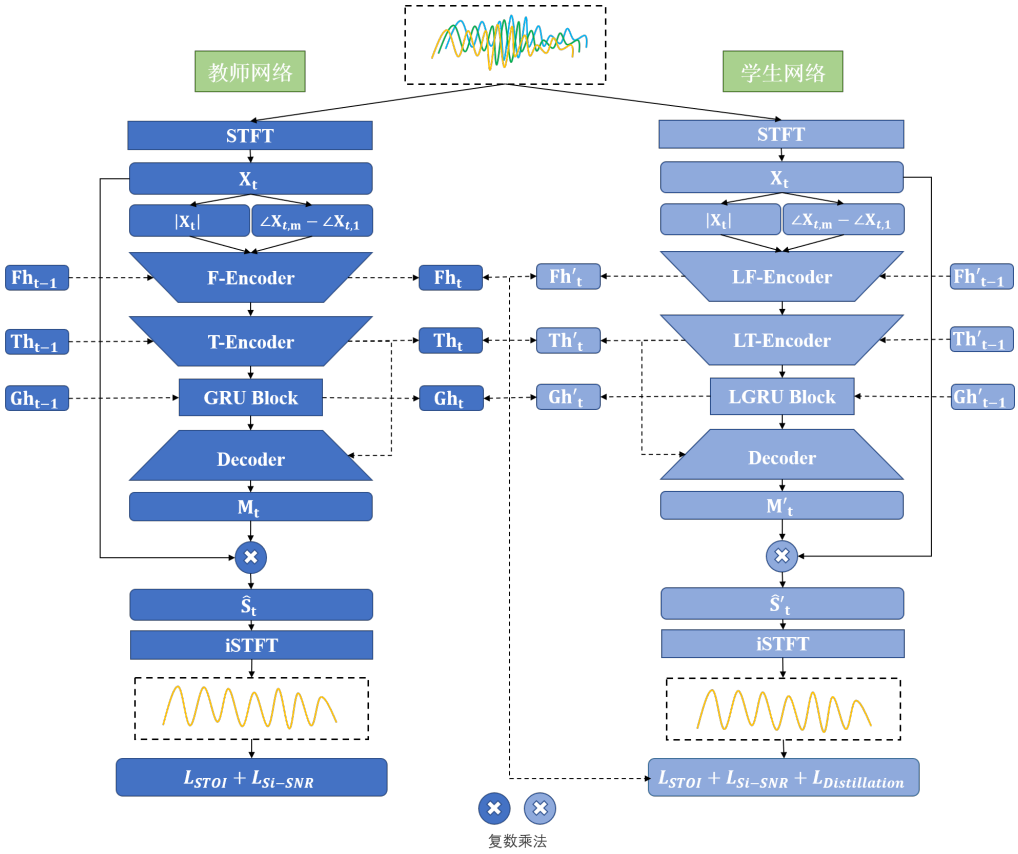

图 6-2: 实时频域卷积时序网络模型结构

们主要关注多通道连续语音信号增强任务，而不会从信号中去除混响。我们希望 RTFCRN 能够对连续语音和噪声信号进行比较准确的表示，从而能够从语音信号中去除噪声，并获得相对于第一个通道的干净混响语音信号。我们分别构建编码器 $\mathcal{E}_\theta$，分离函数 f_θ 和解码器 $\mathcal{D}_\theta$，以得到干净的信号 $\hat{s}_1 = \mathcal{D}_\theta(\mathcal{E}_\theta(f_\theta(x)))$。从人类听觉的角度上，$\hat{s}_1$ 越接近 $s_1 = s \otimes h_{s,1}$，RTFCRN 的连续信号解码表示性能越好。

实时频域卷积时序网络模型的输入信号 x 通过短时傅里叶变换后，获得 M 通道的复频谱 $X \in \mathbb{C}^{M \times F \times T}$，其中 F 是频率维度，T 是时间帧维度。为了更好地提取空间信息，可以分别计算 M 个通道的幅度谱 $|X|$ 和相位角 $\angle X$。然后将它们转换为 M 个通道的幅度谱和 $M-1$ 个通道的相位角差，其中相位角差是相对于第一参考通道的相位角度差，即：$\angle X_M - \angle X_1$。RTFCRN 包括两个编码器、一个门控循环单元（Gated Recurrent Unit，GRU）模块和一个解码器。它可以预测复数理想比值掩码（complex Ideal Ratio Mask，cIRM），并将其与原始复数谱 X 相乘，以获得参考信道的干净信号的频谱表示；也可以直接通过解码器预测干净信号

的频谱表示。前者是预测信号的间接表示，即干净信号与带噪信号表示的比值；而后者是预测信号的直接表示，即预测干净信号表示本身。为了优化 RTFCRN，我们使用了改进的基于信噪比的损失函数和基于人耳听觉的损失函数。它使模型训练更符合真实场景。在训练 RTFCRN 之后，我们使用基于敏感度的剪枝算法和基于 Overhual 的模型蒸馏算法对模型进行压缩和量化，最终获得了一个轻量级的连续信号解码表示模型。

6.2 RTFCRN 模型结构

6.2.1 流式信号表示

如果希望语音信号解码表示模型能够实现流式信号表示，那么需要设计特定的神经网络结构来支持这一点。在传统的卷积神经网络中，卷积核的设计违背了时间先后顺序的因果约束。输出的靠前的（前一时刻）神经元与输入的靠后的（后一时刻）神经元相互之间产生了连接，这在实时流式信号表示中是不被允许的。同样的，普通的自注意力模块也存在类似的问题。而对于单向的时序网络，如门控线性单元或者长短期记忆网络来说，它们则很好地遵循了自回归的因果约束，所以用来进行流式信号表示是最为自然的。但如果只使用时序网络对原始的信号时频谱进行处理，提取特征所需的参数量会大大增加，这又违背了语音信号解码表示模型 RTFCRN 的轻量化原则。

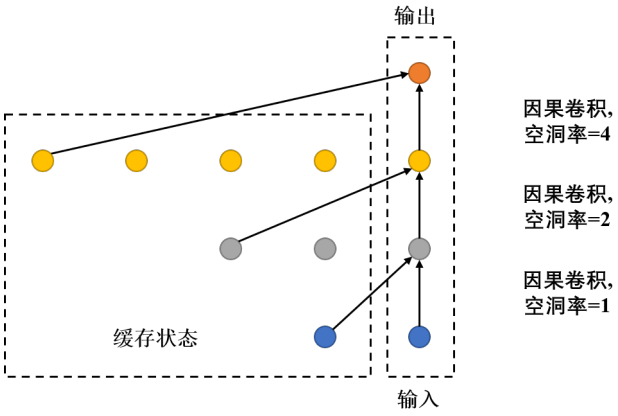

图 6-3: RTFCRN 的带缓存状态的三层因果空洞卷积模块

在本章中，我们使用因果空洞卷积模块代替普通卷积模块。在对比实验中，我们也使用了截断的自注意机制代替标准的注意机制，使得网络结构具有因果

性。因果卷积的示意图如图6-3所示。每一层的帧都仅仅与前一层过去的帧和现在的帧之间有联系，而不会有与未来的帧有联系。如果卷积核的空洞率为 d，则因果卷积需要缓存历史的 d 帧特征，以保持历史状态。缓存的帧被称为缓存状态（Cached State）。因果空洞卷积能够用较少的参数量，将信号时频谱特征进行编码并映射到合适的空间。而门控循环单元模块则已经自然地维护了自身的隐状态，我们只需要将其在 RTFCRN 结构中进行保持，就可以自然地实现实时信号的解码表示和语音信号增强。

6.2.2 编码器

图6-2中的 F-Encoder 和 T-Encoder 由如图6-4所示的卷积模块堆叠而成，用来提取信号表示。之所以用卷积作为网络编码器和解码器的主体，一是因为卷积运算本身是从局部到全局逐步提取和融合特征的算子，与语音信号短时平稳的性质有着良好的匹配度。并且信号的局部特征也是相对更重要的特征。二是因为卷积运算本身是轻量级的，它相对于全连接层有着非常少的参数量。流式运算也能够通过带缓存的卷积核较为容易地实现，所以用它来作为信号解码表示的编码器 $\mathcal{E}_\theta$ 以还原干净语音信号是比较自然的。

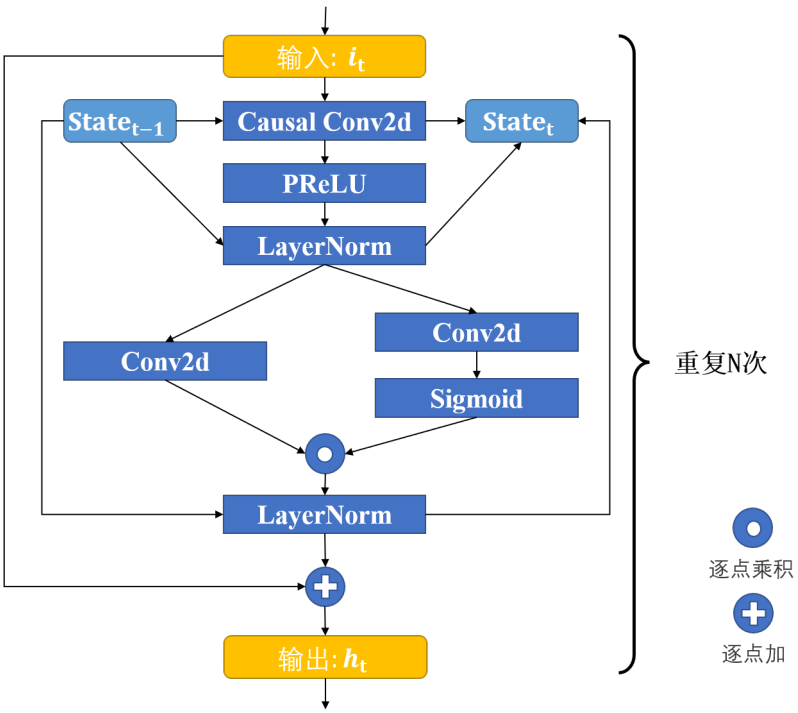

图 6-4: RTFCRN 的编码器结构图

Causal Conv2d 表示因果空洞卷积。其输入是历史状态和当前帧 $[st-1i_t]$ 的连接，其输出是卷积核提取的特征 c_t。接着用特征 c_t 对状态序列 s_{t-1} 进行更新，得到状态序列 s_t。然后使用激活函数 Parametric Rectified Linear Unit（PReLU）对输出 c_t 执行非线性变换，并获得 f_t：

$$f_t = PReLU(c_t) = \begin{cases} c_t, & c_t > 0, \\ \gamma c_t, & c_t \leq 0, \end{cases} \tag{6-2}$$

其中 γ 是可学习的参数，也是激活函数的负斜率。它通常是一个很小的值。PReLU 可以自适应地学习激活函数负值区间的参数，允许不同通道存在不一样的激活函数，能更好地适应不同通道的空间、幅度和相位特性。

状态保持的 LayerNorm 会对历史的 k 帧特征进行归一化得到 f_t'。在归一化时会使用滑动平均算法减少计算量，k 帧 LayerNorm 的滑动平均均值 m_t 和方差 δ_t 更新公式如下：

$$\begin{aligned} m_t &= m_{t-1} + \frac{f_t - f_{t-k}}{k}, \\ \delta_t^2 &= \delta_{t-1}^2 + (m_t^2 - m_{t-1}^2) + \frac{f_t^2 - f_{t-k}^2}{k-1}. \end{aligned} \tag{6-3}$$

从而 LayerNorm 被表示为：

$$f_t' = \frac{f_t - m_t}{\delta_t + \epsilon} \cdot \beta + \alpha, \tag{6-4}$$

其中 α 和 β 是模型参数，而 ϵ 是一个很小的值。

接着，结合门控结构和残差连接来更好地提取特征和稳定训练过程：

$$\sigma(x) = \frac{1}{1+e^{-x}}, \tag{6-5}$$

$$h_t = LN\big(C^{(1)}(f_t') \odot \sigma(C^{(2)}(f_t'))\big) + i_t. \tag{6-6}$$

对于语音增强任务来说，我们需要尽可能地保护时频信息在特征提取过程中不被破坏，以上两者对于实现这个功能有着重要作用。公式 6-6中的 $C^{(1)}$ 和 $C^{(2)}$ 是逐点卷积，用来融合每个时频点的信息。门控结构 $\sigma(x)$ 是 Sigmoid 函数。通

过这种方法，门控结构的卷积模块能够融合不同通道中的信息，对时频图中的各个时频点进行加权，控制不同部分的重要性。并且门控结构也能较好地缓解梯度消失的问题。残差连接也有类似的作用，它能够保留了梯度传导过程中的空间结构，缓解梯度退化问题。

两个 Encoder 都使用了上述介绍的卷积模块，并将其堆叠了 N 次。F-Encoder 和 T-Encoder 会动态地维护每一个卷积模块的时序状态的集合 Fh 和 Th。两者的不同在于，F-Encoder 的空洞卷积运用在频率维度，而 T-Encoder 的空洞卷积则运用在时间维度。两者卷积核的步长也有差异。F-Encoder 通过频率维度空洞卷积，编码了跨频带信息，在关注局部频率信息的基础上，也融合了跨频带的特征。它能够更好地提取不同频率维度间的相关性，从而构建合适的频率特征。T-Encoder 则是通过时间维度的空洞卷积融合时序信息，能够更好地从局部到全局地捕捉信号帧与帧之间的关系，更好地提取带噪信号中非平稳噪声的特征。

6.2.3 门控循环单元模块

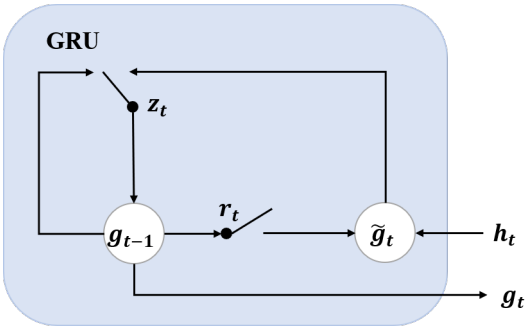

图 6-5: GRU 门控结构示意图

门控循环单元 GRU 是循环神经网络的一种，为了解决长期记忆和反向传播中的梯度等问题而被提出来。它的结构如图6-5. GRU 是轻量级的时序网络结构，它主要包括更新门 z_t 和重置门 r_t：

$$z_t = \sigma(W_z \cdot [g_{t-1}, h_t]) \tag{6-7}$$

$$r_t = \sigma(W_r \cdot [g_{t-1}, h_t]). \tag{6-8}$$

重置门决定了如何将新的输入信息与前面的记忆相结合，更新门决定要将多少

过去的状态信息传递到未来，并用于更新状态 g_t：

$$\widetilde{g}_t = tanh(W \cdot [r_t \cdot g_{t-1}, h_t]) \tag{6-9}$$

$$g_t = (1 - z_t) \cdot g_{t-1} + z_t \cdot \widetilde{g}_t. \tag{6-10}$$

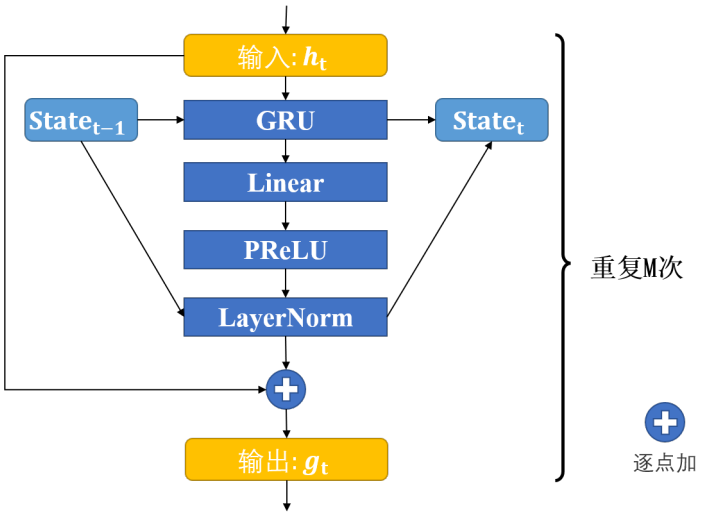

图 6-6: RTFCRN 的门控循环单元模块

相比于 LSTM，GRU 的参数量更小，但在语音增强上的性能表现通常与 LSTM 相当。之所以使用循环神经网络而不是常用的基于相对位置编码的自注意力，是因为我们发现 RNN 相对于注意力结构，能够更加自然地维护帧与帧之间的状态，保持序列的因果性。在同样的参数量下，GRU 对于降噪任务有着更好的效果。在我们的实验中也同样证明了这个结果。门控循环单元模块如图6-6所示。它包括一个单向的 GRU，作为生成信号概率掩膜 cIRM 特征表示的函数 f_θ，用来从信号表示中维护和提取时序信息。接着将 g_t 做非线性变换，并同样进行归一化和残差连接，得到待处理和解码的信号分量表示掩膜。为了更好地处理信号表示，这个门控循环单元模块会重复 M 次。

6.2.4 解码器

与编码器类似，RTFCRN 的解码器也是由如图6-7所示的卷积模块堆叠 N 次得到的。编码器和解码器的整体上的设置类似于 U-Net。假设当前模块是解码器中的第 i 个模块，其输入由两部分组成，第 $i-1$ 个解码器模块的输出 g_t 和 $N-i$ 编码器模块的输出 h_t。解码器的目的是在最小化语音频谱信息损失的前提

下，正确预测语音的时频点概率，并获得干净语音的复数域掩码。

对于语音增强任务，需要预测出每一个时频点的干净语音概率，所以需要将特征映射回原始的时频图大小。首先使用转置卷积处理门控线性单元提取的隐变量，逐步增加其时频分辨率。转置卷积也常常被称为反卷积。它具有可以学习的参数，相对于池化和插值等方法，是一种更精细化的上采样方法。

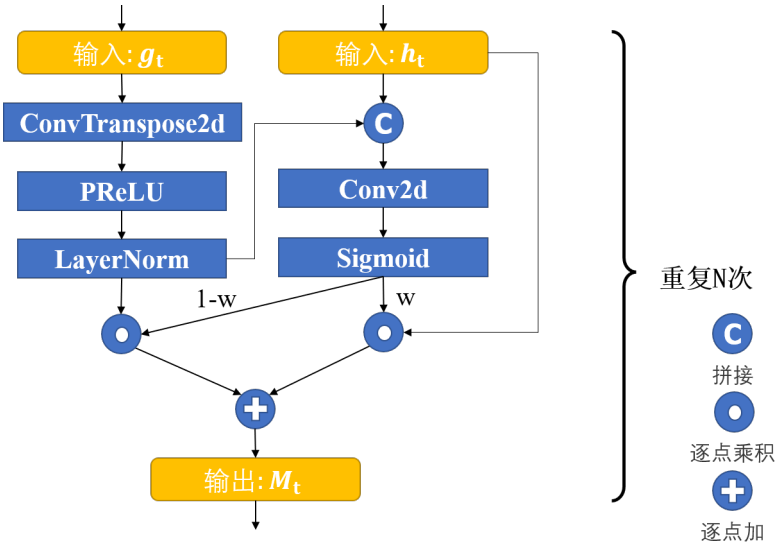

图 6-7: RTFCRN 的解码器结构图

接着，我们将上采样、非线性激活和归一化后的特征与 T-Encoder 输出的相同分辨率的原始特征连接起来，并计算每个通道和时频点的权重 w。最后，将编码器信号特征和解码器掩膜特征按权重 w 加权，以控制原始编码信号表示 h_t 和门控循环单元模块处理的掩码特征 g_t 之间的重要性，以便在保持频谱分辨率的同时使解码的掩膜 M_t 预测得更加准确。

6.3 训练目标和感知损失

6.3.1 预测目标

传统的信号表示和增强系统只增强信号的幅度谱图。这些方法一般会使用噪声相位谱与干净的幅度谱重新变换合成时域信号波形。它们的训练目标一般是理想二值掩膜, 理想比率掩膜或者目标幅度谱等等。相比于直接从连续信号表示中预测干净语音的目标幅度谱，预测 IRM 相对容易。最近的研究发现，在信噪比较低的情况下，相位的预测对于语音信号增强系统的性能影响很大。附带

有相位信息的 cIRM 比 IRM 有着更好的感知质量，对于语音信号的客观可感度的改善有比较显著的影响。所以，RTFCRN 的预测目标是干净语音信号的复数比率掩膜 $M \in \mathbb{C}^{F\times T}$。给定干净语音复数谱 S 和带噪语音复数谱 X，M 可以表示为：

$$M = \frac{X_{r,1}S_r + X_{i,1}S_i}{X_r^2 + X_i^2} + j\frac{X_{r,1}S_i - X_{i,1}S_r}{X_r^2 + X_i^2}, \tag{6-11}$$

其中 $X_{r,m}$ 和 $X_{i,m}$ 分别是带噪语音信号时频谱 X 的第一个通道的实部和虚部，S_r 和 S_i 分别是干净语音信号时频谱 S 的实部和虚部。因此，可以通过 M 来估计干净的语音信号：

$$\hat{S} = X_{r,1}M_r + X_{i,1}M_i + j(X_{r,1}M_i + X_{i,1}M_r). \tag{6-12}$$

在训练过程中，我们将 cIRM 的值压缩到 $-K$ 和 K 之间，在测试过程中则将其解压缩。这在一定程度上能够降低连续信号解码表示模型的预测难度，提高 RTFCRN 的收敛速度。

6.3.2 人耳听感客观评价指标

通常我们会从客观和主观两个角度衡量语音信号增强的效果。过去常用的客观指标包括信噪比、分段信噪比和倒谱距离等等，主要从传统信道角度评价信号质量，但这些指标不能很好体现出人的主观评价。主观评价通常通过人工打分的方法进行，但会耗费较多资源。在语音信号增强任务中，我们最关心的是语音信号在人耳听感方面的降噪效果，所以本节会介绍两种常用的客观评价指标：短时客观可懂度和感知语音质量评价，它们与人耳主观评价指标之间的相关性较高。

短时客观可懂度（Short Time Objective Intelligibility, STOI）[115] 反映了人类的听觉感知系统对语音信号可懂度的客观评价，STOI 的值介于 0 到 1 之间，值越大代表语音信号的可懂度越高，越清晰。STOI 的主要思想是计算增强信号与参考信号之间的自相关。STOI 首先会把信号重采样到 10000Hz，以覆盖语音可感范围。在正式计算 STOI 前会将静音段去除。接着将两个信号分割成 50% 重叠的汉宁加窗后的帧（长度为 256 个样本），以获得其时频表示。其中每个帧被零填充至 512 个样本，并进行傅里叶变换。然后，通过分组离散傅里叶变化单元进行三分之一倍频程分析。STOI 计算中总共使用了 15 个三分之一倍频程频带，

其中最低中心频率设置为 150Hz。接着计算每个三分之一倍频程频带的限制性信号比。其中带噪语音信号的时频谱经过归一化因子进行缩放，使得其频谱能力与干净语音信号一致。为了限制计算中不出现过于异常的信噪比影响整体性能，会将每个时频单元进行裁剪，之后利用归一化和裁切过的时频点计算每个三分之一倍频程和时频单元的线性相关系数。最终的可懂度度量由这些频带和帧的平均得到。

语音质量感知评估 (Perceptual Evaluation of Speech Quality, PESQ) 是窄带电话网络和语音编解码器的端到端语音质量评估的客观方法[116]。真实的系统可能会包括滤波和可变延迟，以及由于信道误差和低比特率编解码器引起的失真。PESQ 通过传递函数均衡、时间校准和新的时间平均失真算法来解决这些影响。窄带 PESQ 用于 3.1kHz(窄带, 8000Hz 采样率) 手机电话和窄带语音编解码器的语音质量评估。而在 2007 年，国际电联公布了 PESQ 的宽带版本 (ITU-T P862.2，PESQ-WB)，用于评估宽带电话网络和语音编解码器，主要用于宽带音频系统 (50 到 7000Hz, 16000Hz 采样率)。PESQ 属于客观评价，但它和主观评价分数之间的相关性约为 0.935。其取值在 −0.5 到 4.5 的范围内，在大多数情况下输出范围在 1.0 到 4.5 之间，得分越高表示语音质量越好。PESQ 的主要思想是将频谱信号转换到巴克谱域计算距离。

6.3.3 损失函数

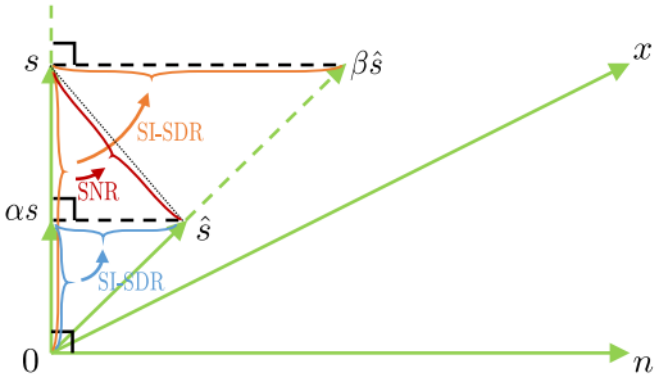

图 6-8: Si-SDR 的两种计算方式

在训练时，其中一项损失函数为尺度不变信噪比（Scale Invariant Signal-to-Distortion Ratio , Si-SDR），它主要衡量的是降噪算法的信噪比提升。SDR 的计

算有一定的缺陷：只需要改变参考信号的幅度，而不需要对估计信号做出任何改变，SDR 就可以得到提升，主要原因在于残差信号与参考信号不是正交的。Si-SDR 正是为了改进这一问题而提出的。如图6-8[117] 所示，为了确保残差确实与参考信号正交，我们可以重新调整参考信号 s 或重新调整估计信号 $\hat{s}$。重新缩放参考信号可以在参考信号的延长线上找到估计信号的正交投影，或者等效地找到沿着该线的最接近估计信号的点[117]。我们通过短时傅里叶逆变换将 $\hat{S}$ 变换为时域波形 $\hat{s}$，并计算 Si-SDR：

$$\alpha = \frac{\hat{s}^T s}{\|s\|^2}, \tag{6-13}$$

$$\mathcal{L}_{Si-SDR} = -log\left(\frac{\|\alpha s\|^2}{\|\alpha s - \hat{s}\|^2}\right). \tag{6-14}$$

连续信号解码表示任务通常希望将初始信号转变为目标信号，所以一般都使用均方误差来作为损失函数。但 MSE 通常假定信号是平稳的，即其频谱特性不会随时间变化。在语音信号增强中，MSE 很难跟踪非平稳噪声，会导致语音信号表示过分平滑，并导致语音信号的清晰度下降。这是因为 MSE 同等地对待被评估的所有时频点。而相比于 MSE 损失, Si-SDR 损失能够显著提升信号增强的效果。另一方面，为了衡量人耳听感效果，STOI 也会被作为损失函数的一部分，使得训练目标与任务目标更加适配。最终的损失函数表示为：

$$\mathcal{L}_e = \mathcal{L}_{Si-SDR} + \lambda_1 \mathcal{L}_{STOI}, \tag{6-15}$$

其中 λ_1 是用于平衡 Si-SDR 和 STOI 的超参数。在实际任务中，还可能在信号增强损失 $\mathcal{L}_e$ 中加入针对说话人约束的损失函数，即预测解码后的干净信号表示 $\hat{S}$ 的说话人 ID，以识别和提取说话人身份，辅助信号增强。

6.4 模型的轻量化

模型轻量化包括许多种不同种类的算法。参数剪枝和共享方法希望从深度神经网络中移除无关紧要的参数，而不对性能产生显著影响。它又可以进一步分为模型量化、模型二值化、结构矩阵和参数共享等。低秩分解方法通过使用矩

阵和张量分解来识别深度神经网络的冗余参数。压缩卷积过滤器方法通过转移或压缩卷积过滤器来去除不重要的参数。知识蒸馏则专注于将知识从更大的深度神经网络中提取到一个小型网络中。

模型压缩和加速四个技术是：设计高效小型网络、剪枝、量化和蒸馏。之前大部分的信号增强研究都只关注于如何设计适合的少参数量小网络，并使得语音信号增强的性能尽可能的好。本节将会在小型连续信号解码表示模型 RTFCRN 上设计和使用敏感度剪枝算法和基于特征的知识蒸馏算法进行模型压缩，并在实验中与其它方法对比效果。最后我们还会对性能最好的模型进行 8 比特动态量化。

6.4.1 基于敏感度的剪枝算法

模型剪枝方法分为非结构化剪枝和结构化剪枝。本文使用 Inf-CP 算法，通过影响函数估计模型权重的敏感性，并进行结构化迭代剪枝[118]。影响函数测量可用于评估模型权重的重要性。将一较小的扰动添加到模型的某个权重中以计算模型的损失函数变化，然后通过对影响函数的定义和推导，得出权重参数对模型的影响。通过计算权重影响，从而确定模型的参数删除策略。

用 T_l 来表示部分神经元连接被删除的稀疏向量。每个 T_l 都是一个二进制向量，其条目指示层第 l 网络通道的连接状态，即它们当前是否被修剪。T_l 可以被视为网络层通道的掩码向量。我们添加一个与参数矩阵大小相同的掩码滤波器，并在通道中将其初始化为常量 1，然后将参数矩阵和掩码滤波器以哈达玛积的方式相乘。通过反向传播使用当前小批量样本作为输入计算掩码滤波器的梯度，得到整个通道和通道中单个权重的影响力。一方面，选择删去通道影响力小于阈值所对应的通道，生成通道目标剪枝策略向量。使用二进制向量的每一个值表示对应通道是否被剪枝，如果值为 1 则保留该通道。另一方面，我们将模型当前层的所有通道中的单个权重影响力通过一层卷积层得到当前的实际通道影响力。然后将这个中间转向量通过 Sigmoid 函数作二值化处理，以获得通道剪枝策略向量 E_l。

在训练过程中逐层进行剪枝，并设计损失函数，让模型除了关注连续信号表示和增强损失之外，还要关注模型的剪枝策略。计算目标编码向量 T_l 和经过

处理的剪枝策略向量 E_l 的欧式距离，将所得值作为 RTFCRN 的正则项：

$$\mathcal{L}_p = \mathcal{L}_e + \lambda_2 \|E_l - T_l\|_2^2, \tag{6-16}$$

其中 λ_2 是用于平衡信号增强损失和剪枝损失的超参数。

6.4.2 基于特征的蒸馏算法

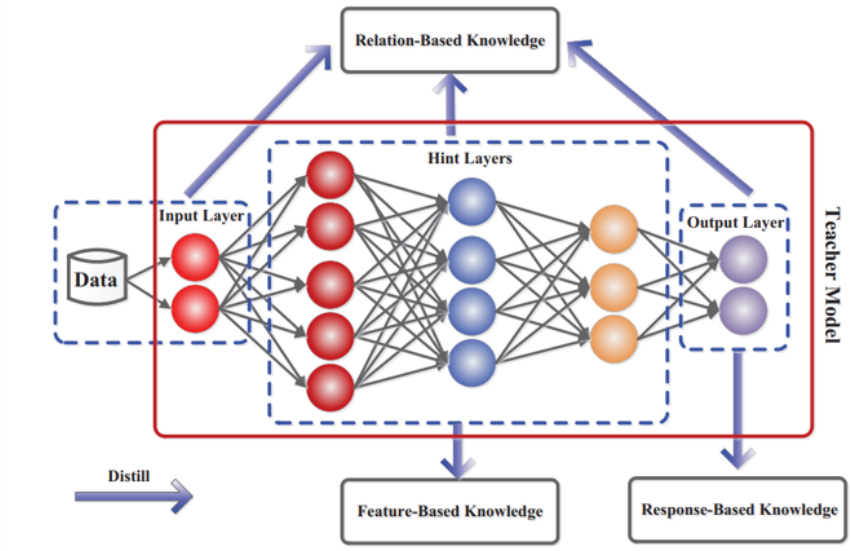

图 6-9: 知识蒸馏的三种类型

知识蒸馏是进行模型轻量化的一种重要方法，如图6-9[119] 所示，它可以分为为三种类型：基于响应的模型蒸馏、基于特征的模型蒸馏和基于关系的模型蒸馏[119]。为了将原始的 RTFCRN（教师网络）的表示移植到轻量级的学生网络上，好的训练目标不是拟合有限的连续信号，而是学习如何泛化到新的连续信号表示上。所以知识蒸馏的目标是让学生网络学习到教师网络的泛化能力和连续信号表示。在减少网络参数量的基础上，学生网络在连续信号解码表示上表现得不亚于教师网络。

我们使用 Overhaul 算法对 RTFCRN 进行基于特征的模型蒸馏。蒸馏的框架图也被表示在了图6-2中, 图的左边为原始的 RTFCRN（教师网络），右边为轻量级的 RTFCRN(学生网络)。Overhaul 是一种强大的基于特征的模型蒸馏方法，进一步提高了特征蒸馏的性能，它是考虑了蒸馏的各个方面而设计的，包括：教师变换、学生变换、蒸馏特征位置和距离函数。Overhaul 的设计理念是区分有益

信息和不利信息，以提高特征蒸馏性能。蒸馏损失被设计为仅将有益的教师信息传递给学生[120]。学生网络的结构与教师网络相似，不同的是通过减少通道数或隐藏层维度的方式，减小了 F-Encoder，T-Encoder 和 GRU 模块的参数量，得到了学生网络的 LF-Encoder，LT-Encoder 和 LGRU 模块. 因为网络参数量主要集中于 GRU 模块的线性层 (Gh_t) 和两组编码器的最后一个卷积模块中 ($Fh_{t,N}$ 和 $Th_{t,N}$)。

定义教师网络的蒸馏特征集合或状态集合为 $T = \{Fh_{t,N}, Th_{t,N}, Gh_t\}$, 学生网络是蒸馏特征集合或状态集合为：$S = \{Fh'_{t,N}, Th'_{t,N}, Gh'_t\}$。知识蒸馏算法的关键是构建适当的教师变换函数、学生变换函数和距离度量函数 $d(T,S)$。教师变换函数是将教师的隐空间特征转化为易于蒸馏的形式。在语音增强任务预测语音信号掩膜的过程中，假设每个模块的激活函数 PReLU 只允许语音信号的特征通过，并阻止噪声信号的特征通过，从而过滤掉负面信息。那么为了获得更好的连续信号表示效果并在 RTFCRN 中保留所需信号的信息，可以选择对 PReLU 之前的特征进行蒸馏。阈值 ReLU 函数被用作教师变换函数。阈值 ReLU 函数 σ_m 被表示为：

$$m = E[T_i | T_i < 0, T_i \in T], \tag{6-17}$$

$$\sigma_m(T) = max(T, m), \tag{6-18}$$

其中 m 是通道负响应的期望，也是阈值 ReLU 中的阈值，而 T_i 是教师隐藏状态集合 T 中的第 i 个状态。为了使学生网络的维度与教师网络相匹配，我们使用 1×1 卷积 c 作为学生变换函数。学生变换增加了学生网络特征通道的数量，以便于计算与教师网络特征的距离。这样，教师网络信息的重要部分可以被迁移到学生网络中。距离函数定义如下：

$$d(T,S) = \sum_i \begin{cases} 0, & if \quad S_i \le T_i \le 0, \\ (T_i - S_i)^2, & otherwise, \end{cases} \tag{6-19}$$

教师网络的正面响应（$T_i > 0$）会被学生网络学习，而负面响应（$T_i < 0$）则不一定会。只需确保学生的负面响应（$S_i < 0$）足够小，小于对应的教师网络的

负面响应即可。因此，基于特征的模型蒸馏损失定义为：

$$\mathcal{L}_{distill} = d(\sigma_m(T), c(S)), \tag{6-20}$$

$$\mathcal{L}_d = \mathcal{L}_e + \lambda_3 \mathcal{L}_{distill}. \tag{6-21}$$

其中 λ_3 是用于平衡信号增强损失和蒸馏损失的超参数。$\mathcal{L}_{distill}$ 表示蒸馏损失函数，$\mathcal{L}_d$ 则表示最终用于学生网络蒸馏训练的损失函数。

6.4.3 参数动态量化

由于 GRU 模块的参数量占 RTFCRN 的 50% 以上，所以需要对其进行量化。模型量化可以保证更少的存储开销和带宽需求。通过量化技术能够显著地减少参数的内存占用，这是因为该技术一般会把模型参数从 32 位的位浮点数量化为 8 位的整型数，从而缩小 75% 的存储空间。这对于计算资源有限嵌入式设备和日常中常见的边缘设备的深度学习模型部署和使用有极大的帮助。一方面，读取 32 位的浮点数需要的带宽能同时读入四个 8 位整型数。另一方面，整型运算相比浮点型运算更快。所以量化技术还能加快模型的运算和推理速度，从而降低设备能耗。

用 R 表示真实的浮点值，Q 表示量化后的定点值，Z 表示 0 浮点值对应的量化定点值，C 则为定点量化后可表示的最小刻度。由浮点到定点的量化公式如下：

$$Q = \frac{R}{C} + Z. \tag{6-22}$$

其中，C 和 Z 的求值公式如下：

$$C = \frac{R_{max} - R_{min}}{Q_{max} - Q_{min}}, \tag{6-23}$$

$$Z = Q_{max} - \frac{R_{max}}{C}. \tag{6-24}$$

$R_{max}, R_{min}, Q_{max}, Q_{min}$ 分别表示最大和最小的浮点数，最大和最小的定点数。如果量化后的 Q 或者反推求得的浮点值 R 超出各自可表示的最大范围，则需进行截断处理。蒸馏后 GRU 模块的线性层的做了动态量化：只量化了模型权重，而

它的激活函数会在推理过程中动态地量化。

6.5 语音信号解码表示理论

同样的，在传统连续信号解码表示基础上更进一步，能够基于深度学习连续信号解码表示形式，构建出语音增强任务上的优化问题形式：

$$\min \quad \mathcal{L}_e(\mathcal{D}_\theta(\mathcal{G}_\theta(h), x)), s) \tag{6-25}$$

$$s.t. \quad h = \mathcal{E}_\theta(x)$$

$$N(\theta) \leq k, \tag{6-26}$$

$$\mathcal{L}_{speaker}(g(h, x), y_s) \leq \delta.$$

其中 $\mathcal{E}_\theta$ 和 $\mathcal{D}_\theta$ 依然分别表示编码器和解码器，而 $\mathcal{G}_\theta$ 则表示门控线性单元模块。h 实际上是带噪语音信号的整体表示，$\mathcal{G}_\theta(h)$ 则是干净语音信号的分量表示。而语音信号 x 经过 RTFCRN 的处理后，与目标干净语音信号 s 计算信号增强损失 $\mathcal{L}_e$。而对模型的轻量化则体现在约束 $N(\theta) \leq k$ 中，主要通过模型剪枝或者蒸馏来实现。任务中通常还会加入说话人约束 $L_{speaker}$，在语音信号增强中融入说话人信息，该约束通常为交叉熵函数。另外，g 表示基于信号表示 h 与原信号 x，提取说话人表示的函数。y_s 则表示语音信号 x 对应的说话人标签。

在基于深度学习编解码理论的连续语音信号解码表示中，我们基于神经网络用 RTFCRN 更好地建模了语音信号表示，能够解决相对于传统方法更加复杂的问题。但也同样因为深度学习的原因，需要采取措施对网络参数量 $N(\theta)$ 进行约束，增加算法实时性，减少存储量，而这正是模型压缩算法所发挥的作用。

6.6 实验和分析

6.6.1 数据集

本章的连续信号解码表示研究面向的是中文场景下的语音信号增强任务，所以会使用开源中文语音数据集和噪声数据集训练和测试 RTFCRN，并将它与其它的语音信号增强算法进行对比。语音信号数据集主要包括 THCHS-30，

Primeword, MAGICDATA 和 Aishell 中的中文语音数据，噪声数据集则主要包括 Wham Noise, MUSAN 和 DNS 中的噪声数据。如果原始数据集已经划分了训练集、验证集和测试集，那么我们直接使用划分好的数据集；否则我们按照 7：2：1 比例随机划分数据集。语音数据的训练集、验证集和测试集分别包括 231565，37094 和 38989 段信号。经过数据增强的噪声训练集、验证集和测试集分别包括 104541，29868 和 38990 段信号。

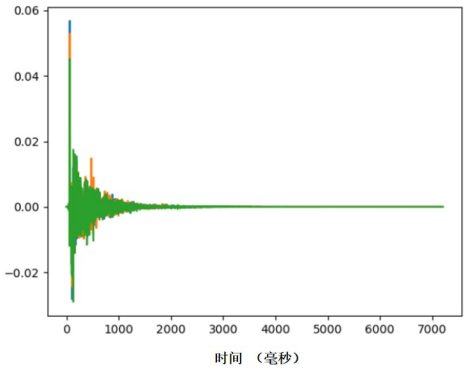

图 6-10: 镜像反射法生成的三通道房间冲激响应

在训练时，我们使用动态混合的方法来生成多通道带噪混响语音信号数据。具体来说，每次会从训练集中随机采样一段语音和一段噪声，并分别进行频率域掩模、时域掩模、速度变换和幅度裁剪等数据增强变换。之后依据随机采样的房间参数，t60, 麦克风位置和声源位置等，用镜像反射法生成房间冲激响应 (Room Impulse Response, RIR)。生成的三通道 RIR 如图6-10所示。用不同麦克风位置的 RIR 对原始语音和噪声进行卷积，构造多通道混响语音和噪声。最后将语音和噪声按照 −5dB 到 25dB 之间的随机信噪比进行混合，得到多通道混响带噪语音信号。

6.6.2 实验设置

在语音信号增强实验中，所有语音和噪声信号都被重采样到 16kHz。对于 STFT，汉宁窗窗长为 25 毫秒，窗口跳跃长度为 10 毫秒，傅里叶变换点数为 512。实验面向三通道混响语音增强任务，第一个通道为参考通道。STFT 之后，模型的输入是三个振幅谱和两个相位差谱。优化器是 Adam，初始学习率是 3×10^{-4}。当验证集的损失值连续两个时间段没有减少时，学习率将减少一半。为了保持

训练的稳定性，将对范数大于 5 的梯度进行梯度剪裁。损失函数 $\mathcal{L}_e$ 中 STOI 的权重 λ_1 为 0.7。为了实验比较的公平性，实验中使用的所有网络结构都将被转换为因果结构，并且它们在压缩之前的网络参数量将保持接近 5M。每个算法都在我们构建的数据集上进行了充分的训练和测试。在训练过程中，每段长为 40 毫秒的带噪语音信号片段被作为一个输入的信号块。在 RTFCRN 中会维护状态 Fh_t，Th_t 和 Gh_t。在其他用于对比的网络结构中也会维护类似的状态以供比较。在测试过程中，我们对三通道带噪语音进行流式降噪，每次推理的信号输入长度也为 40 毫秒。实验中使用的语音信号增强算法的具体设置如下：

- RTFCRN：F-Encoder 由 $N = 3$ 个卷积模块组成。空洞卷积率分别为 $\{1, 2, 4\}$，因果卷积核大小为 $(3, 5)$。T-Encoder 包括 $N = 4$ 个卷积模块，空洞卷积率为 $\{1, 2, 4, 8\}$，因果卷积核大小为 $(5, 3)$，最终卷积层的通道数为 128。门控循环单元模块包含 $M = 2$ 层 GRU，其隐藏层维度为 512。
- FullSubNet[79]：除了需要维护 LSTM 的状态之外，其他网络配置与原始的结构完全一致。
- SA-TCN[81]：LTCN 堆栈的数量 $R = 2$，堆栈中 TCN 的数量 $L = 8$，卷积核的输入通道 $H = 128$，输出通道 $B = 64$，卷积核大小 $P = 3$。原始网络中的非因果结构被因果结构取代，其他配置与原始结构配置相同。
- T-GSA[80]：共包含 10 个因果的高斯自注意力块，其中线性层维度为 1024。实验中增加了额外的卷积层来处理多通道问题，其他配置与原始结构一致。
- GeneralBeamformer[84]：RNN-BF 包括两个 GRU，为了保持参数量基本一致，每个 GRU 的隐藏层维度为 300，非线性 DNN 的维度也是 300。模型输出是复数时频谱。所有模块都被改为带缓存的实时模块，其他配置与原始结构一致。
- HiFi-GAN[82]: HiFi-GAN：生成器的网络结构与本章中介绍的 RTFCRN 接近，鉴别器的网络结构和训练配置与原始结构完全一致。

实验中主要面向人耳听感场景进行与说话人无关的多通道实时语音信号增强，各个用于对比的深度学习网络的降噪性能通过 SNR，STOI 和 PESQ 等客观评价指标进行评估，轻量化效果则由压缩前后的参数量大小，压缩率和算法的实时率体现。

6.6.3 消融实验

为了证明 RTFCRN 信号解码表示算法的各个模块对于连续信号解码表示的有效性，我们在本节中针对算法的关键组件做了消融实验，结果如表6-1所示。表中前两行分别列出了原始带噪信号和经过 RTFCRN 算法进行语音增强后信号的评估结果。表中的五到七行则分别展示了从 RTFCRN 算法中去掉相应组件并且保持其它组件不变的情况下，算法对语音信号增强的效果。

表 6-1: RTFCRN 算法消融实验表

方法	SDR(dB)	STOI	PESQ
原始信号	15.12	0.9013	2.0987
RTFCRN	**20.47**	**0.9243**	**2.6941**
w.o. 因果卷积	19.05	0.9141	2.4916
w.o. 空间信息	19.29	0.9159	2.4702
w.o. F-Encoder	20.17	0.9219	2.6528
w.o. 门控结构	20.19	0.9221	2.6285
w.o. STOI 损失	20.31	0.9136	2.6071

从表6-1中我们发现，无论缺少哪个组件，RTFCRN 算法对于连续信号的解码表示性能都会受到不同程度的影响。其中，编码器因果卷积中保留的缓存状态对于语音增强任务的影响最显著。在只使用普通卷积时，增强信号相对于原始信号的 SDR，STOI 和 PESQ 降噪性能改善程度分别有 26.54%, 44.35% 和 34.01% 的衰减。输入的相位角差空间信息同样也是影响 RTFCRN 算法性能的重要因素，特别是对于 PESQ 指标，缺失空间信息时降噪性能改善程度有 37.60% 的衰减，是所有组件中对于 PESQ 影响最大的。如果不在损失函数中加入 STOI，而只使用 Si-SNR, 对于测试结果中的信噪比没有显著影响，但 STOI 和 PESQ 的改善有比较显著的衰减。另外，F-Encoder 和卷积模块中的门控结构对于连续信号解码表示也有一定程度的影响。

所以，我们在本章中提出的缓存状态有利于处理和保持流式信号的状态，相位角差和可选择的后置滤波器也能够融合信号中的空间信息，是改善 RTFCRN 算法表示性能的最重要的组成部分。F-Encoder 进一步编码时域信息，提取信号跨频带结构，而门控结构保留了所需信号的特征，去除了噪声信号的特征，对连

续信号解码表示有一定程度的积极影响。Si-SNR 和 STOI 损失的选择和配合则显著改善了 RTFCRN 算法在语音增强任务上的人耳听感效果。本节中我们没有使用说话人信息对语音增强任务进行约束和监督，否则 RTFCRN 处理的语音信号期望能够有更高的信噪比和客观感知指标。

6.6.4 对比实验

表6-2中展示了 RTFCRN 与六种对比的语音增强模型在开源中文语音测试数据集上的评估结果。其中“+Beamformer”表示的是首先利用 FullSubNet 算法得到干净语音信号掩膜，再接后置 MVDR 滤波器进行空间滤波的算法。连续语音信号表示算法通常不仅仅需要关注表示效果，也需要关注算法的参数量和速度。本节中通过 SDR，STOI 和 PESQ 三个客观评价指标评估七种算法的语音增强性能，用参数量衡量算法模型的占用的空间大小，并用实时率衡量算法在 CPU 上的运算速度。七种模型均能够实现实时流式的多通道语音增强。

表 6-2: 语音信号解码表示算法对比实验表

语音信号解码表示算法	SDR(dB)	STOI	PESQ	参数量 (M)	实时率
SA-TCN	16.22	0.9021	2.1872	**2.91**	**0.21**
T-GSA	18.74	0.9080	2.4307	4.47	0.45
HiFi-GAN	17.81	0.9065	2.4547	7.92	0.59
GeneralBeamformer	19.97	0.9201	2.6208	5.56	3.13
FullSubNet	20.43	0.9213	2.5136	6.58	0.85
+ Beamformer	18.61	**0.9276**	2.4969	6.58	0.87
RTFCRN	**20.47**	0.9243	**2.6941**	6.16	0.31

从表中发现，所有语音增强算法处理得到的信号相对于原始的带噪信号均有所提升。RTFCRN 算法在 SDR 和 PESQ 上均优于其它六种算法，STOI 与 FullSubNet+Beamforer 算法接近，并优于其它五种算法。GeneralBeamforer 算法的 PESQ 同样较高，但由于需要直接预测滤波相关性矩阵，该算法比较难以实现实时语音增强。六种对比的算法中，FullSubNet 的综合降噪性能和最好，但仍与本章介绍的 RTFCRN 算法有一定差距。SA-TCN 算法的参数量和实时率是包括 RTFCRN 的所有算法中最优的，是一个非常轻量级的算法，但相对的，它的

语音增强性能没有其它算法好。从对比实验中能够发现，RTFCRN 算法在同等参数量和速度的情况下，有着十分良好的降噪性能，特别是在人耳主观的感知上表现优秀。得益于缓存结构和因果卷积的设计，RTFCRN 也是能实现流式信号解码表示的轻量级的模型。

6.6.5 模型轻量化实验

表 6-3: 轻量化算法对比实验表

轻量化算法	SDR(dB)	STOI	PESQ	参数量 (M)	压缩率	实时率
FullSubNet origin	20.43	0.9213	2.5136	6.58	-	0.85
FullSubNet	19.52	0.9156	2.4451	1.39	78.88%	0.73
SA-TCN origin	16.22	0.9021	2.1872	2.91	-	0.21
SA-TCN	16.39	0.9084	2.1858	0.76	73.89%	0.15
RTFCRN origin	20.47	0.9243	2.6941	6.16	-	0.31
RTFCRN light	20.43	0.9242	2.6834	0.81	86.86%	0.21
RTFCRN	20.43	0.9267	2.7341	0.81	86.86%	0.21

为了说明 Overhaul 特征蒸馏算法在 RTFCRN 模型压缩上的有效性，我们还对综合性能最优的 FullSubNet 和最轻量的 SA-TCN 分别进行了剪枝和蒸馏，并与经过模型蒸馏的 RTFCRN 算法进行了对比。表6-3中展示了轻量化实验的结果，其中 “origin” 表示未压缩的模型，“light” 表示不经过蒸馏而是直接训练的参数量较少的模型，其它表示经过模型压缩算法处理后得到的模型。

FullSubNet 经过剪枝后性能虽有一定的下降，但其参数的压缩率很高，CPU 上的实时率也有一定提升。SA-TCN 经过蒸馏后参数量在 1M 以下，性能也略有提升。RTFCRN 算法经过特征蒸馏后，STOI 和 PESQ 均有所提升，分别提升了 0.26% 和 1.48%。另一方面，模型的参数量压缩了 86%，达到了 0.81M, 与 SA-TCN 不相上下。为了证明是特征蒸馏算法起到了作用，而不是轻量级网络带来的性能提升，我们还直接训练了 0.81M 的小参数量网络的 RTFCRN light，其性能相对于 RTFCRN origin 有一定的下降。

图6-11则更具体地展示了原始的 RTFCRN 和轻量化的 RTFCRN 在各个信

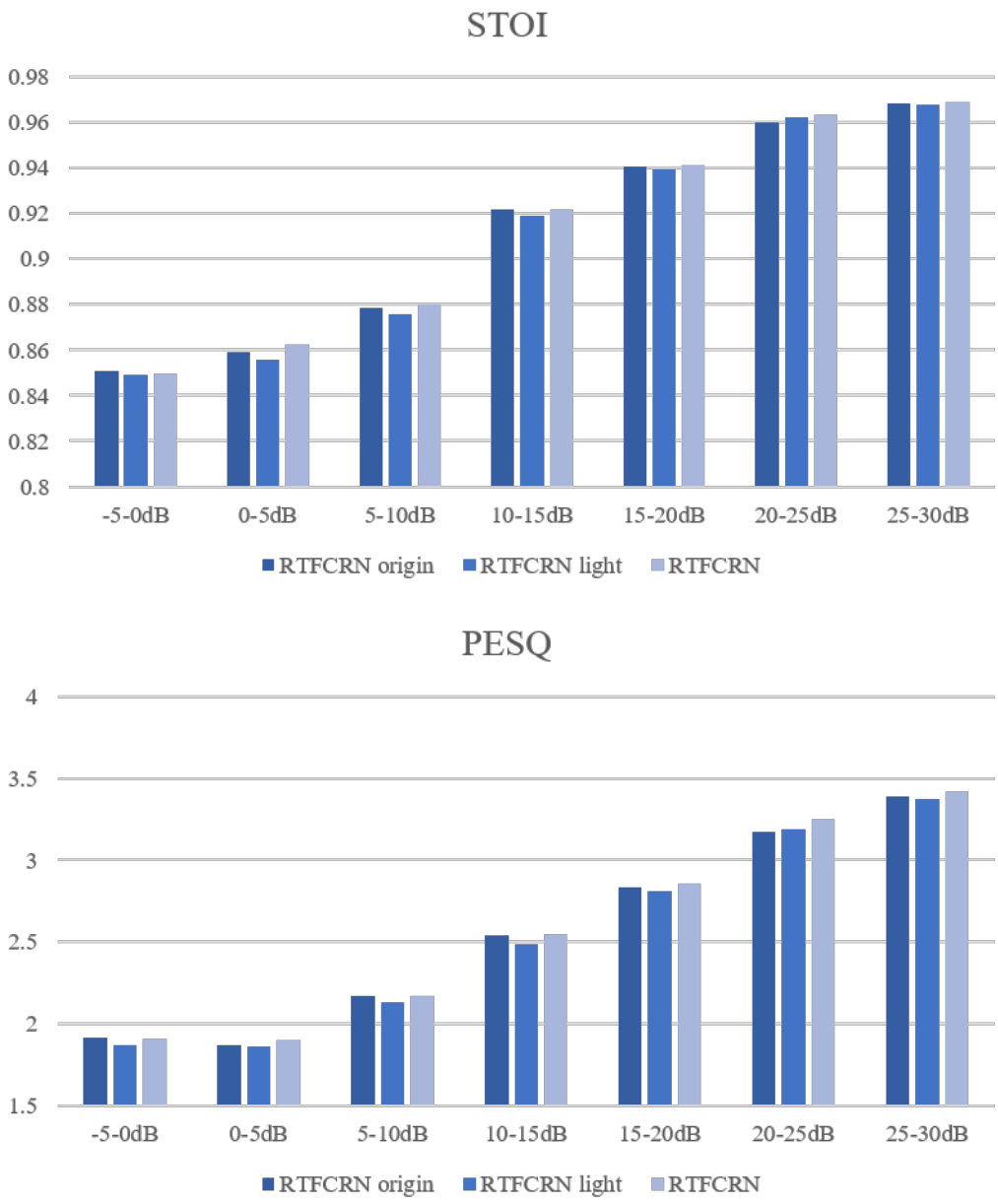

图 6-11: 原始 RTFCRN，直接轻量化的 RTFCRN 和蒸馏得到的 RTFCRN 对各个信噪比范围内的带噪语音信号的增强效果

噪比范围内的 STOI 和 PESQ 的性能对比。能够发现在 0 – 30dB 的信噪比范围内，模型蒸馏得的 RTFCRN 均比原始 RTFCRN 和直接训练的轻量化 RTFCRN 在听感上表现更好。虽然在 −5 – 0dB 信噪比范围内，语音增强的性能有轻微的损失，但相比 RTFCRN light 依然有较明显的提升。整体上看，基于特征的模型蒸馏算法在减少 RTFCRN 算法模型参数量的基础上，依然能保持原有连续信号解码表示在各个信噪比环境下的性能，甚至使其性能有所提升。这验证了 Overhaul 模型蒸馏算法在语音信号增强任务上的有效性。经过模型压缩后，RTFCRN 依然

是一个在连续信号解码表示上具有良好性能的实时信号处理模型，具有参数量小、速度快的优势。

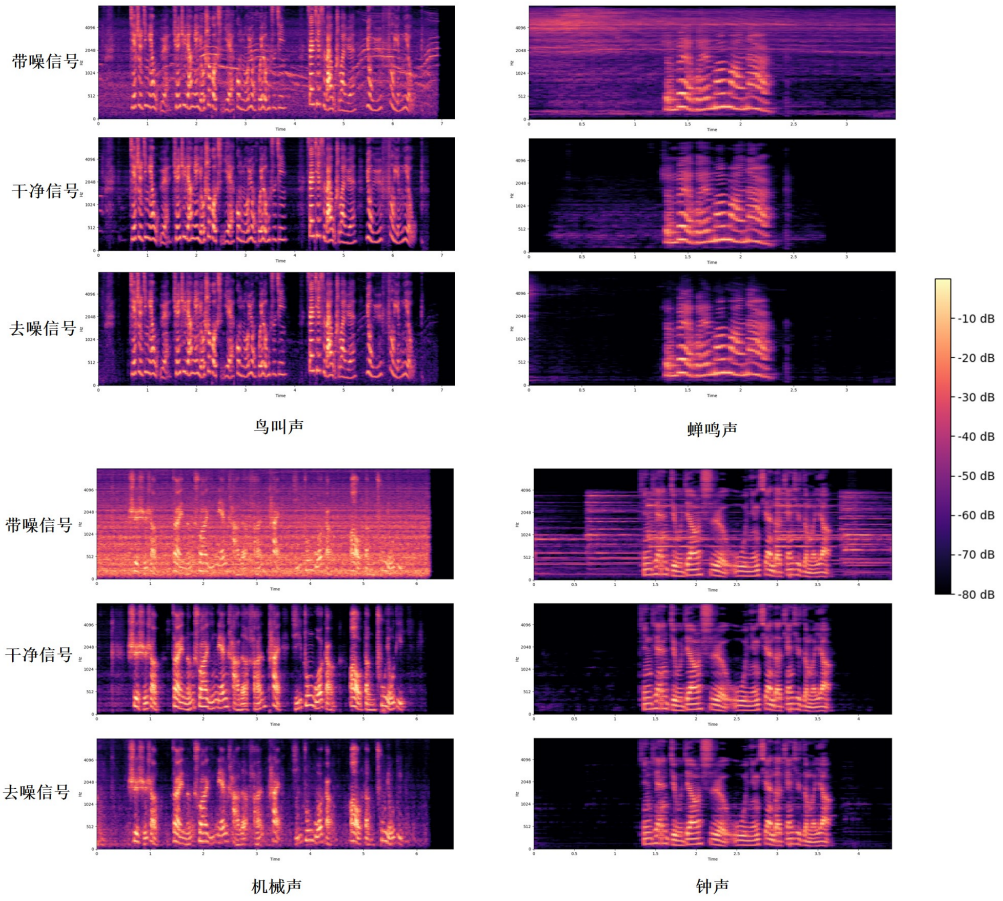

图 6-12: RTFCRN 对于几种噪声的语音增强效果时频谱（噪声从左到右从上到下分别为鸟叫声、蝉鸣声、机械声和钟声。每个子图从上到下分别表示噪声、参考干净语音和算法预测的干净语音时频谱）

实验中包括各种各样生活中常见的实采背景噪声，有的是平稳噪声，有的是突变噪声。RTFCRN 对这些噪声都有着比较良好的适应和去除能力。在实验的最后，我们展示了四个在不同背景噪声（鸟叫声、蝉鸣声、机械声和钟声）环境下进行语音信号增强得到的语音频谱图。如图6-12所示，无论是对于宽带的机械噪声，还是相对窄带的猫叫声，RTFCRN 都能够很好地处理。并且，从机械噪声语音时频谱中能够发现，在较低信噪比下，具有较高能量的噪声也能够被 RTFCRN 完全地去除。对于不稳定噪声，类似于钟声，算法也能够在很少非线性频谱失真的前提下进行语音增强。从蝉鸣噪声的语音增强结果中能发现，原始的参考干净语音信号中也存在着少量的背景噪声，但 RTFCRN 也能够将其去除，并很好地保留下语音时频谱成分。

6.7 本章小结

本章主要基于连续信号解码表示理论，提出了基于实时频域卷积时序网络的深度学习信号处理算法。该算法模型是一种轻量级的实时多通道信号表示方法，能够在符合任务目标的前提下控制算法的速度和参数量。我们使用带缓存状态的因果空洞卷积模型构建频率编码器 F-Encoder 和时域编码器 T-Encoder 对多通道语音信号进行编码，构建解码器对信号复数掩膜特征进行解码。通过门控循环单元提取有效信号的时序特征。网络训练时使用了任务相关的改进的尺度不变信噪比，以语音信号为例，还使用了基于人耳主观听感的短时客观可感度损失。为了降低 RTFCRN 参数量并保持其性能不变，我们还使用了模型压缩算法，特别是基于特征的蒸馏算法训练得到轻量级学生网络，并进行了参数动态量化。经过实验验证，RTFCRN 在语音信号增强任务上不仅显著提高了信号的信噪比，也有着良好的人耳感知性能。RTFCRN 算法相比常用的语音增强算法，能够适应多种噪声场景，有着更为良好的降噪性能，更少的网络参数量和在 CPU 上更快的实时运行速度。

第七章　总结与展望

信号处理是日常和生产生活中重要问题，随着深度学习的不断发展，信号表示在其中的地位日渐凸显，并且不同领域信号表示算法之间出现了许多共通点，然而并未提出一个系统的框架。所以，本文以信号表示算法为中心，在以往研究工作的基础之上，基于编解码理论形式化了信号表示算法理论框架。该理论框架将整个信号表示算法定义为多约束的优化问题，并依据信号类型的不同以及是否对信号分量进行解码表示将信号表示问题分为三类：信号纯表示问题、脉冲信号解码表示问题和连续信号解码表示问题。我们在每个独立的章节中详细说明了三种类型基于深度学习编解码理论的信号表示问题的特点并提出了解决方案，并在算法理论基础上考虑不同任务的限制，使其能够应用于实际，解决实际问题。本文研究对基于编解码理论的信号表示问题具有一定的普适性，是解决该问题的一种系统框架。事实证明，在每种类型的信号表示问题上，对应的算法框架都能以优秀的性能对不同领域的实际信号进行表示，具有良好的鲁棒性和实际性能，并实现了算法的落地。

本文将信号纯表示的两种范式相结合，充分利用了信号的内容表示和关系表示，通过解纠缠方法解决信号多义性问题，从而提出了信号纯表示框架 SPRF。越来越多的任务依赖于信号表示的有效性，而通过融合信号内容本身、信号元信息和关系信息，基于本文提出的多重关系损失函数，我们全面地解决了信号的纯表示问题。该问题的解决不仅影响直接的信号纯表示应用，比如音乐推荐，也对其它基于信号纯表示的任务表现有着显著影响。实验表明，SPRF 在音乐推荐任务上表现优异。比起同类算法，它能缓解音乐信号的长尾问题，解决实际存在的音乐推荐问题，其得到的音乐信号表示同样能应用于下游的相似艺人推荐任务并取得良好效果。

本文聚焦于多信号调制模式下的不定分量数目脉冲信号的解码表示，提出了脉冲信号掩膜 DPSM 算法。该算法以递归表示网络 RRN 为核心，辅以预处理和后聚类算法，用带噪的脉冲描述字信息完成脉冲信号分量表示和数目预测。该算法即使只使用 ToA 特征，在复杂环境、强噪声环境下也能以极高的精度预

测脉冲分量数目，并能够完成脉冲信号的分选。如果结合后处理 GMM 聚类算法，DPSM 算法能够更好地利用带噪的脉冲描述字，获得更精确的脉冲信号表示和分选结果。通过仿真实验验证，DPSM 算法能够应对脉冲信号分量表示的歧义性问题，也能对不同调制模式的雷达脉冲信号进行良好的解码表示，分选准确度和鲁棒性超越了现有的方法。

由于现实边缘设备的算力和实时性的限制，本文在平衡信号表示效果和深度学习模型参数量、速度的前提下，提出了实时频域卷积时序网络 RTFCRN。该网络通过独特的因果缓存结构和流式归一化等手段实现了实时的多通道信号处理。基于复数掩膜表示的任务目标和损失函数改善了信号分量的表示性能。通过知识蒸馏等压缩算法，模型得到了系统化的轻量化。在多个开源语音信号数据集和噪声数据集上，验证了 RTFCRN 的语音信号表示和增强效果，证明了算法的有效性和优越性，以及对不同种类噪声的鲁棒性。同时 RTFCRN 网络参数量小，实时性高，能够实际应用于边缘设备，对连续信号进行实时表示和处理。

本文从理论和实际应用两方面针对信号表示算法进行了相关研究，在本文研究的基础上，我们认为有以下几个可以继续深入的方向：

- 本文基于编解码理论的信号表示框架提取出了各个不同领域信号表示算法的共性，但主要是从总体上进行抽象和形式化，考虑的不一定周全。如果将该框架进一步细化，从三种类型的信号表示算法中进一步细化出更典型的理论方法，对将来信号表示的研究和各领域的借鉴会更有指导意义。
- 信号纯表示中的信号内容表示方法仅仅只使用了信号本身数据进行弱监督训练，倘若能用网络上大量的领域数据比如音乐数据库进行无监督预训练，再基于用户评价的各维度相似性打分进行训练，那么能得到更好的解纠缠内容表示。另一方面，对于关系表示，可以考虑将用户和信号、信号和信号、用户和用户的交互表示在同一张图上，从而得到更好的关系表示。
- 脉冲信号解码表示在多脉冲调制模式下需要额外的先验信息，但实际上也可以使用深度学习来直接预测脉冲调制模式类型。将其作为 DPSM 算法的其中一个模块，能够降低算法对于输入的要求难度，省略额外信息的提供。
- 在连续信号解码表示中，对于多通道信号的通道间空间信息的利用相对来说比较简单，将来可以以此为切入点，研究如何平衡空间信息的利用以及网络参数量和时延。

致　　谢

毕业论文的完成，标志着我在南京大学的研究生生活即将结束。回顾这三年，我感受到了科研的艰辛和乐趣，也收获了很多宝贵的知识和经验。在此过程中，有很多人给予了我无私的帮助和支持，让我能够克服困难，实现自己的目标。在此，我想向他们表达我诚挚的感谢。

首先，我要感谢我的导师申富饶教授。导师对我的学习和研究给予了耐心的指导和建议，为我提供了优良的科研条件和平台。导师经常通过个人讨论与我谈心，给我提供科研建议，并鼓励我积极参与国内外学术交流，参与项目实践。导师在学术上严谨求实、创新开拓、敢于探索，在人格上正直诚信、谦逊友善、乐于奉献、关心学生。导师不仅教会了我如何做科研，更教会了我如何做人。

其次，我要感谢 RINC 研究组的同学们。他们是我的好朋友，在学习上给予了我很多指点和启发，在生活上给予了我很多关心和陪伴。我们一起讨论问题、解决问题、分享成果、进步成长。我们之间相处和睦，经常一起出去吃饭、运动、旅游等等。在我的毕业设计中，他们也给予了我很多技术上和思路上的支持和协助。我们在项目中共同合作攻克难关，在实验室中共同创造美好回忆。

最后，还要感谢一下我的父母和女朋友。他们是我的精神支柱，在这三年中始终陪伴着我，在我遇到困难时鼓励着我，在我取得进步时为我感到欣慰，在每一个重要时刻都为着我的幸福而祝福。他们的支持和爱，让我在这条学术之路上不断前行，让我在每一次的选择中都坚定自信。

在此，我向所有帮助过我的人表示诚挚的感谢。未来的日子里，我将继续努力学习和工作，不负众望。

参考文献

[1] RABINER L R, GOLD B. Theory and application of digital signal processing [J]. Englewood Cliffs: Prentice-Hall, 1975.

[2] AHARON M, ELAD M, BRUCKSTEIN A. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation[J]. IEEE Transactions on Signal Processing, 2006, 54(11):4311-4322.

[3] HUANG K, AVIYENTE S. Sparse representation for signal classification[J]. Advances in Neural Information Processing Systems, 2006, 19.

[4] ZOU Z, CHEN K, SHI Z, et al. Object detection in 20 years: A survey[J]. Proceedings of the IEEE, 2023.

[5] REN S, HE K, GIRSHICK R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. Advances in Neural Information Processing Systems, 2015, 28.

[6] SIDDIQUE N, PAHEDING S, ELKIN C P, et al. U-net and its variants for medical image segmentation: A review of theory and applications[J]. IEEE Access, 2021, 9:82031-82057.

[7] DUAN K, BAI S, XIE L, et al. Centernet: Keypoint triplets for object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. [S.l.: s.n.], 2019: 6569-6578.

[8] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

[9] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[J]. Advances in Neural Information Processing Systems, 2014, 27.

[10] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30.

[11] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[12] SILVER D, HUANG A, MADDISON C J, et al. Mastering the game of go with deep neural networks and tree search[J]. Nature, 2016, 529(7587):484-489.

[13] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540):529-533.

[14] MNIH V, BADIA A P, MIRZA M, et al. Asynchronous methods for deep reinforcement learning[C]//International Conference on Machine Learning. [S.l.]: PMLR, 2016: 1928-1937.

[15] HESSEL M, MODAYIL J, VAN HASSELT H, et al. Rainbow: Combining improvements in deep reinforcement learning[C]//Proceedings of the AAAI Conference on Artificial Intelligence: volume 32. [S.l.: s.n.], 2018.

[16] COVINGTON P, ADAMS J, SARGIN E. Deep neural networks for youtube recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems. [S.l.: s.n.], 2016: 191-198.

[17] MALKOV Y A, YASHUNIN D A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 42(4):824-836.

[18] ZHOU G, ZHU X, SONG C, et al. Deep interest network for click-through rate prediction[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. [S.l.: s.n.], 2018: 1059-1068.

[19] WANG R, FU B, FU G, et al. Deep & cross network for ad click predictions [M]//Proceedings of the ADKDD'17. [S.l.: s.n.], 2017: 1-7.

[20] SZU H H, TELFER B A, KADAMBE S L. Neural network adaptive wavelets for signal representation and classification[J]. Optical Engineering, 1992, 31(9): 1907-1916.

[21] VAN DEN OORD A, DIELEMAN S, SCHRAUWEN B. Deep content-based music recommendation[J]. Advances in Neural Information Processing Systems, 2013, 26.

[22] SCHEDL M. Deep learning in music recommendation systems[J]. Frontiers in Applied Mathematics and Statistics, 2019:44.

[23] WANG S Q, HU G P, ZHANG Q L, et al. The background and significance of radar signal sorting research in modern warfare[J]. Procedia Computer Science, 2019, 154:519-523.

[24] WANG S Q, BAI J, HUANG X Y, et al. Analysis of radar emitter signal sorting and recognition model structure[J]. Procedia Computer Science, 2019, 154: 500-503.

[25] HASANI H, KHOSRAVI M R. Pulse deinterleaving based on fusing pdws and pri extraction process for radar-assisted edge devices considering computational costs[J]. EURASIP Journal on Wireless Communications and Networking, 2021, 2021(1):1-14.

[26] MARDIA H. New techniques for the deinterleaving of repetitive sequences[C]// IEE Proceedings F (Radar and Signal Processing): volume 136. [S.l.]: IET, 1989: 149-154.

[27] LI H, HAN Y, CAI Y, et al. Overview of the crucial technology research for radar signal sorting[J]. Systems Engineering and Electronics, 2005, 27(12): 2035-2040.

[28] WILEY R. Elint: The interception and analysis of radar signals[M]. [S.l.]: Artech, 2006.

[29] IJSSENNAGGER C, TEN HOORN S, GIRBES A, et al. A new speech enhancement device for critically ill patients with communication problems: a prospective feasibility study[J]. Intensive Care Medicine, 2017, 43:460-462.

[30] DAS N, CHAKRABORTY S, CHAKI J, et al. Fundamentals, present and future perspectives of speech enhancement[J]. International Journal of Speech Technology, 2021, 24(4):883-901.

[31] COOLEY J W, TUKEY J W. An algorithm for the machine calculation of complex fourier series[J]. Mathematics of Computation, 1965, 19(90):297-301.

[32] ALLEN J B, RABINER L R. A unified approach to short-time fourier analysis and synthesis[J]. Proceedings of the IEEE, 1977, 65(11):1558-1564.

[33] LOGAN B, et al. Mel frequency cepstral coefficients for music modeling.[C]// Ismir: volume 270. [S.l.]: Plymouth, MA, 2000: 11.

[34] JACOVITTI G, MANCA A, NERI A. Hypercomplete circular harmonic pyramids[C]//Wavelet Applications in Signal and Image Processing IV: volume 2825. [S.l.]: SPIE, 1996: 352-363.

[35] DUFF I S. A survey of sparse matrix research[J]. Proceedings of the IEEE, 1977, 65(4):500-535.

[36] STEPHANE M. A wavelet tour of signal processing[M]. [S.l.]: Elsevier, 1999.

[37] SOLIMAN S S, HSUE S Z. Signal classification using statistical moments[J]. IEEE Transactions on Communications, 1992, 40(5):908-916.

[38] KNEES P, SCHEDL M. A survey of music similarity and recommendation from music context data[J]. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2013, 10(1):1-21.

[39] KORZENIOWSKI F, ORAMAS S, GOUYON F. Artist similarity with graph neural networks[J]. arXiv preprint arXiv:2107.14541, 2021.

[40] DHRUV A, KAMATH A, POWAR A, et al. Artist recommendation system using hybrid method: A novel approach[M]//Emerging Research in Computing, Information, Communication and Applications. [S.l.]: Springer, 2019: 527-542.

[41] HANSEN C, HANSEN C, MAYSTRE L, et al. Contextual and sequential user embeddings for large-scale music recommendation[C]//Fourteenth ACM Conference on Recommender Systems. [S.l.: s.n.], 2020: 53-62.

[42] WANG M, RAO X, CHEN L, et al. SSAR-GNN: Self-supervised artist recommendation with graph neural networks[J]. 2022.

[43] SALHA-GALVAN G, HENNEQUIN R, CHAPUS B, et al. Cold start similar artists ranking with gravity-inspired graph autoencoders[C]//Fifteenth ACM Conference on Recommender Systems. [S.l.: s.n.], 2021: 443-452.

[44] DELDJOO Y, SCHEDL M, KNEES P. Content-driven music recommendation: Evolution, state of the art, and challenges[J]. arXiv preprint arXiv:2107.11803, 2021.

[45] LEE J, BRYAN N, SALAMON J, et al. Metric learning vs classification for disentangled music representation learning[J]. arXiv preprint arXiv:2008.03729, 2020.

[46] LEE J, BRYAN N, SALAMON J, et al. Disentangled multidimensional metric learning for music similarity[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2020: 6-10.

[47] HUANG Q, JANSEN A, ZHANG L, et al. Large-scale weakly-supervised content embeddings for music recommendation and tagging[C]//ICASSP 2020-2020

IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2020: 8364-8368.

[48] SCHINDLER A, KNEES P. Multi-task music representation learning from multi-label embeddings[C]//2019 International Conference on Content-Based Multimedia Indexing (CBMI). [S.l.]: IEEE, 2019: 1-6.

[49] THOMÉ C, PIWELL S, UTTERBÄCK O. Musical audio similarity with self-supervised convolutional neural networks[J]. arXiv preprint arXiv:2202.02112, 2022.

[50] CLEVELAND J, CHENG D, ZHOU M, et al. Content-based music similarity with triplet networks[J]. arXiv preprint arXiv:2008.04938, 2020.

[51] FESSAHAYE F, PEREZ L, ZHAN T, et al. T-recsys: A novel music recommendation system using deep learning[C]//2019 IEEE International Conference on Consumer Electronics (ICCE). [S.l.]: IEEE, 2019: 1-6.

[52] WHITTALL N. Signal sorting in esm systems[C]//IEE Proceedings F (Communications, Radar and Signal Processing): volume 132. [S.l.]: IET, 1985: 226-228.

[53] SRIDHARAN S, NNSSRK P, RIYA G, et al. Improved pulse repetition interval (pri) deinterleaving for electronic support measure (esm) receiver[J]. Int. Journal of Advanced Computing and Electronics Technology (IJACET), 2015, 2(3):37-43.

[54] BAGHERI M, SEDAAGHI M H. A new method for detecting jittered pri in histogram-based methods[J]. Turkish Journal of Electrical Engineering and Computer Sciences, 2018, 26(3):1214-1224.

[55] MANICKCHAND K, STRYDOM J J, MISHRA A K. Comparative study of toa based emitter deinterleaving and tracking algorithms[C]//2017 IEEE AFRICON. [S.l.]: IEEE, 2017: 221-226.

[56] FENG X, HU X, LIU Y. Radar signal sorting algorithm of k-means clustering based on data field[C]//2017 3rd IEEE International Conference on Computer and Communications (ICCC). [S.l.]: IEEE, 2017: 2262-2266.

[57] GENÇOL K, KARA A, AT N. Improvements on deinterleaving of radar pulses in dynamically varying signal environments[J]. Digital Signal Processing, 2017, 69:86-93.

[58] WANG S, GAO C, ZHANG Q, et al. Research and experiment of radar signal support vector clustering sorting based on feature extraction and feature selection [J]. IEEE Access, 2020, 8:93322-93334.

[59] MOTTIER M, CHARDON G, PASCAL F. Deinterleaving and clustering unknown radar pulses[C]//2021 IEEE Radar Conference (RadarConf21). [S.l.]: IEEE, 2021: 1-6.

[60] YOUNG J, HØST-MADSEN A, NOSAL E M. Deinterleaving of mixtures of renewal processes[J]. IEEE Transactions on Signal Processing, 2018, 67(4): 885-898.

[61] GONG X, MENG H, WANG X. A gmm-based algorithm for classification of radar emitters[C]//2008 9th International Conference on Signal Processing. [S.l.]: IEEE, 2008: 2434-2437.

[62] GASPERINI S, PASCHALI M, HOPKE C, et al. Signal clustering with class-independent segmentation[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2020: 3982-3986.

[63] GUO Q, TENG L, QI L, et al. A novel radar signals sorting method-based trajectory features[J]. IEEE Access, 2019, 7:171235-171245.

[64] LIU Z M, PHILIP S Y. Classification, denoising, and deinterleaving of pulse streams with recurrent neural networks[J]. IEEE Transactions on Aerospace and Electronic Systems, 2018, 55(4):1624-1639.

[65] MUN J, HA S, LEE J. Automotive radar signal interference mitigation using rnn with self attention[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2020: 3802-3806.

[66] LI F, YANG Z, YANG C. Radar signal sorting technology based on image processing and hough transform[C]//2018 International Conference on Microwave and Millimeter Wave Technology (ICMMT). [S.l.]: IEEE, 2018: 1-3.

[67] LI X, LIU Z, HUANG Z. Deinterleaving of pulse streams with denoising autoencoders[J]. IEEE Transactions on Aerospace and Electronic Systems, 2020, 56(6):4767-4778.

[68] REN C, CAO J, FU Y, et al. Improved method for pulse sorting based on pri transform[C]//Signal Processing, Sensor/Information Fusion, and Target Recognition XXIII: volume 9091. [S.l.]: International Society for Optics and Photonics, 2014: 90911T.

[69] GE Z, SUN X, REN W, et al. Improved algorithm of radar pulse repetition interval deinterleaving based on pulse correlation[J]. IEEE Access, 2019, 7: 30126-30134.

[70] MAO Y, HAN J, GUO G, et al. An improved algorithm of pri transform[C]// 2009 WRI Global Congress on Intelligent Systems: volume 3. [S.l.]: IEEE, 2009: 145-149.

[71] ANDERSON J A, GATELY M T, PENZ P A, et al. Radar signal categorization using a neural network[J]. Proceedings of the IEEE, 1990, 78(10):1646-1657.

[72] HAN J W, PARK C H. A unified method for deinterleaving and pri modulation recognition of radar pulses based on deep neural networks[J]. IEEE Access, 2021, 9:89360-89375.

[73] BOLL S. Suppression of acoustic noise in speech using spectral subtraction[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1979, 27(2): 113-120.

[74] PRATT W K. Generalized wiener filtering computation techniques[J]. IEEE Transactions on Computers, 1972, 100(7):636-641.

[75] GANNOT S, COHEN I. Speech enhancement based on the general transfer function gsc and postfiltering[J]. IEEE Transactions on Speech and Audio Processing, 2004, 12(6):561-571.

[76] BENESTY J, CHEN J, HUANG Y. A generalized mvdr spectrum[J]. IEEE Signal Processing Letters, 2005, 12(12):827-830.

[77] DOUMANIDIS C C, ANAGNOSTOU C, ARVANITI E S, et al. Rnnoise-ex: Hybrid speech enhancement system based on rnn and spectral features[J]. arXiv preprint arXiv:2105.11813, 2021.

[78] LIU Y, ZHANG H, ZHANG X, et al. Supervised speech enhancement with real spectrum approximation[C]//ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2019: 5746-5750.

[79] HAO X, SU X, HORAUD R, et al. Fullsubnet: A full-band and sub-band fusion model for real-time single-channel speech enhancement[C]//ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2021: 6633-6637.

[80] KIM J, EL-KHAMY M, LEE J. T-gsa: Transformer with gaussian-weighted self-attention for speech enhancement[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2020: 6649-6653.

[81] PANDEY A, WANG D. Tcnn: Temporal convolutional neural network for real-time speech enhancement in the time domain[C]//ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2019: 6875-6879.

[82] SU J, JIN Z, FINKELSTEIN A. Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks[J]. arXiv preprint arXiv:2006.05694, 2020.

[83] XU Y, YU M, ZHANG S X, et al. Neural spatio-temporal beamformer for target speech separation[J]. arXiv preprint arXiv:2005.03889, 2020.

[84] XU Y, ZHANG Z, YU M, et al. Generalized spatio-temporal rnn beamformer for target speech separation[J]. arXiv preprint arXiv:2101.01280, 2021.

[85] PORTNOFF M. Time-frequency representation of digital signals and systems based on short-time fourier analysis[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1980, 28(1):55-69.

[86] DINIZ P S, DA SILVA E A, NETTO S L. Digital signal processing: system analysis and design[M]. [S.l.]: Cambridge University Press, 2010.

[87] SIFUZZAMAN M, ISLAM M R, ALI M. Application of wavelet transform and its advantages compared to fourier transform[J]. 2009.

[88] JOHANSSON M. The hilbert transform[J]. Mathematics Master' s Thesis. Växjö University, Suecia. Disponible en internet: http://w3. msi. vxu. se/exarb/mj_ex. pdf, consultado el, 1999, 19.

[89] HUANG N E. Introduction to the hilbert–huang transform and its related mathematical problems[M]//Hilbert–Huang Transform and Its Applications. [S.l.]: World Scientific, 2014: 1-26.

[90] DAVIS S, MERMELSTEIN P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1980, 28(4):357-366.

[91] MOLAU S, PITZ M, SCHLUTER R, et al. Computing mel-frequency cepstral coefficients on the power spectrum[C]//2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (cat. No. 01CH37221): volume 1. [S.l.]: IEEE, 2001: 73-76.

[92] ÖZEN S, ÖNER M, ÇAVUŞLU M A, et al. Simulation and estimation of underwater acoustical tonals emanating from naval platforms[C]//2013 21st Signal Processing and Communications Applications Conference (SIU). [S.l.]: IEEE, 2013: 1-4.

[93] BANASIAK K, PIENIEZNY A. Radar pulse repetitive paterns detection[C]// 11-th International Radar Symposium. [S.l.]: IEEE, 2010: 1-4.

[94] WIENER N. Response of a non-linear device to noise[R]. [S.l.]: Massachusetts Inst Of Tech Cambridge Radiation Lab, 1942.

[95] CHEN J, BENESTY J, HUANG Y, et al. New insights into the noise reduction wiener filter[J]. IEEE Transactions on Audio,Speech, and Language Processing, 2006, 14(4):1218-1234.

[96] CIOFFI J, KAILATH T. Fast, recursive-least-squares transversal filters for adaptive filtering[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1984, 32(2):304-337.

[97] LEE D D, SEUNG H S. Learning the parts of objects by non-negative matrix factorization[J]. Nature, 1999, 401(6755):788-791.

[98] YANG Z, YUAN Z, LAAKSONEN J. Projective non-negative matrix factorization with applications to facial image processing[J]. International Journal of Pattern Recognition and Artificial Intelligence, 2007, 21(08):1353-1362.

[99] KIM J, PARK H. Sparse nonnegative matrix factorization for clustering[R]. [S.l.]: Georgia Institute of Technology, 2008.

[100] VEIT A, BELONGIE S, KARALETSOS T. Conditional similarity networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l.: s.n.], 2017: 830-838.

[101] RONNEBERGER O, FISCHER P, BROX T. U-net: Convolutional networks for biomedical image segmentation[C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. [S.l.]: Springer, 2015: 234-241.

[102] ZHU M, WANG S, LI Y. Model-based representation and deinterleaving of mixed radar pulse sequences with neural machine translation network[J]. IEEE Transactions on Aerospace and Electronic Systems, 2021, 58(3):1733-1752.

[103] ZEGHIDOUR N, GRANGIER D. Wavesplit: End-to-end speech separation by speaker clustering[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021, 29:2840-2849.

[104] TAN K. Convolutional and recurrent neural networks for real-time speech separation in the complex domain[M]. [S.l.]: The Ohio State University, 2021.

[105] 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. [S.l.: s.n.], 2016: 770-778.

[106] WANG X, HAN X, HUANG W, et al. Multi-similarity loss with general pair weighting for deep metric learning[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. [S.l.: s.n.], 2019: 5022-5030.

[107] SUNITHA M, ADILAKSHMI T. Music recommendation system with user-based and item-based collaborative filtering technique[C]//Networking Communication and Data Knowledge Engineering. Singapore: [s.n.], 2018: 267-278.

[108] BROOKNER E. Phased arrays and radars-past, present and future[J]. Microwave Journal, Cover Feature, 2006.

[109] JAJOO G, KUMAR Y, YADAV S K, et al. Blind signal modulation recognition through clustering analysis of constellation signature[J]. Expert Systems with Applications, 2017, 90:13-22.

[110] WANG X, HUANG G, ZHOU Z, et al. Radar emitter intrapulse signal blind sorting under modified wavelet denoising[J]. The Journal of Engineering, 2019, 2019(21):8013-8017.

[111] IKEDA S, MURATA N. A method of ica in time-frequency domain[C]//in Proc. ICA. [S.l.: s.n.], 1999.

[112] YU F, KOLTUN V. Multi-scale context aggregation by dilated convolutions[Z]. [S.l.: s.n.], 2016.

[113] BENGIO Y, LOURADOUR J, COLLOBERT R, et al. Curriculum learning[C]// Proceedings of the 26th Annual International Conference on Machine Learning. [S.l.: s.n.], 2009: 41-48.

[114] XI Y, WU Y, WU X, et al. An improved sdif algorithm for anti-radiation radar using dynamic sequence search[C]//2017 36th Chinese Control Conference (CCC). [S.l.]: IEEE, 2017: 5596-5601.

[115] TAAL C H, HENDRIKS R C, HEUSDENS R, et al. A short-time objective intelligibility measure for time-frequency weighted noisy speech[C]//2010 IEEE International Conference on Acoustics, Speech and Signal Processing. [S.l.]: IEEE, 2010: 4214-4217.

[116] RIX A W, BEERENDS J G, HOLLIER M P, et al. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs[C]//2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221): volume 2. [S.l.]: IEEE, 2001: 749-752.

[117] LE ROUX J, WISDOM S, ERDOGAN H, et al. Sdr–half-baked or well done? [C]//ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2019: 626-630.

[118] LAI B, XIANG H, SHEN F. Inf-cp: A reliable channel pruning based on channel influence[J]. arXiv preprint arXiv:2112.02521, 2021.

[119] GOU J, YU B, MAYBANK S J, et al. Knowledge distillation: A survey[J]. International Journal of Computer Vision, 2021, 129(6):1789-1819.

[120] HEO B, KIM J, YUN S, et al. A comprehensive overhaul of feature distillation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. [S.l.: s.n.], 2019: 1921-1930.

[121] HU Y, LIU Y, LV S, et al. Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement[J]. arXiv preprint arXiv:2008.00264, 2020.

[122] LI L, KÜRZINGER L, WATZEL T, et al. Lightweight end-to-end speech enhancement generative adversarial network using sinc convolutions[J]. Applied Sciences, 2021, 11(16):7564.

[123] OOSTERMEIJER K, WANG Q, DU J. Lightweight causal transformer with local self-attention for real-time speech enhancement.[C]//Interspeech. [S.l.: s.n.], 2021: 2831-2835.

[124] HO M T, LEE J, LEE B K, et al. A cross-channel attention-based wave-u-net for multi-channel speech enhancement.[C]//Interspeech. [S.l.: s.n.], 2020: 4049-4053.

[125] DIAZ-GUERRA D, MIGUEL A, BELTRAN J R. gpurir: A python library for room impulse response simulation with gpu acceleration[J]. Multimedia Tools and Applications, 2021, 80(4):5653-5671.

简历与科研成果

基本信息

向浩然，男，汉族，1997 年 9 月出生，云南省红河哈尼族彝族自治州人。

教育背景

2020 年 9 月 — 2023 年 6 月 南京大学人工智能学院 硕士

2016 年 9 月 — 2020 年 6 月 电子科技大学资源与环境学院 本科

攻读硕士学位期间完成的学术成果

1. **Haoran Xiang**, Furao Shen, Jian Zhao, "Deep ToA Mask Based Recursive Radar Pulse Deinterleaving", IEEE Transactions on Aerospace and Electronic Systems (2022).
2. Bilan Lai, **Haoran Xiang**, Furao Shen, "Inf-CP: A Reliable Channel Pruning based on Channel Influence", arXiv preprint arXiv:2112.02521 (2021).

攻读硕士学位期间的发明专利

1. 林晓慧，戴俊宇，张露曦，吕维宗，元民主，宋旭晨，贺杰，**向浩然**，歌单生成方法、装置、电子设备及存储介质。专利申请号：202210775313.9.
2. 申富饶，**向浩然**，赵健，一种基于 DTM 算法的雷达脉冲去交错方法。专利申请号：202111420021.5.
3. 郑泽忠，马鹏程，彭庆军，谢础航，**向浩然**，李江，一种基于电容型设备缺陷数据的设备缺陷时间预测方法。专利申请号：202110118818.3.

攻读硕士学位期间参与的科研课题

1. 国家自然科学基金面上项目“基于深度感知增量式联想记忆神经网络的信息融合系统研究”（项目编号 61876076，课题年限 2019 年 1 月-2022 年 12 月），负责语音增强相关问题的研究。