# A Lightweight Cross-Domain Front-End Feature Extractor for Multichannel Voice Activity and Overlapped Speech Detection

1st Shaojie Li
*College of Computer Science*
*Inner Mongolia University*
Hohhot, China
lishaojie@mail.imu.edu.cn

2nd Qintuya Si
*College of Electronic and Information Engineering*
*Inner Mongolia University*
Hohhot, China
siqty@imu.edu.cn

3rd De Hu*
*College of Computer Science*
*Inner Mongolia University*
Hohhot, China
cshood@imu.edu.cn

***Abstract*—Although recent advances using microphone arrays have shown impressive voice activity detection (VAD) or overlapped speech detection (OSD) performance, their network architectures often incur high computational costs (especially as the number of microphones increases). In this work, by developing a lightweight cross-domain feature extractor (L-CD-FE), we propose a novel joint approach for VAD and OSD. Recognizing the unique contributions of time-domain (TD) or time-frequency (TF) representations, the L-CD-FE integrates TD and TF features through a bidirectional cross-domain fusion module. Here, TD and TF features are obtained from the weighted sum of multichannel TD and TF representations using a lightweight channel aggregation (CA) module. Finally, the L-CD-FE is cascaded with the existing sequence modeling architecture to jointly achieve VAD and OSD. Numerical experiments show that the proposed method provides comparable VAD and OSD performance with state of the arts, but it shows a remarkable superiority in terms of computational efficiency.**

***Index Terms*—voice activity detection, overlapped speech detection, cross-domain fusion, channel aggregation.**

## I. Introduction

Voice activity detection (VAD) and overlapped speech detection (OSD) are important pre-processing tasks in many acoustic applications [1]–[3], in which the former detects speech segments in audio streams while the latter detects segments containing at least two simultaneously active speakers.

Early studies [4]–[6] on VAD were based on statistical modeling of acoustic features. With the development of deep learning, some sequence modeling networks were applied to VAD, such as long-short time memory (LSTM) [7] and convolutional neural networks (CNN) [8]. Similarly, the most recent OSD approaches were also based on deep neural networks, for example, LSTM neural networks were applied to OSD in [9]. Alternatively, some other works were based on the temporal convolutional network (TCN) [10], [11]. The above-mentioned researches focus on single-channel speech signals, which is often applied to close-talk scenarios where the speaker is close to the microphone. In a more general acoustic scene, e.g., speech signal is recorded by a distant device, the microphone array is usually used to capture the scene [12].

For now, few studies were conducted on distant multichannel VAD and OSD [12]–[15]. Specifically, Cornell *et al.* [13] randomly selected the acoustic features of one channel from a multichannel signal as model input. To utilize the implicit information in the multichannel signals, some researchers [14], [15] explored different spatial features based on handcrafted design and demonstrated that they are beneficial to improve VAD and OSD performance. However, those approaches are limited by the number of microphones and the microphone array configuration. Recent advances focused on designing a learnable front-end feature extractor to weight and sum multichannel signals as single-channel representations, which can be optimized together with sequence modeling networks. In [16], Gong *et al.* proposed a self-attention channel combination (SACC) algorithm to compute combination weights, which is based on the self-attention mechanism [17] module and time-frequency (TF) features (e.g., Short-Time Fourier Transform (STFT) magnitude). Subsequently, Mariotte *et al.* [12] used SACC as the front-end feature extractor for multichannel VAD and OSD, and proposed several variant methods based on SACC that utilize STFT phase information, including Explicit cSACC (EcSACC) and Implicit cSACC (IcSACC). Although the recent methods show excellent performance, they suffer from significant computational costs when fusing multichannel signals into a single-channel representation. In addition, they often only employ the TF features (e.g., STFT magnitude or phase), ignoring the contribution of time-domain (TD) representations.

To this end, we present a lightweight cross-domain feature extractor (L-CD-FE) for multichannel VAD and OSD, which extracts deep features from time-domain (TD) or time-frequency (TF) representations. Unlike self-attention-based methods, we employ a lightweight channel aggregation (CA) module to weight and combine multichannel TD or TF representations, which involves a simple $\ell_1$ normalization without

* Corresponding author. This work was supported in part by the National Natural Science Foundation of China under Grants 62361045 and 62201297; and in part by the fund of Supporting the Reform and Development of Local Universities (Disciplinary Construction) and the special research project of First-class Discipline of Inner Mongolia A. R. of China under Grant YLXKZX-ND-036.

introducing excessive parameters and computational costs. Then, we propose a bidirectional cross-domain fusion module to efficiently integrate information from both the TD and TF. The proposed L-CD-FE is compared with the state-of-the-art SACC-based methods on the AMI meeting corpus [18], and the results show that the system with proposed L-CD-FE can achieve lightweight and efficient multichannel VAD and OSD.

## II. Problem Formulation

VAD+OSD can be formulated as a three-category classification task: non-speech ($n_{spk}$=0), single speech ($n_{spk}$=1), and overlapped speech ($n_{spk}\geq$2) with $n_{spk}$ being the number of active speakers. Fig. 1 shows the classical flowchart of multichannel VAD+OSD. First, multichannel audio signals $\mathbf{X} \in \mathbb{R}^{C\times N}$ with $C$ channels and $N$ samples are fed into the feature extractor to output single-channel frame-wise representations $\mathbf{X}' = [\mathbf{x}'_1, ..., \mathbf{x}'_t, ...\mathbf{x}'_T] \in \mathbb{R}^{E\times T}$, where $t$ the time frame index, $E$ and $T$ represent the numbers of features and frames, respectively. Then, VAD+OSD can be implemented by using sequence modeling network to produce the prediction sequence $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, ..., \hat{\mathbf{y}}_t, ..., \hat{\mathbf{y}}_T] \in \mathbb{R}^{3\times T}$, where $\hat{\mathbf{y}}_t = [p(n_{spk}=0|\mathbf{x}'_t), p(n_{spk}=1|\mathbf{x}'_t), p(n_{spk}\geq 2|\mathbf{x}'_t)]^T$ denotes the probabilities of each class at the $t$-th frame. Finally, VAD can be solved by combining the last two probabilities outputs, i.e. $p(n_{spk}=1|\mathbf{x}'_t)$+$p(n_{spk}\geq 2|\mathbf{x}'_t)$, while OSD is inferred from $p(n_{spk}\geq 2|\mathbf{x}'_t)$.

The sequence modeling network using LSTM or TCN performs well for multichannel VAD+OSD tasks [12]–[14], and the state-of-the-art feature extractor [12] using cross-channel attention outperforms handcrafted schemes in accuracy. However, the latter brings substantial computational overhead, especially as the number of channels increases.

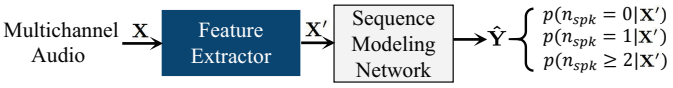


Fig. 1. Simplified flowchart for multichannel VAD+OSD.

## III. Lightweight cross-domain feature extractor

Recognizing the unique contributions of time-domain (TD) or time-frequency (TF) representations [19], we propose a lightweight cross-domain feature extractor (L-CD-FE), which integrates the TD feature and the TF feature in a parallel architecture, as shown in Fig. 2. The TD feature is extracted from frame-processed multichannel raw speech waveform, while the TF feature is extracted from multichannel STFT magnitudes. There are three major parts: channel aggregation (CA), an encoder, and bidirectional cross-domain fusion (B-CDF). To be specific, CA integrates the multichannel (TD or TF) features into the single-channel one by weighting and summing operations in a lightweight manner. The encoder extracts the deep TD features, which consists of simple convolution, activation function, and normalization layer. The B-CDF fuses the TD feature $\mathbf{O}_F$ and the TF feature $\mathbf{O}_A$ into a cross-modal representation $\mathbf{X}'$, based on the cross-attention between the two modal features.

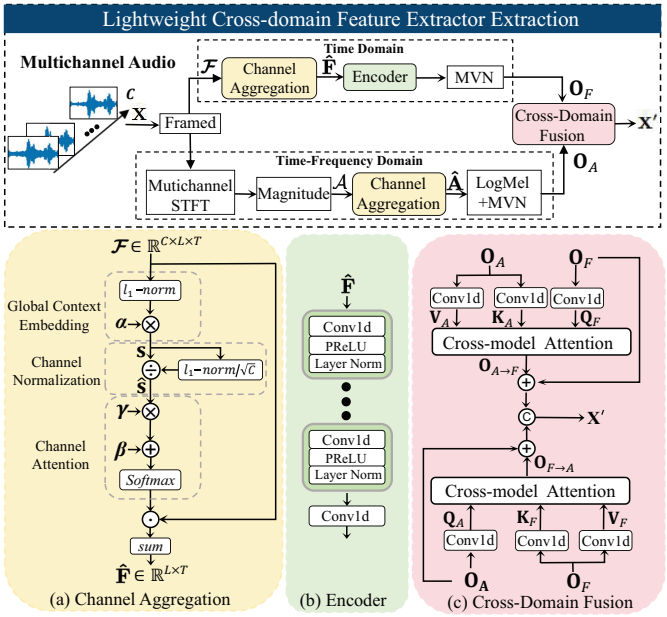


Fig. 2. A Lightweight cross-domain feature extractor (L-CD-FE) architecture.

### A. Channel Aggregation (CA)

To mitigate computational burden, inspired by [20], we design a lightweight channel aggregation (CA) module to integrate multichannel signals. Let $\Psi(\cdot)$ denote the mapping of the proposed CA module. In the time domain, the integrated TD feature $\hat{\mathbf{F}} \in \mathbb{R}^{L\times T}$ can be obtained by $\hat{\mathbf{F}} = \Psi(\boldsymbol{\mathcal{F}})$, where $\boldsymbol{\mathcal{F}} \in \mathbb{R}^{C\times L\times T}$ is the TD signal with $L$ being the frame length. Similarly, in the TF domain, the integrated TF feature $\hat{\mathbf{A}} \in \mathbb{R}^{M\times T}$ can be obtained by $\hat{\mathbf{A}} = \Psi(\boldsymbol{\mathcal{A}})$, where $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{C\times M\times T}$ is the STFT magnitude with $M$ being the number of frequencies.

The CA module $\Psi(\cdot)$ includes three key components: global context embedding, channel normalization, and channel attention, as shown in Fig. 2 (a). Here, the detail of $\Psi(\cdot)$ is formulated in the time domain, for example, by defining $\boldsymbol{\mathcal{F}} = [\mathbf{F}_1, \mathbf{F}_2, ..., \mathbf{F}_C]$, where $\mathbf{F}_c = [\boldsymbol{f}_{c,1}, \boldsymbol{f}_{c,2}, ..., \boldsymbol{f}_{c,T}] \in \mathbb{R}^{L\times T}$ is the TD signal of the $c$-th channel with $\boldsymbol{f}_{c,t} = [f^i_{c,t}] \in \mathbb{R}^L$, and $c \in \{1, 2, ..., C\}$ and $t \in \{1, 2, ..., T\}$.

**Global Context Embedding**: Global information can capture a larger receptive field, avoiding local ambiguities [21]. As frame-level prediction is required in multichannel VAD and OSD, we first use a global context embedding module to aggregate the global context information in each frame for each channel. Given the channel embedding weight $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_C]$, the global context embedding can be modeled as

$$s_{c,t} = \alpha_c ||\boldsymbol{f}_{c,t}||_1 = \alpha_c \left\{ \left[ \sum_{i=1}^{L} |f^i_{c,t}| + \epsilon \right] \right\} , \qquad (1)$$

where $\epsilon$ is a small constant, $||\cdot||_1$ denotes $\ell_1$-norm, and $||\boldsymbol{f}_{c,t}||_1$ is the sum of amplitudes within frame. As different channels

have different significance, we use the learnable parameter $\boldsymbol{\alpha}$ to control the channel weights.

**Channel Normalization**: Normalization methods can stabilize training performance with lightweight computing resource [20], [22]. We use an $\ell_1$ normalization to operate across channels, obtaining

$$\hat{\boldsymbol{s}}_t = \frac{\boldsymbol{s}_t}{\epsilon + ||\boldsymbol{s}_t||_1} = \frac{\boldsymbol{s}_t}{\epsilon + \sum_{c=1}^{C} |s_{c,t}|}, \tag{2}$$

where $\boldsymbol{s}_t = [s_{1,t}, s_{2,t}, ..., s_{C,t}]^{\mathrm{T}}$ and $\hat{\boldsymbol{s}}_t = [\hat{s}_{1,t}, \hat{s}_{2,t}, ..., \hat{s}_{C,t}]^{\mathrm{T}}$.

**Channel Attention**: We adopt a channel attention mechanism to assign a weight for each channel, i.e.,

$$\boldsymbol{w}_t = \mathrm{softmax}(\boldsymbol{\gamma} \odot \hat{\boldsymbol{s}}_t + \boldsymbol{\beta}), \tag{3}$$

where $\boldsymbol{w_t}$ is the weight of channels at frame $t$, $\boldsymbol{\gamma} = [\gamma_1, ..., \gamma_C]$ and $\boldsymbol{\beta} = [\beta_1, ..., \beta_C]$ are learnable weights and biases, respectively, and $\odot$ represents the element-wise product. Then, the integrated TD feature $\hat{\mathbf{F}}$ can be computed by

$$\hat{\boldsymbol{f}}_t = \sum_{c=1}^{C} \boldsymbol{f}_{c,t} \odot w_{t,c}, \tag{4}$$

$$\hat{\mathbf{F}} = [\hat{\boldsymbol{f}}_1, \hat{\boldsymbol{f}}_2, , ..., \hat{\boldsymbol{f}}_T], \tag{5}$$

where $w_{t,c}$ is the $c$-th element of $\boldsymbol{w}_t$.

### B. Encoder

The integrated TD feature $\hat{\mathbf{F}}$, obtained from the CA module in the upper branch in Fig. 2, is further refined by an encoder with Mean and Variance Normalization (MVN), generating the final TD feature $\mathbf{O}_F \in \mathbb{R}^{D \times T}$, where $D$ is the dimension after refinement. In detail, as depicted in Fig. 2 (b), the encoder consists of a stack of $K$ basic layers, each containing a Conv1D layer, a PReLU activation, and a LayerNorm operation, followed by a Conv1D layer that controls the dimension of output features.

Alternatively, the integrated TF feature $\hat{\mathbf{A}}$, obtained from the CA module in the lower branch in Fig. 2, is further refined through well-known log-mel filters with MVN, generating the final TF feature $\mathbf{O}_A \in \mathbb{R}^{D \times T}$.

### C. Cross Fusion Module

To effectively fuse the final TD feature $\mathbf{O}_F$ and TF feature $\mathbf{O}_A$, we propose a novel bidirectional cross-domain fusion (B-CDF) module, as shown in Fig. 2 (c). The design of the B-CDF is based on a cross-attention mechanism. Thus, the query vectors ($\mathbf{Q}_F$,$\mathbf{Q}_A$), key vectors ($\mathbf{K}_F$,$\mathbf{K}_A$) and value vectors ($\mathbf{V}_F$,$\mathbf{V}_A$) can be obtained from the embeddings of $\mathbf{O}_F$ and $\mathbf{O}_A$, respectively, through 1D convolutional layer mapping. Then, the TD attention feature $\mathbf{O}_{F \to A}$ and TF attention feature $\mathbf{O}_{A \to F}$ can be computed respectively by

$$\mathbf{O}_{F \to A} = \mathrm{softmax}(\frac{\mathbf{Q}_A \mathbf{K}_F^T}{\sqrt{D}})\mathbf{V}_F + \mathbf{O}_A, \tag{6}$$

$$\mathbf{O}_{A \to F} = \mathrm{softmax}(\frac{\mathbf{Q}_F \mathbf{K}_A^T}{\sqrt{D}})\mathbf{V}_A + \mathbf{O}_F, \tag{7}$$

where the feature dimension of $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ is $D$. Finally, the fused feature $\mathbf{X}'$ is generated by concatenating $\mathbf{O}_{F \to A}$ and $\mathbf{O}_{A \to F}$ along the first dimension, i.e.,

$$\mathbf{X}' = \mathrm{concat}(\mathbf{O}_{F \to A}, \mathbf{O}_{A \to F}). \tag{8}$$

After fusing TD feature and TF feature, $\mathbf{X}'$ provides complementary information from both domains for the sequence modeling network.

## IV. Experiments

### A. Dataset

Experiments were conducted on the AMI meeting corpus [18], which contains 100 hours of realistic meeting recordings. We adopted the AMI $Array1$ data that was captured by an 8-microphone circular array placed in the center of the table. Training, Development (Dev), and Evaluation (Eval) partitions followed the protocol proposed in [23]. During the training phase, ground truth was generated via Forced-Alignment [13]. The results on the Dev and Eval sets were evaluated using the official annotation.

### B. Sequence Modeling Networks

To evaluate the robustness of the proposed feature extractor, we considered two well-known sequence model architectures used in VAD and OSD: namely the Bidirectional LSTM (BLSTM) and the TCN.

**BLSTM**: This sequence modeling architecture was composed of a single BLSTM layer with 256 cells, which was connected to a three-layer feed-forward network (FFN) for post-processing. Note that the output sizes of FFN layers were $L_1 = 128$, $L_2 = 128$, and $L_3 = 3$ respectively, and the first two FFN layers were followed by a PReLu activation function.

**TCN**: The adopted TCN consisted of 1D convolutional layers with exponentially increasing dilation rates to efficiently capture long-range temporal dependencies. We utilized 5 residual convolutional blocks replicated 3 times. This above TCN architecture was identical to [12, Sec.III-B].

### C. Baselines

The proposed L-CD-FE was compared with state-of-the-art learnable front-end feature extractors [12], including SACC, EcSACC and IcSACC. Consistent with the experimental steps in [12], all these feature extractors output a 64-dimensional feature in each frame. For L-CD-FE, the encoder stacked $K = 3$ basic layers. The dimension of the TD feature $\mathbf{O}_A$ and the TF feature $\mathbf{O}_F$ was fixed to $D$ = 64, outputting $E = 128$ dimensional cross-domain feature in each frame.

### D. Training and Evaluation Setup

The LSTM and TCN were trained using 3 seconds of audio segments randomly sampled from the training set. We used ADAM [24] with a mini-batch size of 32. The frame length was equal to 25 ms with 60% overlap. All experiments were executed on an NVIDIA L40S GPU. VAD performance was evaluated using the false alarm rate (FA), the miss detection rate (Miss), and the segmentation error rate (SER),

TABLE I

VAD AND OSD PERFORMANCE, WHICH ARE OBTAINED ON AMI DEVELOPMENT (DEV) AND EVALUATION (EVAL) SETS. BOLD VALUES INDICATE THE BEST-PERFORMING RESULTS, AND † DENOTES RE-IMPLEMENTATION IN OUR EXPERIMENTAL SETUP. THE FLOATING-POINT OPERATIONS (FLOPS) ARE PERFORMED ON THE INPUT SIGNAL $\mathbf{X} \in \mathbb{R}^{8\times 400}$.

| Seq. Mod. | Fea. Ext. | Param. | FLOPs | VAD | | | | | | OSD | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | **FA ↓** | | **Miss ↓** | | **SER ↓** | | **Pr↑** | | **Re↑** | | **F1↑** | |
| | | | | Dev | Eval | Dev | Eval | Dev | Eval | Dev | Eval | Dev | Eval | Dev | Eval |
| BLSTM | SACC† | 0.87M$_{+0.00\%}$ | 2.30M$_{+0.00\%}$ | **15.65** | **12.91** | 1.93 | 2.75 | **17.58**$_{+0.00\%}$ | **15.66**$_{+0.00\%}$ | **60.69** | 57.70 | 76.73 | **69.51** | 67.78$_{+0.00\%}$ | **63.06**$_{+0.00\%}$ |
| | EcSACC† | 1.00M$_{+14.94\%}$ | 4.48M$_{+94.78\%}$ | 16.00 | 13.42 | 1.88 | 2.69 | 17.88$_{+0.30\%}$ | 16.11$_{+0.45\%}$ | 56.30 | 53.34 | 70.10 | 62.23 | 62.45$_{-5.33\%}$ | 57.52$_{-5.54\%}$ |
| | IcSACC† | 0.87M$_{+0.00\%}$ | 4.55M$_{+97.83\%}$ | 16.47 | 13.36 | 2.11 | 3.12 | 18.58$_{+1.00\%}$ | 16.48$_{+0.82\%}$ | 58.87 | 56.93 | 70.09 | 57.08 | 64.00$_{-3.78\%}$ | 57.01$_{-6.05\%}$ |
| | L-CD-FE | **0.75M**$_{-13.79\%}$ | **0.23M**$_{-90.00\%}$ | 15.97 | 13.30 | **1.74** | **2.43** | 17.71$_{+0.13\%}$ | 15.73$_{+0.07\%}$ | 60.10 | **58.89** | **78.37** | 67.83 | **68.03**$_{+0.25\%}$ | 63.04$_{-0.02\%}$ |
| TCN | SACC† | 0.40M$_{+0.00\%}$ | 2.69M$_{+0.00\%}$ | **14.08** | **11.94** | 1.81 | 2.46 | **15.89**$_{+0.00\%}$ | **14.40**$_{+0.00\%}$ | **66.62** | 65.14 | **81.47** | **74.59** | **73.33**$_{+0.00\%}$ | 69.55$_{+0.00\%}$ |
| | EcSACC† | 0.53M$_{+32.50\%}$ | 4.83M$_{+79.55\%}$ | 14.76 | 12.06 | 2.13 | 2.89 | 16.89$_{+1.00\%}$ | 14.95$_{+0.55\%}$ | 63.26 | 60.40 | 77.74 | 69.60 | 69.76$_{-3.57\%}$ | 64.68$_{-4.87\%}$ |
| | IcSACC† | 0.40M$_{+0.00\%}$ | 4.89M$_{+81.78\%}$ | 15.45 | 13.16 | 2.54 | 2.83 | 17.99$_{+2.10\%}$ | 15.99$_{+1.59\%}$ | 59.29 | 58.66 | 77.68 | 66.95 | 67.25$_{-6.08\%}$ | 62.53$_{-7.02\%}$ |
| | L-CD-FE | **0.30M**$_{-25.00\%}$ | **0.64M**$_{-76.21\%}$ | 14.96 | 12.08 | **1.78** | **2.45** | 16.74$_{+0.85\%}$ | 14.53$_{+0.13\%}$ | 66.49 | **66.65** | 80.77 | 73.39 | 72.94$_{-0.39\%}$ | **69.86**$_{+0.31\%}$ |

where $\mathrm{SER} = \mathrm{FA} + \mathrm{Miss}$. OSD performance was evaluated using precision (Pr), recall (Re), and F1-score (F1), where $\mathrm{F1} = 2 \cdot (\mathrm{Pr} \cdot \mathrm{Re})/(\mathrm{Pr} + \mathrm{Re})$.

## V. RESULTS AND DISCUSSION

In this section, we tested the VAD and OSD performance of both BLSTM and TCN architectures using different feature extractors. We can see from Table I that the proposed L-CD-FE shows competitive VAD and OSD performance with the smallest complexity.

### A. BLSTM Results

The BLSTM with L-CD-FE demonstrates significant advantages in terms of parameter count (0.75M) and FLOPs (0.23M). For example, its FLOPs are 90% lower than those of the second-best BLSTM using SACC. For the VAD task, the SERs of the BLSTM using L-CD-FE are 17.71% and 15.73% on the Dev and Eval sets, respectively. These values are slightly higher than those of the BLSTM using SACC (by approximately 0.1%) but lower than those of EcSACC or IcSACC. For the OSD task, the BLSTM with L-CD-FE achieves the best F1 score on the Dev set and the second-best F1 score on the Eval set.

### B. TCN Results

Overall, the TCN provides better VAD and OSD performance than the BLSTM when using all kinds of feature extractors. The TCN with L-CD-FE provides the smallest number of parameters and FLOPs. Specifically, its FLOPs are 0.64M, which are 76.21% lower than that of the second-best TCN using SACC. In addition, the TCN using L-CD-FE outperforms the TCN using EcSACC or IcSACC in terms of VAD and OSD performance. Moreover, using L-CD-FE achieves OSD performance comparable to that of using SACC, while the latter provides better VAD performance.

## VI. ABLATION EXPERIMENTS

In this section, we performed ablation experiments on the AMI Eval set to verify the effectiveness of the proposed feature extractor, where the TCN was adopted to act as the sequence modeling network.

### A. Contributions from Different Domains

In this part, we tested VAD and OSD performance by extracting the TD feature, the TF feature, and the cross-domain feature. Note that the TD (or TF) feature was extracted after removing the lower (or upper) branch in the proposed L-CD-FE architecture. As shown in Table II, the TD feature shows the worst performance, with a VAD SER of 17.8% and an OSD F1-score of 61.58%. The TF feature achieves better performance compared to the TD feature, reducing VAD SER by 3.21% and increasing the OSD F1-score by 7.5%. The cross-domain feature achieves the best VAD (SER 14.53%) and OSD (F1-score 69.86%) results. This illustrates that both the time domain and time-frequency domain make a unique contribution to VAD and OSD tasks.

TABLE II

VAD AND OSD RESULTS USING FEATURES FROM DIFFERENT DOMAINS.

| Domains | VAD | | | OSD | | |
|---|---|---|---|---|---|---|
| | FA↓ | Miss↓ | SER↓ | Pr↑ | Re↑ | F1↑ |
| Time | 14.95 | 2.85 | 17.80$_{+0.00\%}$ | 57.51 | 66.26 | 61.58$_{+0.00\%}$ |
| Time-frequency | 11.89 | 2.70 | 14.59$_{-3.21\%}$ | 63.58 | 75.62 | 69.08$_{+7.50\%}$ |
| Cross | 12.08 | 2.45 | **14.53**$_{-3.27\%}$ | 66.65 | 73.39 | **69.86**$_{+8.28\%}$ |

### B. Impact of Encoder Depths

The encoder used in the TD feature extractor consists of stacked basic layers, each comprising a Conv1D layer, a PReLU activation, and a LayerNorm operation. To investigate the impact of the number of encoder depths on VAD and OSD performance, we tested three different configurations: $K$=0 layers, $K$=3 layers, and $K$=6 layers. As shown in Table III, removing all the basic layers (i.e., $K$=0 layers) or using too many encoder layers (i.e., $K$=6 layers) leads to reduced VAD and OSD performance compared to the encoder with 3 layers. This is probably because the 0-layer encoder does not extract deep TD features, while the encoder with deeper layers may lead to overfitting.

### C. Different Domain Fusion Strategies

To verify the effectiveness of the proposed bidirectional cross-domain fusion method, we compared it with two classic fusion strategies: concatenation-based fusion in Fig. 3(a) and

TABLE III
VAD AND OSD RESULTS AT DIFFERENT ENCODER DEPTHS.

| Encoder depth | VAD | | | OSD | | |
|---|---|---|---|---|---|---|
| | FA↓ | Miss↓ | SER↓ | Pr↑ | Re↑ | F1↑ |
| $K = 0$ | 12.24 | 2.60 | 14.84$_{+0.00\%}$ | 63.50 | 75.97 | 69.18$_{+0.00\%}$ |
| $K = 3$ | 12.08 | 2.45 | **14.53**$_{-0.31\%}$ | 66.65 | 73.39 | **69.86**$_{+0.68\%}$ |
| $K = 6$ | 11.85 | 2.94 | 14.79$_{-0.05\%}$ | 63.34 | 74.77 | 68.58$_{-0.60\%}$ |

TABLE IV
VAD AND OSD RESULTS BASED ON DIFFERENT DOMAIN FUSION METHODS.

| Fusion methods | VAD | | | OSD | | |
|---|---|---|---|---|---|---|
| | FA↓ | Miss↓ | SER↓ | Pr↑ | Re↑ | F1↑ |
| Concatenation | 12.34 | 2.34 | 14.68$_{+0.00\%}$ | 64.31 | 74.98 | 69.24$_{+0.00\%}$ |
| Self-Attention | 12.15 | 2.63 | 14.78$_{+0.10\%}$ | 66.75 | 72.84 | 69.66$_{+0.42\%}$ |
| Bidirectional Cross | 12.08 | 2.45 | **14.53**$_{-0.15\%}$ | 66.65 | 73.39 | **69.86**$_{+0.62\%}$ |

self-attention-based fusion in Fig. 3(b). From Table IV, we can see that concatenation-based fusion exhibits the worst VAD SER and OSD F1 performance. By comparison, self-attention-based fusion improves the F1 score by 0.4% in OSD tasks, but results in a 0.1% increase in VAD SER. Compared to these strategies, the bidirectional cross-domain fusion strategy improves both VAD and OSD performance, achieving the best VAD SER of 14.53% and OSD F1 score of 69.86%. These results demonstrate that bidirectional cross-domain fusion can combine information from different domains more effectively.

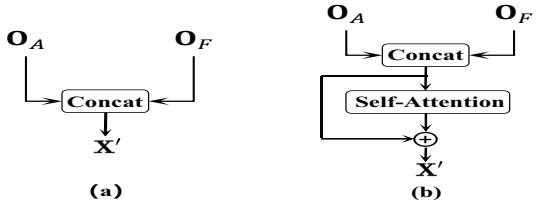


Fig. 3. Comparison domain-fusion strategies: (a) concatenation-based fusion, (b) self-attention-based fusion.

## VII. CONCLUSION

In this work, we designed a lightweight cross-domain feature extractor (L-CD-FE) for multichannel VAD and OSD, which extracts deep features from both the time domain and time-frequency domain. The L-CD-FE was validated on two well-known sequence modeling networks (i.e., BLSTM and TCN). The results showed that our method achieves competitive VAD and OSD performance with significantly reduced parameter count and FLOPs compared to the state-of-the-art front-end feature extractors. In future work, we will study lightweight sequence modeling networks and combine them with the L-CD-FE, developing more lightweight multichannel VAD and OSD.

## REFERENCES


[1] S. Wang, Z. Chen, K. A. Lee, Y. Qian, and H. Li, "Overview of speaker modeling and its applications: From the lens of deep speaker representation learning," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 2024.

[2] J. Han, F. Landini, J. Rohdin, M. Diez, L. Burget, Y. Cao, H. Lu, and J. Černockỳ, "Diacorrect: Error correction back-end for speaker diarization," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2024, pp. 11 181–11 185.

[3] K. Radha, M. Bansal, and R. B. Pachori, "Speech and speaker recognition using raw waveform modeling for adult and children's speech: A comprehensive review," *Eng. Appl. Artif*, vol. 131, p. 107661, 2024.

[4] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 6, no. 1, pp. 1–3, 1999.

[5] T. Ng, B. Zhang, L. Nguyen, S. Matsoukas, X. Zhou, N. Mesgarani, K. Veselỳ, and P. Matejka, "Developing a speech activity detection system for the DARPA RATS program." in *Proc. Interspeech*, vol. 9, 2012, pp. 1–4.

[6] R. Sarikaya and J. H. Hansen, "Robust detection of speech activity in the presence of noise," in *Proc. ICSLP*, 1998, pp. 1455–8.

[7] M. Lavechin, M.-P. Gill, R. Bousbib, H. Bredin, and L. P. Garcia-Perera, "End-to-end domain-adversarial voice activity detection," in *Proc. Interspeech*, 2020, pp. 3685–3689.

[8] S. Thomas, S. Ganapathy, G. Saon, and H. Soltau, "Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2014, pp. 2519–2523.

[9] J. T. Geiger, F. Eyben, B. Schuller, and G. Rigoll, "Detecting overlapping speech with long short-term memory recurrent neural networks," in *Proc. Interspeech*, 2013, pp. 1668–1672.

[10] J.-w. Jung, H.-S. Heo, Y. Kwon, J. S. Chung, and B.-J. Lee, "Three-class overlapped speech detection using a convolutional recurrent neural network," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 3086–3090.

[11] M. Kunešová, M. Hrúz, Z. Zajíc, and V. Radová, "Detection of overlapping speech for the purposes of speaker diarization," in *Proc. Int. Conf. Speech Comput.*, 2019, pp. 247–257.

[12] T. Mariotte, A. Larcher, and S. Montrésor, "Channel-combination algorithms for robust distant voice activity and overlapped speech detection," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 32, pp. 1859–1872, 2024.

[13] S. Cornell, M. Omologo, S. Squartini, and Vincent, "Detecting and counting overlapping speakers in distant speech scenarios," in *Proc. Interspeech*, 2020, pp. 3107–3111.

[14] S. Cornell, M. Omologo, S. Squartini, and E. Vincent, "Overlapped speech detection and speaker counting using distant microphone arrays," *Comput. Speech Lang.*, vol. 72, p. 101306, 2022.

[15] T. Mariotte, A. Larcher, S. Montrésor, and J.-H. Thomas, "Multi-microphone automatic speech segmentation in meetings based on circular harmonics features," in *Proc. Interspeech*, 2023, pp. 2783–2787.

[16] R. Gong, C. Quillen, D. Sharma, A. Goderre, J. Laınez, and L. Milanovic, "Self-attention channel combinator frontend for end-to-end multichannel far-field speech recognition," in *Proc. Interspeech*, 2021, pp. 3840–3844.

[17] A. Vaswani, "Attention is all you need," in *Proc. Int. Conf. Neural Inf. process. Syst.*, 2017, pp. 6000–6010.

[18] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, and M. Kronenthal, "The AMI meeting corpus: A pre-announcement," in *Proc. Int. Workshop Mach. Learn. Multimodal Interact.*, 2005.

[19] L. Wan, H. Liu, L. Shi, Y. Zhou, and L. Gan, "Cross domain optimization for speech enhancement: Parallel or cascade?" *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 2024.

[20] Z. Yang, L. Zhu, Y. Wu, and Y. Yang, "Gated channel transformation for visual recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11 794–11 803.

[21] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4489–4497.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Adv. Neural Inf. Process. Syst.*, vol. 25, 2012.

[23] F. Landini, J. Profant, M. Diez, and L. Burget, "Bayesian HMM clustering of x-vector sequences (VBX) in speaker diarization: Theory, implementation and analysis on standard tasks," *Comput. Speech Lang.*, vol. 71, p. 101254, 2022.

[24] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015.