第 50 卷第 2 期 声 学 学 报 Vol. 50, No. 2
2025 年 3 月 ACTA ACUSTICA Mar., 2025

纪念马大猷先生诞辰 110 周年

基于离散小波变换及高低频子带解耦的低计算资源占用端到端语音识别*

田三力[1,2] 黎 塔[1,2†] 叶凌轩[1,2] 吴石松[3] 赵庆卫[1,2] 张鹏远[1,2]

(1 中国科学院声学研究所 语音与智能信息处理实验室 北京 100190)
(2 中国科学院大学 北京 100049)
(3 南方电网人工智能科技有限公司 广州 510000)

2024 年 7 月 12 日收到
2024 年 9 月 9 日定稿

摘要 针对目前端到端语音识别模型计算资源占用过高的问题, 提出了一种将离散小波变换 (DWT) 与端到端语音识别相融合的方法 (WLformer), 大幅降低计算资源占用量的同时还可提升识别性能。WLformer 的构建以目前端到端语音识别中广泛使用的 Conformer 模型为基础, 在模型中引入所提出的基于 DWT 的信号压缩模块, 该模块通过去除模型中间层表征内信息量较少的高频成分从而对该表征进行压缩, 进而降低模型的计算资源占用。此外还提出了 DWT 子带解耦前馈网络的子模块结构以替换原模型中部分前馈网络, 从而进一步降低模型的计算量。在 Aishell-1、HKUST 和 LibriSpeech 三个常用的中英文数据集上的实验表明, 提出的 WLformer 相较于 Conformer 的显存占用相对下降 47.4%, 计算量 Gflops 相对下降 39.2%, 同时还获得了平均 13.1% 的错误率改善。此外, WLformer 在计算资源占用少于其他主流端到端语音识别模型的情况下同样取得了更好的识别性能, 进一步验证了所提方法的有效性。

关键词 语音识别, 离散小波变换, 低计算资源占用, 端侧部署

PACS: 43.72 **DOI:** 10.12395/0371-0025.2024205 **CSTR:** 32049.14.11-2065.2024205

Low computational cost end-to-end speech recognition based on discrete wavelet transform and subband decoupling

TIAN Sanli[1,2] LI Ta[1,2†] YE Lingxuan[1,2] WU Shisong[3] ZHAO Qingwei[1,2] ZHANG Pengyuan[1,2]

(1 *Key Laboratory of Speech Acoustics and Content Understanding*, *Institute of Acoustics*, *Chinese Academy of Sciences* Beijing 100190)
(2 *University of Chinese Academy of Sciences* Beijing 100049)
(3 *China Southern Power Grid Artificial Intelligence Technology Co., Ltd.* Guangzhou 510000)

Received Jul. 12, 2024
Revised Sept. 9, 2024

Abstract To solve the problem of high computational cost of the current end-to-end automatic speech recognition (E2E ASR), a method (WLformer) that integrates discrete wavelet transform (DWT) with E2E ASR is proposed, which can significantly reduce the computing resource usage while improving performance. WLformer is built upon the mostly used Conformer model. WLformer introduces the proposed DWT Signal Compression Module, which compresses the model's middle hidden representation by removing its high-frequency components with less information. In addition, a new module structure named DWT Subband Decoupling Feed-Forward Network (DSD-FFN) is proposed to further reduce the model's computational cost. Experiments are conducted on Aishell-1, HKUST, and LibriSpeech datasets. The results show that WLformer achieves 47.4% relative memory usage reduction and 39.2% relative Gflops reduction, and achieves an average 13.1% relative character/word error rate reduction compared to Conformer. In addition, WLformer also achieves better recognition performance while occupying fewer computing resources than other mainstream E2E ASR models, which further verifies its effectiveness.

Keywords Speech recognition, Discrete wavelet transform, Low computational cost, Edge-device deployment

* 科技创新 2030 项目 (2022ZD0116103) 资助
† 通讯作者: 黎塔, lita@hccl.ioa.ac.cn

引言

近年来, 语音识别 (ASR) 技术作为一种关键的人机交互方法在人们的生活中起到越来越重要的作用。其中端到端 ASR 是目前主流的 ASR 技术[1-4], 尤其是近年来各种 Transformer[5-10] 类模型的出现, 进一步促进了端到端 ASR 的发展。然而, 端到端 ASR 的优异性能却往往伴随着巨大的计算资源占用, 这使得其难以部署在计算资源有限的端侧设备上, 一定程度上限制了其应用场景[11,12]。

为了解决这个问题, 国内外众多研究工作聚焦于如何有效地降低端到端 ASR 的计算量。其中一种方法是利用模型压缩技术降低模型参数从而降低计算资源占用量。文献 [13] 使用相关性、能量和基于梯度的指标对模型的每个层进行评分, 并对评分进行排序以挑选出用于修剪的候选层, 并修剪训练期间指标较低的层以降低模型的计算量。文献 [14] 设计了一种两阶段知识蒸馏算法, 通过让计算量小但性能差的学生模型去模仿计算量大但性能好的教师模型, 使学生模型的性能得到有效的提升。另一种降低计算量的主要方法是设计更合理且高效的端到端 ASR 模型结构, 以使用更少的计算资源占用达到更好的识别性能。文献 [7] 提出了一种名为 BranchFormer 的模型, 将用于提取全局特征的注意力模块和用于提取局部特征的 cgMLP 模块从常用的级联结构变为双分支并联结构, 同时文中还尝试仅在训练阶段采用完整的双分支并联结构, 但是在推理的时候去除其中某个分支。虽然该推理方式会降低性能, 但是可以进一步降低模型的计算量。文献 [8] 在 Branchformer 的基础之上提出了 E-Branchformer, 通过引入深度卷积增强原有 Branchformer 的合并模块, 这允许在合并全局分支与局部分支的信息时考虑相邻特征之间的关系。此外, 在 E-Branchformer 中还增加了额外的逐点 (point-wise) 模块, 进一步提升了模型处理局部信息的能力。文献 [9] 提出了 Efficient-Conformer, 以 Conformer 模型[6] 为基底模型, 使用卷积神经网络对模型中间层表征进行降采样, 通过降低编码序列的长度来降低模型的计算量; 同时还提出了多头分组注意力机制, 通过在特征维度上将序列的相邻时间元素进行分组, 然后应用缩放点积注意力, 从而降低原注意力机制的复杂度。文献 [10] 同样以 Conformer 模型作为基底模型, 提出名为 Squeezeformer 的低计算量端到端 ASR 架构, 与 Efficient-Conformer 类似, Squeezeformer 也使用卷积神经网络对模型的中间层表征进行降采样, 但 Squeezeformer 又在模型的最后引入了一个上采样层, 并通过一个跳跃链接残差来链接该上采样层与降采样层, 以融合两者之间的信息; 此外, Squeezeformer 还使用深度可分离卷积替换模型前端的二维卷积特征提取模块, 进一步降低了模型的计算量。文献 [15]提出了低计算量端到端 ASR 模型 Zipformer, 与 Squeezeformer 类似, 该模型也采用了类似 U-Net 的编码器结构, 在模型的中间使用卷积神经网络对模型的隐层表征进行降采样, 而在模型的后半部分再通过上采样恢复原采样率, 同时通过注意力权重复用机制以降低计算量; 此外, 还提出了训练优化器 ScaledAdam, 通过每个张量的当前比例来缩放更新, 从而实现更好的性能。

上述工作探索了如何降低端到端 ASR 的计算资源占用, 其中 Efficient-Conformer[9] 与 Squeezeformer[10] 验证了在模型的中间部分使用卷积神经网络对隐层表征进行降采样可以有效降低模型的计算资源占用。但是本文认为使用卷积神经网络对 ASR 模型的隐层表征进行降采样并非是最优选择, 使用卷积神经网络进行降采样时并未考虑到语音信号内高频与低频成分之间的信息量存在巨大差异的先验信息, 且使用卷积神经网络直接对序列进行降采样并未考虑采样定理, 这可能会导致在降采样时发生频域混叠。已有研究表明[16-18], 语音信号的主要能量与信息大多集中在低频成分中, 而高频成分中语义相关信息较少。因此, 本文从语音信号高频与低频成分之间的信息量差异出发, 将离散小波变换 (DWT) 与深度学习模型相结合, 将高频与低频成分进行解耦并进行差异化操作, 提出新的低计算资源占用端到端 ASR 模型 WLformer。

WLformer 以端到端 ASR 中常用的 Conformer 模型作为基底模型。首先, WLformer 引入所提出的 DWT 信号压缩模块, 对模型中间层的隐层表征进行信号压缩, 将绝大部分信息压缩至长度仅为原来一半的序列内, 从而降低模型后续的计算量。DWT 信号压缩模块首先会将输入的隐层表征通过调整滤波器和小波滤波器 (分别具有低通与高通特性), 得到信号带宽仅为原来一半的低频和高频子带分量, 然后根据带限信号的采样定理, 可以将该低频与高频子带分量分别进行二倍降采样而不造成信息损失[19]。由于经过调整滤波器和小波滤波器后得到的信号的带宽为原来的一半, 如果在此处进行二倍以上的降采样, 就会造成频域混叠, 从而造成信息量的损失, 并且无法再重构回原信号。二倍降采样后得到的输

出分别被称为尺度系数与小波系数, 其中尺度系数与小波系数分别表示了模型隐层表征的低频与高频成分[19]。语音信号的主要能量与信息都集中在低频成分中, 高频成分中语义相关信息较少[16-18], 因此 WLformer 中的 DWT 信号压缩模块将表示高频成分的小波系数从中去除, 而仅保留表示低频成分的尺度系数, 从而将大部分信息压缩至更短的序列内, 提高信息密度, 降低计算资源占用的同时提升性能。

此外, 位置前馈网络 (FFN) 模块是 Transformer 类模型性能的关键之一, FFN 模块在 Transformer 类模型的总计算量中占据了较大比例, 降低 FFN 模块的计算量是重要且较为困难的[20]。但是, 之前的 ASR 模型研究[5-10,15]集中于从分支结构、注意力机制、降采样机制以及参数复用等方面降低模型计算量, 并未对如何降低 FFN 模块的计算量进行过探究。针对该问题, 本文同样基于高低频解耦的思想提出了 DWT 子带解耦前馈网络 (DSD-FFN), 用于替换原模型中的部分 FFN 模块, 以在几乎不影响性能的情况下进一步降低模型的计算资源占用。

本文通过引入高低频子带解耦的思想提出了 DWT 信号压缩模块与 DSD 子带解耦前馈网络, 所提 WLformer 相较于其他主流端到端 ASR 模型在 Aishell-1[21]、HKUST[22] 和 LibriSpeech[23] 三个常用的中英文数据集上以更少的计算资源占用取得了更好的识别性能。

1　WLformer 的整体结构

WLformer 的整体结构如图 1 所示。对于输入的一段待识别的语音数据 S, 首先通过一个二维卷积特征提取模块。之后, 该二维卷积特征提取模块的输出会通过 N_1 个 Block-A 模块, 该 N_1 个 Block-A 模块被命名为 Group1。而后将 Group1 的输出通过 DWT 信号压缩模块-1, 将 Group1 输出的隐层表征压缩至时序长度仅为原来一半的序列内。之后, DWT 信号压缩模块-1 的输出将通过 N_2 个 Block-B 模块, 该 N_2 个 Block-B 模块被命名为 Group2。与上述类似, Group2 输出的隐层表征被输入进 DWT 信号压缩模块-2, 时序长度进一步被减少一半。此后, DWT 信号压缩模块-2 的输出则被输入进 N_3 个 Block-A 模块, 该 N_3 个 Block-A 模块被命名为 Group3。最后, Group3 的输出则会通过一个线性分类器得到模型的最终输出。

2　DWT 与 DWT 信号压缩模块

前期研究工作发现, 语音信号的主要能量与信息大多集中在低频成分中, 而高频成分中语义相关的信息较少[16-18]。WLformer 首次将语音信号中的低频与高频成分之间的信息量差异作为先验知识, 通过解耦低频与高频成分, 而后分别对两者进行差异化操作, 降低端到端 ASR 模型的计算资源占用。此外, DWT 是一种常用的时频分析算法, 具有很强的时频域局部化能力, 可以很好地将信号分解为低频与高频成分, 且由于其优异的时频特性被广泛应用于信号压缩、信号降噪等领域[24-25]。因此, WLformer 中使用 DWT 作为工具进行低频与高频成分的解耦。

一阶 DWT 算法如图 2(a) 所示。不失一般性, 设一阶 DWT 的输入为一段长度为 T 的序列 $\boldsymbol{X}=[X_1, X_2,\cdots,X_T]$, 一阶 DWT 会将 $\boldsymbol{X}$ 首先通过调整滤波器 $\boldsymbol{h}$ 与小波滤波器 $\boldsymbol{g}$, 将其分解为低频子带与高频子带分量, 其中 $\boldsymbol{h}$ 和 $\boldsymbol{g}$ 分别具备低通与高通特性, 具体可写为

$$\boldsymbol{h}=[h_1,h_2,\cdots,h_J], \tag{1}$$

$$\boldsymbol{g}=[g_1,g_2,\cdots,g_J]. \tag{2}$$

然后根据带限信号采样定理, 可将该分解后的低频与高频子带分量进行二倍降采样而不造成任何信息量的损失[19]。因此, 为了不保留冗余信息, 对两个子带分量分别进行二倍降采样, 所得到的输出被分别称为尺度系数 $\boldsymbol{c}$ 和小波系数 $\boldsymbol{d}$:

$$\boldsymbol{c}=[c_1,c_2,\cdots,c_{T/2}], \tag{3}$$

$$\boldsymbol{d}=[d_1,d_2,\cdots,d_{T/2}], \tag{4}$$

其中, 尺度系数 $\boldsymbol{c}$ 表示输入信号 $\boldsymbol{X}$ 的低频成分, 而小波系数 $\boldsymbol{d}$ 则表示输入信号 $\boldsymbol{X}$ 的高频成分[19], 且 $\boldsymbol{c}$ 与 $\boldsymbol{d}$ 的长度皆为 $T/2$, 即 $\boldsymbol{X}$ 长度的一半。上述过程的数学表达为

$$\boldsymbol{c}(n)=\sum_{j=1}^{J}\boldsymbol{X}(2n-j)\cdot\boldsymbol{h}(j), \tag{5}$$

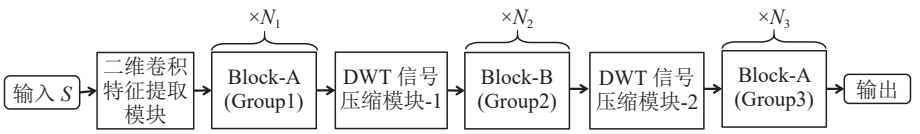

图 1　WLformer 的整体结构

$$\boldsymbol{d}(n)=\sum_{j=1}^{J}\boldsymbol{X}(2n-j)\cdot\boldsymbol{g}(j). \tag{6}$$

WLformer 中的 DWT 信号压缩模块则是基于一阶 DWT 设计的, 其中使用的小波基为信号压缩领域常用的多贝西小波-4 (Db4)。WLformer 与其中的 DWT 信号压缩模块具体结构如图 2(b) 所示。在 WLformer 的两个 DWT 信号压缩模块中, 模型隐层表征首先通过一阶 DWT 分解为表示低频成分与高频成分的尺度系数与小波系数, 然后考虑到语音信号高频成分中语义相关的信息量较少[16-18], DWT 信号压缩模块将表示高频成分的小波系数 $\boldsymbol{d}_1$ 与 $\boldsymbol{d}_2$ 去除, 仅保留尺度系数 $\boldsymbol{c}_1$ 与 $\boldsymbol{c}_2$。如此, 每经过一次 DWT 信号压缩模块, 隐层表征中信息量较少的高频成分都会被去除, 而蕴藏在低频成分中的绝大部分信息都被压缩至时序长度仅为原来一半的序列内, 从而提高了信息的密集程度, 减少了无用的计算, 进而大幅降低模型的计算资源占用。在 5.1 节与 5.2 节中将通过实验与可视化验证该方法的有效性与设想的合理性。

3 WLformer 的 Block 结构

WLformer 中有两种不同的 Block, 分别为 Block-A 与 Block-B, 具体结构如图 3(a)(b) 所示。其中 Block-A 的结构与标准 Conformer Block[6] 结构一致, 由两个 FFN、用于提取全局特征的多头注意力模块、用于提取局部特征的卷积模块以及一系列

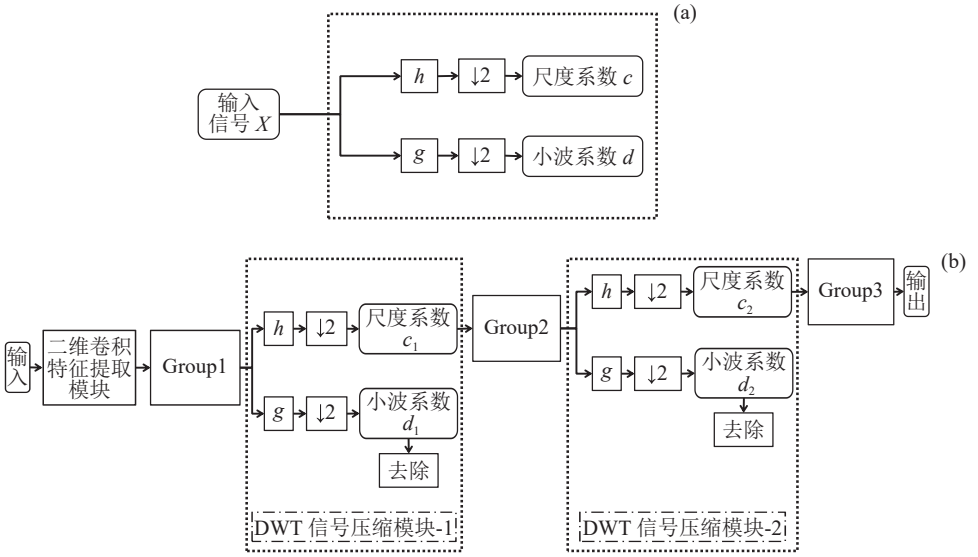

图 2 DWT 与 WLformer (a) 一阶 DWT 算法; (b) WLformer 与 DWT 信号压缩模块的具体结构

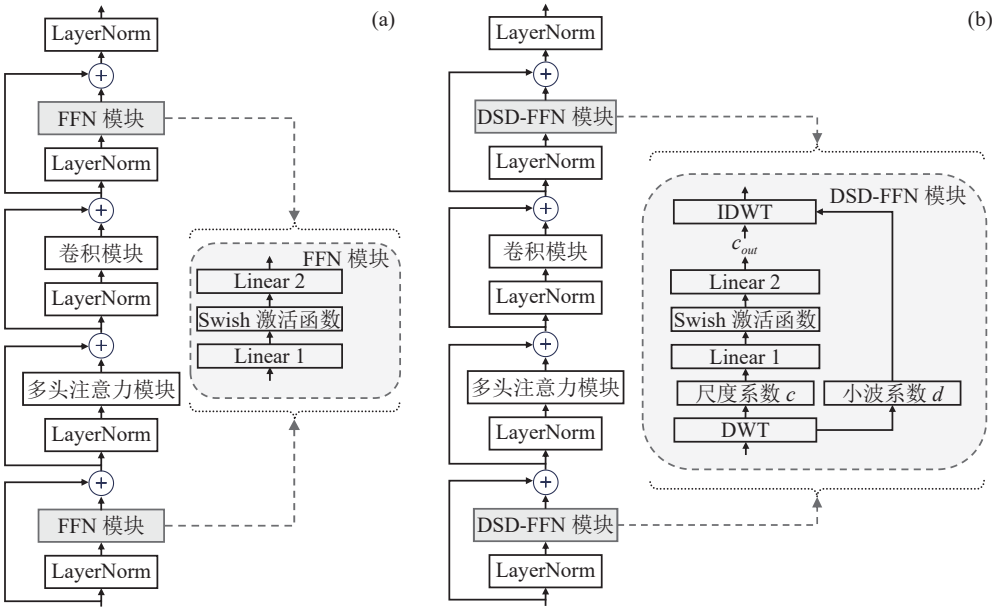

图 3 WLformer 的 Block 结构 (a) Block-A 具体结构; (b) Block-B 具体结构

LayerNorm 和残差结构组成。而与 Block-A 不同的是, Block-B 将 Block-A 中的 FFN 模块替换为本文提出的 DSD-FFN, 其中 DSD-FFN 模块的计算量约为标准 FFN 模块的一半。

标准 FFN 的具体结构如图 3(a) 中的右侧虚线框内所示, 其中包含了两个 Linear 层, 记为 Linear 1 与 Linear 2, 其对序列的每一帧向量都独立进行计算。在两个 Linear 层之间有一个非线性 Swish 激活函数。在 Transformer 类模型中, FFN 模块占用了较大的计算量, 并且 FFN 对 Tranformer 类模型的性能至关重要。针对该问题, 本文采用与 DWT 信号压缩模块类似的思想设计了 DSD-FFN, 通过将信号解耦为低频与高频成分后分别进行差异化的操作从而降低计算量。DSD-FFN 的具体结构如图 3(b) 中的右侧虚线框内所示。不失一般性, 设 DSD-FFN 模块的输入为 $\boldsymbol{I}=[I_1,I_2,\cdots,I_T]$, 首先将 $\boldsymbol{I}$ 通过 DWT, 获得表示 $\boldsymbol{I}$ 中低频成分的尺度系数 $\boldsymbol{c}$ 与表示高频成分的小波系数 $\boldsymbol{d}$, 且 $\boldsymbol{c}$ 与 $\boldsymbol{d}$ 的序列长度皆为 $T/2$。接下来, DSD-FFN 仅将尺度系数 c 进行神经网络计算: 通过 Linear 1、非线性激活函数 Swish 以及 Linear 2, 这部分得到的输出设为 $\boldsymbol{c}_{\text{out}}$, 而小波系数 $\boldsymbol{d}$ 不参与神经网络的计算, 因此 DSD-FFN 中参与神经网络计算的时序长度仅为标准 FFN 中的一半。之后, 与 DWT 信号压缩模块不同的是, DSD-FFN 中的高低频解耦操作仅局限于 DSD-FFN 模块的内部而不影响到 Group2 中其余部分的计算与特征提取, 因此 DSD-FFN 模块并不会将小波系数 $\boldsymbol{d}$ 去除, 而是在 DSD-FFN 模块的最后, 将经过神经网络计算的低频成分 $\boldsymbol{c}_{\text{out}}$ 与未经任何操作的高频成分 $\boldsymbol{d}$ 通过逆离散小波变换 (IDWT) 进行融合重构, 重构后的序列设为 $\boldsymbol{O}$, $\boldsymbol{O}$ 的时序长度与该模块的输入 $\boldsymbol{I}$ 一致。$\boldsymbol{c}_{\text{out}}$ 与 $\boldsymbol{d}$ 通过 IDWT 获得 $\boldsymbol{O}$ 的表达式:

$$\boldsymbol{O}(n)=\text{IDWT}(\boldsymbol{c}_{\text{out}},\boldsymbol{d})=\sum_{j=1}^{J}\boldsymbol{c}_{\text{out}}(n-2j)\cdot\boldsymbol{h}_r(j)+\sum_{j=1}^{J}\boldsymbol{d}(n-2j)\cdot\boldsymbol{g}_r(j),\quad(7)$$

其中, $\boldsymbol{h}_r$ 与 $\boldsymbol{g}_r$ 分别为 IDWT 中的调整重构滤波器与小波重构滤波器。需说明的是, DWT 与 IDWT 互为逆变换, 即将输入 $\boldsymbol{I}$ 经 DWT 分解为尺度系数 $\boldsymbol{c}$ 与小波系数 $\boldsymbol{d}$ 后, $\boldsymbol{c}$ 与 $\boldsymbol{d}$ 可通过 IDWT 完全重构回输入序列 $\boldsymbol{I}$。

WLformer 中并非全部 FFN 模块都替换为 DSD-FFN 模块。在 WLformer 中, 仅 Group2 中使用 DSD-FFN, 有如下两个原因: 第一, 高低频的解耦操作不应在模型的浅层进行, 在模型浅层时应尽量从包含完整频带信息的序列中提取特征, 这可以为模型的后续层提供更加丰富的信息, 也为后续的高低频解耦操作提供基础, 因此 Group1 中使用标准 FFN 而非 DSD-FFN; 第二, 输入 Group3 的信号已经过两次 DWT 信号压缩模块, 因此 Group3 的输入信号压缩比例已处于较高水平。若在 Group3 中使用 DSD-FFN, 将会从被进一步压缩的序列中提取特征与信息, 过高的压缩比例可能导致信息丢失与性能下降。因此, WLformer 选择在位于模型中间部分的 Group2 中使用 DSD-FFN, 此设置可以在几乎不影响模型性能的情况下进一步降低模型的计算资源占用。在 5.3 节中也通过实验验证了该设想与此 DSD-FFN 位置设置的合理性。

4　实验设置

4.1　实验数据集

为验证所提方法的有效性, 选取三个常用的中英文数据集 Aishell-1[21]、HKUST[22] 和 LibriSpeech[23] 进行相关实验。Aishell-1 是中文语音数据集, 包含时长 178 h 的语音数据。录音文本涉及智能家居、无人驾驶、工业生产等 11 个领域。录制过程使用 3 种不同设备: 高保真传声器、Android 系统手机、iOS 系统手机。400 名来自中国不同口音区域的发言人参与录制。该数据被划分为“train”、“dev”和“test”三个子集。其中“train”子集包含约 120000 条数据、“dev”子集包含约 14000 条数据、“test”子集包含约 7000 条数据。HKUST 是香港科技大学主导录制的中文普通话电话语音数据集, 其中包含了 200 h 的语音数据, 并被划分为“train”和“test”两个子集。每条电话语音包括两个说话人, 内容为日常对话, 囊括了社会、经济、娱乐、体育等多个话题。LibriSpeech 为大规模英语语音数据集, 数据时长大约为 1000 h, 在实验的训练阶段使用“train-clean-100”、“train-clean-360”和“train-other-500”这三个子集, 在评估阶段使用“test-clean”和“test-other”两个子集。

4.2　基线与对比方法参数设置

所提 WLformer 以目前业界常用的 Conformer[6] 架构作为基底模型, 因此采用 Conformer 架构作为基线系统。同时, 除了与基线系统的对比, 还进一步与其他主流端到端 ASR 模型进行了对比, 这些对比方法分别为 Transformer[5]、E-Branchformer[8]、Efficient-Conformer[9] 和 Squeezeformer[10]。训练均使用 CTC 损失函数, 且不使用外部语言模型辅助解码。

模型的实现使用基于 PyTorch[26] 的开源端到端 ASR 训练工具 ESPnet-2[27]。其中 Conformer、Transformer 以及 E-Branchformer 在 ESPnet-2 中提供了开源的模型框架代码与训练配置, 其模型框架与训练相关的超参数均遵循 ESPnet-2 开发人员提供的最优设置。Efficient-Conformer 与 Squeezeformer 也经过了仔细的参数调优以确保其性能。不同模型结构的具体设置如下:

(1) Conformer 模型参数设置: 多头注意力模块的维度是 256、头数为 4, FFN 中的 Linear 1 与 Linear 2 的维度皆为 2048, 卷积模块中卷积核的大小为 31, 总 block 数为 12。

(2) Transformer 模型参数设置: 多头注意力模块的维度是 256、头数为 4, FFN 中的 Linear 1 与 Linear 2 的维度皆为 2048, 总 block 数为 12。

(3) E-Branchformer 模型参数设置: 多头注意力模块的维度是 256、头数为 4, cgMLP 中的 Linear 维度为 1024, cgMLP 中的卷积层卷积核大小为 31, FFN 中的 Linear 1 与 Linear 2 维度皆为 1024, 总 block 数为 12。

(4) Efficient-Conformer 模型参数设置: Efficient-Conformer 中时序降采样卷积神经网络层位于模型的第 4 个 block 内, 第 1~4 个 block 内使用分组多头注意力模块, 第 5~12 个 block 内使用标准的多头注意力模块, 其维度为 256, 头数为 4; FFN 中的 Linear 1 与 Linear 2 维度皆为 2048, 卷积模块的卷积核大小为 31; 总 block 数为 12。

(5) Squeezeformer 模型参数设置: 多头注意力模块的维度是 256、头数为 4, FFN 中的 Linear 1 与 Linear 2 的维度皆为 2048, 卷积模块的卷积核大小为 31; 时序降采样卷积神经网络层位于模型的第 6 个 block, 而上采样层位于模型的第 12 个 block; 总 block 数为 12。

(6) WLformer 模型参数设置: 多头注意力模块的维度是 256、头数为 4, FFN 中的 Linear 1 与 Linear 2 的维度皆为 2048; DWT 信号压缩模块-1 与 DWT 信号压缩模块-2 分别位于模型的第 4 个与第 8 个 block 前; WLformer 中 Group1、Group2 以及 Group3 内卷积模块的卷积核大小分别为 31, 15, 7; 总 block 数为 12。

(7) WLformer-S 模型参数设置: 多头注意力模块的维度是 256、头数为 4, FFN 中的 Linear 1 与 Linear 2 的维度皆为 1024; DWT 信号压缩模块-1 与 DWT 信号压缩模块-2 分别位于模型的第 4 个与第 8 个 block 前; WLformer 中 Group1、Group2 以及 Group3 内卷积模块的卷积核大小分别为 31, 15, 7; 总 block 数为 12。

4.3 评价指标

在中文数据集 Aishell-1 与 HKUST 上使用字错率 (CER) 作为识别性能评价指标, CER 是所有替换错、删除错、插入错的字数与总字数的比值。在英文数据集 Librispeech 上使用词错率 (WER) 作为识别性能评价指标, WER 的计算方式与 CER 一致, 但是是以词作为最小单位。CER 或 WER 的数值越小, 表示语音识别系统的识别性能越好。此外, 本文使用 30 s 语音数据作为输入时模型的显存占用、以及十亿浮点运算数 (GFLOPs) 作为衡量模型资源占用和算力开销的指标, 其中显存占用越小, 说明该模型的资源占用越少; GFLOPs 数值越低说明模型需要的浮点运算就越少, 即该模型所需的计算量越小。

5 实验结果

5.1 WLformer 与对比方法

表 1 为所提 WLformer 与其他业界主流端到端 ASR 模型计算资源占用的对比, 表 2 为在 Aishell-1[21]、HKUST[22]、Librispeech[23] 三个数据集上 WLformer 与其他业界主流到端 ASR 模型识别性能的对比。综合表 1 与表 2 的结果来看, 虽然 WLformer 的显存占用、Gflops 这两个衡量模型资源占用和算力开销的指标与 Transformer 大致相当, 但是其识别性能却远好于 Transformer。除 Transformer 之外, WLformer 在显存占用和 Gflops 明显低于其余方法的同时, 仍取得了最佳的识别性能表现。具体而言, 与本文的基线系统 Conformer 模型相比, WLformer 在显存占用相对下降 47.4%、计算量 Gflops 相对下降 39.2% 的情况下, 仍获得了性能提升: 在 Aishell-1 的 dev 和 test 集上, WLformer 的 CER 分别相对下降 18.6% 和 18.8%; 在 HKUST 的 test 集上, WLformer 的 CER 相对下降 2.2%; 在 Librispeech 的 test-clean 和 test-other 集上, WLformer 的 WER 分别相对下降 11.4% 和 14.5%。此外, 相较于其他对比方法, WLformer 同样可以在使用更少资源占用与算力开销的同时获得更佳的识别性能。WLformer-S 的性能较 WLformer 有所下降, 但仍具有一定的性能优势。这进一步证明了 WLformer 结构设计的有效性, 以及所提高低频解耦处理的合理性。

上述模型皆采用纯 CTC 准则训练的模型, 因此模型不含有 Decoder 部分。接下来将在模型具有

表 1　WLformer 与其他方法计算资源占用的对比

方法	显存占用 (GB)	Gflops	参数量 (M)
Conformer[6]	1.52	42.1	34.60
Transformer[5]	0.79	27.3	**17.61**
E-Branchformer[8]	1.34	34.3	26.24
Efficient-Conformer[9]	0.93	31.0	34.60
Squeezeformer[10]	1.21	26.6	32.96
WLformer	0.80	25.6	34.55
WLformer-S	**0.63**	**20.5**	20.86

表 2　WLformer 与其他方法识别性能的对比

方法	Aishell-1		HKUST	Librispeech	
	CER (%)		CER (%)	WER (%)	
	dev	test	test	test-clean	test-other
Conformer[6]	5.9	6.4	22.6	3.5	8.3
Transformer[5]	7.0	7.5	24.0	4.8	12.2
E-Branchformer[8]	5.4	5.8	22.3	3.4	8.0
Efficient-Conformer[9]	5.3	5.9	22.6	3.4	8.1
Squeezeformer[10]	5.5	6.0	22.4	**3.1**	7.5
WLformer	**4.8**	**5.2**	**22.1**	**3.1**	**7.1**
WLformer-S	4.9	5.5	22.4	3.3	7.6

表 3　LibriSpeech 数据集 Zipformer 与 WLformer 对比

数据集	LibriSpeech	训练准则	解码方式	test-clean	test-other	Gflops	参数量
Libri-Speech	Zipformer[15]	Pruned transducer	Pruned transducer	2.4	5.7	40.8	23.3M
	Zipformer[15]	CTC/AED	CTC解码	3.0	7.0	40.8	46.3M
	Zipformer[15]	CTC/AED	CTC/AED联合解码	2.5	6.1	40.8	46.3M
	WLformer	CTC/AED	CTC解码	2.8	6.5	25.6	46.8M
	WLformer	CTC/AED	CTC/AED联合解码	**2.3**	**5.5**	25.6	46.8M

表 4　Aishell 数据集 Zipformer 与 WLformer 对比

数据集	模型	训练准则	解码方式	Dev	Test	参数量
Aishell-1	Zipformer[15]	Pruned transducer	Pruned transducer	4.4	4.7	30.2M
	Zipformer[15]	Pruned transducer	Pruned transducer	**4.1**	**4.4**	73.4M
	WLformer	CTC/AED	CTC解码	4.5	4.9	46.2M
	WLformer	CTC/AED	CTC/AED联合解码	**4.1**	**4.4**	46.2M

Decocder 的情况下进行 Zipformer 与 WLformer 的对比。其中 LibriSpeech 与 Aishell-1 数据集上的对比结果分别如表 3 与表 4 所示。

在 LibriSpeech 数据集中, Zipformer[15] 公开了基于 Pruned transducer 与基于 CTC/AED 联合训练准则所得到的模型性能结果、编码器计算量 Gflops 与模型整体参数量, 且由于 WLformer 模型训练所使用的 Espnet2 开源框架原生代码中不包含 Pruned transducer 训练准则, 因此所对比的 WLformer 相应地采用 CTC/AED 联合训练准则。从表 3 对比结果中可以看出, WLformer 相较于同样使用 CTC/AED 联合训练准则的 Zipformer, 在模型参数量基本相同、编码器的计算量更小的情况下, 取得了更佳的识别性能; 此外, 虽然基于 CTC/AED 联合训练准则的 WLformer 模型的整体参数量大于基于 Pruned transducer 训练的 Zipformer, 但是 WLformer 的 Gflops 更小, 并且性能更佳。

在 Aishell-1 数据集中, Zipformer 仅公开了基于 Pruned transducer 训练准则的性能结果与相应的模型参数量, 所对比的 WLformer 类似地采用 CTC/AED 联合训练准则。由表 4 对比结果可见, 参数量 46.2M 的 WLformer 与 73.4M 的 Zipformer 性能相当, 这同

样验证了 WLformer 方法的合理性。

5.2 DWT 信号压缩模块的消融实验

Efficient-Conformer[9] 与 Squeezeformer[10] 验证了在模型的中间部分对隐层表征进行降采样操作可以有效降低模型的计算量, 且这两个前期工作皆采用步长为 2 的卷积神经网络对模型中间层的隐层表征进行降采样。为了进一步验证 DWT 信号压缩模块的优势与合理性, 对比 DWT 信号压缩模块与卷积神经网络的降采样性能。公平起见, 将 WLformer 中的 DWT 信号压缩模块-1 和 DWT 信号压缩模块-2 替换为步长为 2 的卷积神经网络, 其余部分完全保持一致。为进行更全面的比较, 对比所使用的降采样卷积神经网络采用两种超参数设置, 卷积核大小分别为 4 或 8。结果如表 5 所示, 使用 DWT 信号压缩模块的性能明显且稳定地好于使用降采样卷积神经网络的性能, 验证了其性能优势与设计的合理性。

为进一步验证 DWT 信号压缩模块的合理性, 对上述实验中使用的降采样卷积神经网络 (卷积核大小为 4) 与 DWT 信号压缩模块的输出进行可视化对比。不失一般性, 设 WLformer Group1 输出的隐层表征为 $\boldsymbol{R}$; 将 $\boldsymbol{R}$ 通过降采样卷积神经网络后的输出设为 $\boldsymbol{R}_{\mathrm{CNN}}$; 将 $\boldsymbol{R}$ 通过 DWT 信号压缩模块-1 得到的尺度系数设为 $\boldsymbol{c}_1$, 小波系数设为 $\boldsymbol{d}_1$。然后对 $\boldsymbol{R}$, $\boldsymbol{R}_{\mathrm{CNN}}$, $\boldsymbol{c}_1$, $\boldsymbol{d}_1$ 分别进行可视化, 如图 4 所示。对比 $\boldsymbol{R}$, $\boldsymbol{R}_{\mathrm{CNN}}$, $\boldsymbol{c}_1$ 三者的可视化结果可见, 虽然通过 DWT 信号压缩模块-1 获得的 c_1 的时序长度仅为 $\boldsymbol{R}$ 的一半, 但却保留了 $\boldsymbol{R}$ 中绝大部分的信息, 并且相较于使用降采样卷积神经网络所获得的 $\boldsymbol{R}_{\mathrm{CNN}}$ 有明显更小的失真, 这也说明了 DWT 信号压缩模块性能的优势。此外, 由 DWT 信号压缩模块中获得的表示高频成分的小波系数 $\boldsymbol{d}_1$ 的可视化结果可见, $\boldsymbol{d}_1$ 中蕴含的信息量明显

表 5 使用 DWT 信号压缩模块与降采样卷积神经网络性能对比

降采样方法	Aishell-1		HKUST	Librispeech	
	CER (%)		CER (%)	WER (%)	
	dev	test	test	test-clean	test-other
降采样卷积神经网络 (卷积核大小: 4)	5.0	5.5	22.6	3.2	7.5
降采样卷积神经网络 (卷积核大小: 8)	5.1	5.4	22.8	3.3	7.6
DWT信号压缩模块	**4.8**	**5.2**	**22.1**	**3.1**	**7.1**

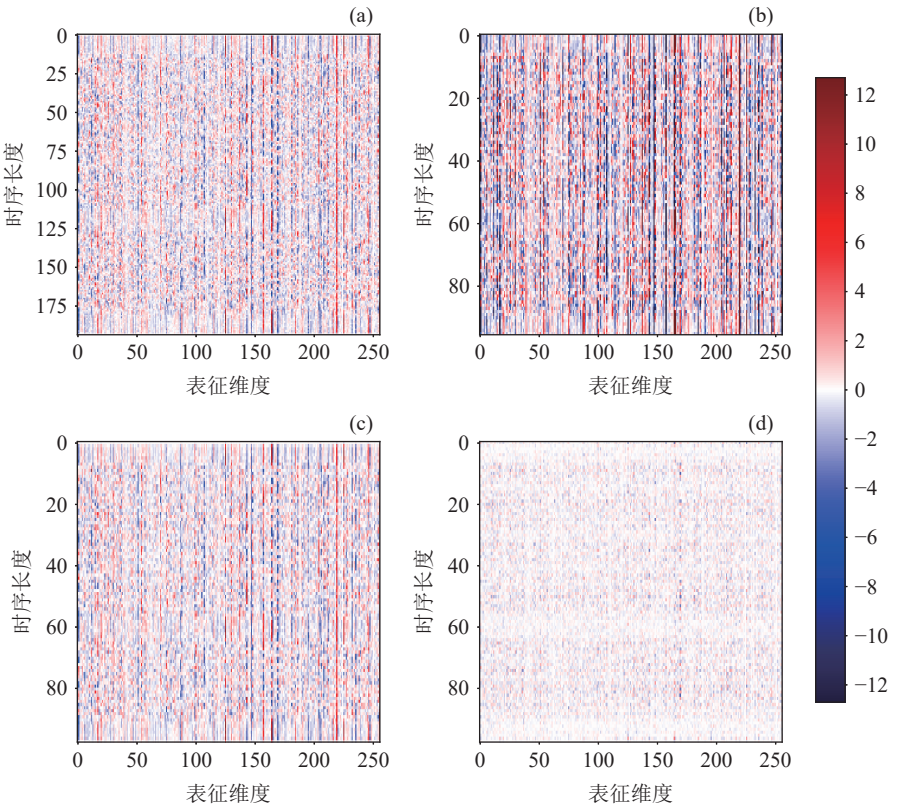

图 4 不同降采样方法输出表征的可视化 (a) WLformer Group1 输出的隐层表征 $\boldsymbol{R}$; (b) 将 $\boldsymbol{R}$ 通过降采样卷积神经网络后的输出 $\boldsymbol{R}_{\mathrm{CNN}}$; (c) 将 $\boldsymbol{R}$ 通过 DWT 信号压缩模块-1 得到的尺度系数 $\boldsymbol{c}_1$; (d) 将 $\boldsymbol{R}$ 通过 DWT 信号压缩模块-1 得到的小波系数 $\boldsymbol{d}_1$

相对较少, 这也印证了语音信号中主要信息大多集中在低频成分中, 而高频中语义相关信息较少的结论[16-18], 同时也证明了 DWT 信号压缩模块中将表示高频成分的小波系数去除以降低计算量的合理性。

进一步研究语音信号与模型的隐层表征之间高低频的关联性。将语音信号和隐层特征的高低频成分分别进行可视化, 如图 5 所示。语音信号与隐层特征高频成分的能量都相对较低, 且两者高频成分所包含的信息量也都明显少于各自的低频成分, 主要的能量与信息都集中于两者的低频成分中。因此, 语音信号与隐层表征中的高低频成分之间具有一致的能量与信息量分布特点。因此, 可认为语音信号内高频中语义相关信息量较少、而主要能量与信息量都集中于低频成分中的结论也适用于模型的隐层表征。

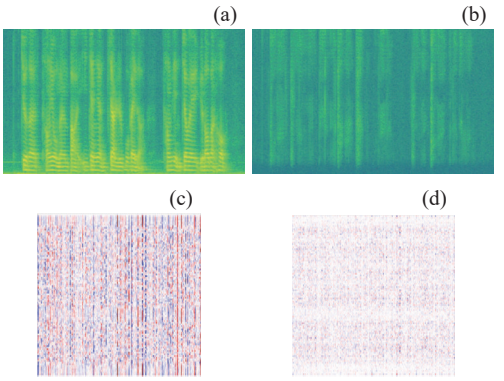

图 5　语音信号与隐层特征的高低频成分可视化 (a) 语音信号低频成分; (b) 语音信号高频成分; (c) 隐层特征低频成分; (d) 隐层特征高频成分

5.3　DSD-FFN 消融实验

为了验证所提 DSD-FFN 的合理性, 首先进行以下两种模型设置的性能对比: 全部 FFN 皆为标准 FFN, 以及 Group2 中使用 DSD-FFN (即 WLformer)。实验结果如表 6 所示, 将 Group2 中的 FFN 替换为 DSD-FFN, 可以在几乎不影响性能的情况下进一步将显存占用与 Gflops 分别相对降低 7.0% 与 7.2%, 这验证了 DSD-FFN 的合理性。

研究 DSD-FFN 所设置的位置对性能的影响。进行以下五种模型设置的性能对比: 仅 Group1 中使用 DSD-FFN, 仅 Group2 中使用 DSD-FFN, 仅 Group3 中使用 DSD-FFN, Group1 与 Group2 中皆使用 DSD-FFN, Group1、Group2 与 Group3 皆使用 DSD-FFN。此实验在 Aishell-1 数据集上进行, 实验结果如表 7 所示。在 Group1、Group2、Group3 中分别使用 DSD-FFN, 这三者之中, Group1 中使用 DSD-FFN 虽然可以带来最多的模型计算量降低, 但是其性能却有所下降, 这也验证了先前提及的在模型浅层时需从包含完整频带信息的序列中提取特征的设想; 而只在 Group2 中使用 DSD-FFN 可以保证在几乎不影响性能的前提下降低模型计算量; 在 Group3 中使用 DSD-FFN 不仅计算量降低最少, 并且对其性能的影响也是最大的, 该现象可能是由于在 Group3 之前已经经过了两次 DWT 信号压缩模块, 信息的冗余度较低, 因此不再适合高低频解耦操作。因此, 此实验结果也证明了 WLformer 中对于 DSD-FFN 位置设置的合理性。

5.4　WLformer 中使用不同小波基的消融实验

上述实验中 DWT 信号压缩模块与 DSD-FFN 模块中的小波基皆采用信号压缩中常用的 Db4 小波基。不同的小波基对应的滤波器系数不同, 一般而言, 某个任务对于最优小波基的选择较难有标准的

表 6　不同 FFN 设置的性能对比

FFN设置	Aishell-1		HKUST	Librispeech		显存占用 (GB)	Gflops
	CER (%)		CER (%)	WER (%)			
	dev	test	test	test-clean	test-other		
全部为标准FFN	**4.8**	**5.2**	**22.1**	**3.0**	**7.1**	0.86	27.6
WLformer	**4.8**	**5.2**	**22.1**	3.1	**7.1**	**0.80**	**25.6**

表 7　DSD-FFN 的位置对性能的影响

使用DSD-FFN的Group	CER (%)		显存占用 (GB)	Gflops
	dev	test		
Group1	5.0	5.3	0.78	24.7
Group2	**4.8**	**5.2**	0.80	25.6
Group3	5.2	5.6	0.82	26.4
Group1 + Group2	5.1	5.4	0.73	22.7
Group1 + Group2 + Group3	5.2	5.8	**0.70**	**21.5**

表 8 各小波基对应的滤波器系数

小波基	调整滤波器h	小波滤波器g	调整重构滤波器hr	小波重构滤波器gr
Db2	h0 = −0.1294095226 h1 = 0.2241438680 h2 = 0.8365163037 h3 = 0.4829629131	g0 = −0.4829629131 g1 = 0.8365163037 g2 = −0.2241438680 g3 = −0.1294095226	hr0 = 0.4829629131 hr1 = 0.8365163037 hr2 = 0.2241438680 hr3 = −0.1294095226	gr0 = −0.1294095226 gr1 = −0.2241438680 gr2 = 0.8365163037 gr3 = −0.4829629131
Db4	h0 = −0.0105974018 h1 = 0.0328830117 h2 = 0.0308413818 h3 = −0.1870348117 h4 = −0.0279837694 h5 = 0.6308807679 h6 = 0.7148465706 h7 = 0.2303778133	g0 = −0.2303778133 g1 = 0.7148465706 g2 = −0.6308807679 g3 = −0.0279837694 g4 = 0.1870348117 g5 = 0.0308413818 g6 = −0.0328830117 g7 = −0.0105974018	hr0 = 0.2303778133 hr1 = 0.7148465706 hr2 = 0.6308807679 hr3 = −0.0279837694 hr4 = −0.1870348117 hr5 = 0.0308413818 hr6 = 0.0328830117 hr7 = −0.0105974018	gr0 = −0.0105974018 gr1 = −0.0328830117 gr2 = 0.0308413818 gr3 = 0.1870348117 gr4 = −0.0279837694 gr5 = −0.6308807679 gr6 = 0.7148465706 gr7 = −0.2303778133
Coif1	h0 = −0.0156557281 h1 = −0.0727326195 h2 = 0.3848648469 h3 = 0.8525720202 h4 = 0.3378976625 h5 = −0.0727326195	g0 = 0.0727326195 g1 = 0.3378976625 g2 = −0.8525720202 g3 = 0.3848648469 g4 = 0.0727326195 g5 = −0.0156557281	hr0 = −0.0727326195 hr1 = 0.3378976625 hr2 = 0.8525720202 hr3 = 0.3848648469 hr4 = −0.0727326195 hr5 = −0.0156557281	gr0 = −0.0156557281 gr1 = 0.0727326195 gr2 = 0.3848648469 gr3 = −0.8525720202 gr4 = 0.3378976625 gr5 = 0.0727326195
Bior3.3	h0 = 0.0662912607 h1 = −0.1988737822 h2 = −0.1546796084 h3 = 0.9943689110 h4 = 0.9943689110 h5 = −0.1546796084 h6 = −0.1988737822 h7 = 0.0662912607	g0 = 0 g1 = 0 g2 = −0.1767766953 g3 = 0.5303300859 g4 = −0.5303300859 g5 = 0.1767766953 g6 = 0 g7 = 0	hr0 = 0 hr1 = 0 hr2 = 0.1767766953 hr3 = 0.5303300859 hr4 = 0.5303300859 hr5 = 0.1767766953 hr6 = 0 hr7 = 0	gr0 = 0.0662912607 gr1 = 0.1988737822 gr2 = −0.1546796084 gr3 = −0.9943689110 gr4 = 0.9943689110 gr5 = 0.1546796084 gr6 = −0.1988737822 gr7 = −0.0662912607

表 9 不同小波基之间的性能对比

不同小波基	Aishell-1		HKUST	Librispeech	
	CER (%)		CER (%)	WER (%)	
	dev	test	test	test-clean	test-other
Db2	**4.8**	**5.2**	22.2	**3.1**	**7.1**
Db4	**4.8**	**5.2**	**22.1**	**3.1**	**7.1**
Coif1	4.9	**5.2**	22.2	3.2	**7.1**
Bior3.3	4.9	5.3	22.3	**3.1**	7.2

定量方法, 往往需要结合经验与实验结果进行选择[28,29]。使用四种不同的常用小波基: Db2、Db4、Coif1、Bior3.3, 研究不同的小波基对 WLformer 性能的影响。这四种小波基对应的滤波器系数如表 8 所示, 使用四种小波基的实验结果如表 9 所示。Db4 小波基可以取得相对更好的性能, 但这四种不同小波基之间的性能差异很小, 因此总体来说 WLformer 对于小波基的选择并不敏感。

5.5 WLformer 增强性能的原因讨论

将 Conformer 每个 block 的隐层表征中每两个相邻隐层表征帧之间的平均余弦相似度可视化, 如图 6 黑色曲线所示。原 Conformer 模型中相邻隐层表征帧之间的平均余弦相似度非常高, 最高达 0.9 以上, 说明原 Conformer 模型的隐层表征中存在大量的冗余信息, 这也导致模型在训练中出现非常多的重复信息, 进而造成模型性能下降, 且模型推理时也存在大量重复且冗余的计算。将 Efficient Conformer、Squeezeformer、WLformer 每个 block 的隐层表征中

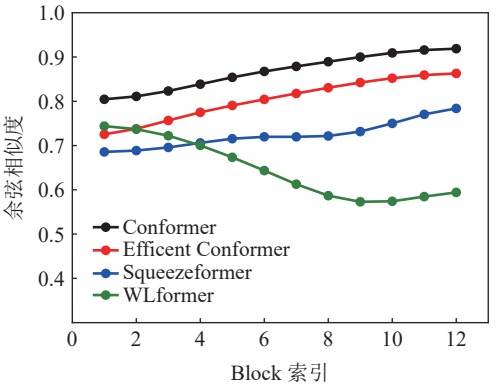

图 6 不同模型每两个相邻隐层表征帧之间的平均余弦相似度

每两个相邻隐层表征帧之间的平均余弦相似度分别可视化, Efficient Conformer 与 Squeezeformer 也都通过在模型中间层使用 CNN 进行降采样的方式降低了信息的冗余度, 结合表 2 的结果可知, 这两个模型在降低信息冗余度的同时性能得到提升。此外, 由图 6 可知, WLformer 较其他模型信息冗余明显降低更多, 验证了 WLformer 在特征提取方面的优越性。

6 结论

为了解决端到端语音识别模型计算资源占用过高的问题, 提出了一种将离散小波变换与深度学习相结合的方法 WLformer, 在降低计算资源占用量的同时还提升了识别的性能。WLformer 旨在利用离散小波变换将模型的表征分解为低频与高频两种成分, 并根据低频与高频成分中的信息量差异对两者进行差异化操作, 从而降低模型的计算量。实验结果表明, 相较于基线系统与对比方法, WLformer 在多个数据集上都以更低的计算资源占用取得了更优的识别性能。因此, WLformer 是一种有效的低计算资源占用的端到端语音识别方法。

参　考　文　献

1 Li K, Li J, Ye G, *et al*. Towards code-switching ASR for end-to-end CTC models. IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, Brighton, 2019: 6076−6080

2 刘育坤, 郑霖, 黎塔, 等. 多声学场景下端到端语音识别声学编码器的自适应. 声学学报, 2023; **48**(6): 1260−1268

3 高长丰, 程高峰, 张鹏远. 面向鲁棒自动语音识别的一致性自监督学习方法. 声学学报, 2023; **48**(3): 578−587

4 Deng K, Woodland P C. Adaptable end-to-end ASR models using replaceable internal LMs and residual softmax. IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, Greece, 2023: 1−5

5 Dong L, Xu S, Xu B. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, Alberta, 2018: 5884−5888

6 Gulati A, Qin J, Chiu C C, *et al*. Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint: 200508100, 2020

7 Peng Y, Dalmia S, Lane I, *et al*. Branchformer: Parallel MLP-attention architectures to capture local and global context for speech recognition and understanding. International Conference on Machine Learning, PMLR, Hawaii, 2022: 17627−17643

8 Kim K, Wu F, Peng Y, *et al*. E-branchformer: Branchformer with enhanced merging for speech recognition. IEEE Spoken Language Technology Workshop, IEEE, Qatar, 2023: 84−91

9 Burchi M, Vielzeuf V. Efficient conformer: Progressive downsampling and grouped attention for automatic speech recognition. IEEE Automatic Speech Recognition and Understanding Workshop, IEEE, Cartagena, 2021: 8−15

10 Kim S, Gholami A, Shaw A, *et al*. Squeezeformer: An efficient transformer for automatic speech recognition. Neural Information Processing Systems, NIPS Foundation, New Orleans, 2022: 9361−9373

11 Pang R, Sainath T N, Prabhavalkar R, *et al*. Compression of end-to-end models. Interspeech, ISCA, Hyderabad, 2018: 27−31

12 Tian S, Li Z, Lyv Z, *et al*. Factorized and progressive knowledge distillation for CTC-based ASR models. *Speech Commun.*, 2024; **160**: 103071

13 Bekal D, Gopalakrishnan K, Mundnich K, *et al*. A metric-driven approach to conformer layer pruning for efficient ASR inference. Interspeech, ISCA, Dublin, 2023: 2958−1796

14 Yoon J W, Lee H, Kim H Y, *et al*. TutorNet: Towards flexible knowledge distillation for end-to-end speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.*, 2021; **29**: 1626−1638

15 Yao Z, Guo L, Yang X, *et al*. Zipformer: A faster and better encoder for automatic speech recognition. International Conference on Learning Representations, Vienna, 2024

16 Mavaddaty S, Ahadi S M, Seyedin S. Speech enhancement using sparse dictionary learning in wavelet packet transform domain. *Comput. Speech Lang.*, 2017; **44**: 22−47

17 李如玮, 鲍长春, 窦慧晶. 基于小波变换的语音增强算法综述. 数据采集与处理, 2009; **24**(3): 362−368

18 Daqrouq K, Abu-Isbeih I N, Daoud O, *et al*. An investigation of speech enhancement using wavelet filtering method. *Int. J. Speech Technol.*, 2010; **13**: 101−115

19 潘泉, 张磊, 孟晋丽, 等. 小波滤波方法及应用. 北京: 清华大学出版社, 2005: 42−43

20 Lin T, Wang Y, Liu X, *et al*. A survey of transformers. *AI Open*, 2022; **3**: 111−132

21 Bu H, Du J, Na X, *et al*. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. International Coordinating Committee on Speech Databases and Speech I/O systems and assessment (O-COCOSDA), IEEE, Seoul, 2017: 1−5

22 Liu Y, Fung P, Yang Y, *et al*. HKUST/MTS: A very large scale mandarin telephone speech corpus. Chinese Spoken Language Processing, ISCA SIG-CSLP, Springer, 2006: 724−735

23 Panayotov V, Chen G, Povey D, *et al*. Librispeech: An ASR corpus based on public domain audio books. IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, Queensland, 2015: 5206−5210

24 Xiang S, Liang Q, Fang L. Discrete wavelet transform-based Gaussian mixture model for remote sensing image compression. *IEEE Trans. Geosci. Remote Sens.*, 2023; **61**: 3000112

25 Tian C, Zheng M, Zuo W, *et al*. Multi-stage image denoising with the wavelet transform. *Pattern Recogn.*, 2023; **134**: 109050

26 Paszke A, Gross S, Massa F, *et al*. Pytorch: An imperative style, high-performance deep learning library. Neural Information Processing Systems, NIPS Foundation, Vancouver, 2019: 32

27 Watanabe S, Hori T, Karita S, *et al*. Espnet: End-to-end speech processing toolkit. arXiv preprint: 180400015, 2018

28 王瑞琳, 王立, 贺盈波. 基于小波和动态互补滤波的图像与事件融合方法. 工程科学学报, 2024; **46**(11): 2076−2084

29 Liu C, Chen W, Zhang T. Wavelet-Hilbert transform based bidirectional least squares grey transform and modified binary grey wolf optimization for the identification of epileptic EEGs. *Biocybern. Biomed. Eng.*, 2023; **43**(2): 442−462