INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

# Multi-channel multi-speaker transformer for speech recognition

*Guo Yifan*[1], *Tian Yao*[1], *Suo Hongbin*[1], *Wan Yulong*[1]

[1]Data & AI Engineering System, OPPO, Beijing, China

{guoyifan, aaron1, suohongbin, wanyulong}@oppo.com

## Abstract

With the development of teleconferencing and in-vehicle voice assistants, far-field multi-speaker speech recognition has become a hot research topic. Recently, a multi-channel transformer (MCT) has been proposed, which demonstrates the ability of the transformer to model far-field acoustic environments. However, MCT cannot encode high-dimensional acoustic features for each speaker from mixed input audio because of the interference between speakers. Based on these, we propose the multi-channel multi-speaker transformer (M2Former) for far-field multi-speaker ASR in this paper. Experiments on the SMS-WSJ benchmark show that the M2Former outperforms the neural beamformer, MCT, dual-path RNN with transform-average-concatenate and multi-channel deep clustering based end-to-end systems by 9.2%, 14.3%, 24.9%, and 52.2% respectively, in terms of relative word error rate reduction.

**Index Terms**: multi-channel ASR, multi-speaker ASR, transformer

## 1. Introduction

As teleconferencing and in-vehicle voice assistants become increasingly popular, far-field multi-speaker speech recognition has become a hot research topic. Current mainstream methods are based on the permutation invariant training (PIT) [1]. Specifically, they first use far-field speech separation frontends [2–8] to output single-channel speech feature for each speaker, and then decode these features with single-channel automatic speech recognition (ASR) backend with PIT. Compared to the serialized output training [9] based methods, PIT based methods could decode faster and are closer to practical applications.

Recently, researchers [10, 11] proposed a multi-channel transformer (MCT) for far-field speech recognition. Experiments show that MCT outperforms the commonly used systems [12–14] which use enhancement models as frontends and single-channel ASR models as backends. This is mainly because the inconsistency in functional design between frontends and backends brings performance limitations to the latter systems. Inspired by this, we try to bypass the paradigm of separation frontends recognition backends (Sep-ASR) [2–8] and propose a multi-channel multi-speaker transformer (M2Former), whose encoder can encode high dimensional acoustic embeddings for each speaker from the mixture input audio directly.

However, there are certain problems in using MCT for multi-speaker scenario. The encoder of MCT encodes the contextual relationship under far-field environments by utilizing the intra-channel continuity information and cross-channel spatial information. When utilizing the cross-channel information, MCT combines channels using learnable weights which are expected to learn to model the spatial relationships implicitly. But in the multiple speakers case, the interference between speakers and noise makes it difficult for the encoder to generate high quality acoustic embeddings for each speaker.

To alleviate such interference, two dimensional convolutional neural networks (2D-CNN) and an improved multi-channel attention mechanism called multi-channel multi-speaker attention (M2A) are used. 2D-CNN are widely used in image tasks and speech enhancement tasks [15–19] as feature decoupling modules. This is due to their ability to learn diverse facets with different filter channels [15, 17, 20, 21]. For example, researchers [15] find that different filters of the CNN focus on signals coming from different directions. Based on this, we use a 2D-CNN to decouple the multi-channel inputs first. As there are always physical or content differences between signal sources, each output channel of the CNN is expected to have a high probability of containing information corresponding to only one speaker (or noise). Then we propose the M2A module to encode contextual relationship. Instead of using all channels like MCT, the M2A only utilize the cross-channel information between channels with high similarity. Therefore, signals corresponding to different sources are encoded separately, and the interference mentioned above is avoided to a certain extent.

Furthermore, we employ a spectral clustering [22] based approach when separating speaker-specific features from the input mixtures. Compared to the projection based method [4–6, 8] which use linear projection layers to predict masks for speakers, there are two advantages. 1) After clustering, channels corresponding to different sound sources are assigned to different clusters. Using M2A within each cluster can better avoid the interference between sources. 2) There is no need to train additional linear layers, so it is easier to be applied to the situation of more speakers and unfixed number of speakers. Moreover, the similarity matrix required for clustering can be easily obtained from the M2A. Besides, we utilize a method [23] to distinguish the noise and speech. By discarding noise cluster, the background noise is filtered out.

Based on the above approaches, we propose the M2Former for far-field multi-speaker ASR. Experiments on the SMS-WSJ [24] show that it outperforms the neural beamformer [4], MCT, dual-path RNN with transform-average-concatenate [6] and multi-channel deep clustering [3] based systems by 9.2%, 14.3%, 24.9%, and 52.2% respectively, in terms of relative WER reduction.

## 2. Proposed Method

This section describes the proposed M2Former framework, which is shown in Figure 1.

The structure and function of the M2Former's decoder are consistent with the decoder in the original speech transformer [25, 26]. It decodes each speaker's single-channel acoustic em-



10.21437/Interspeech.2023-257

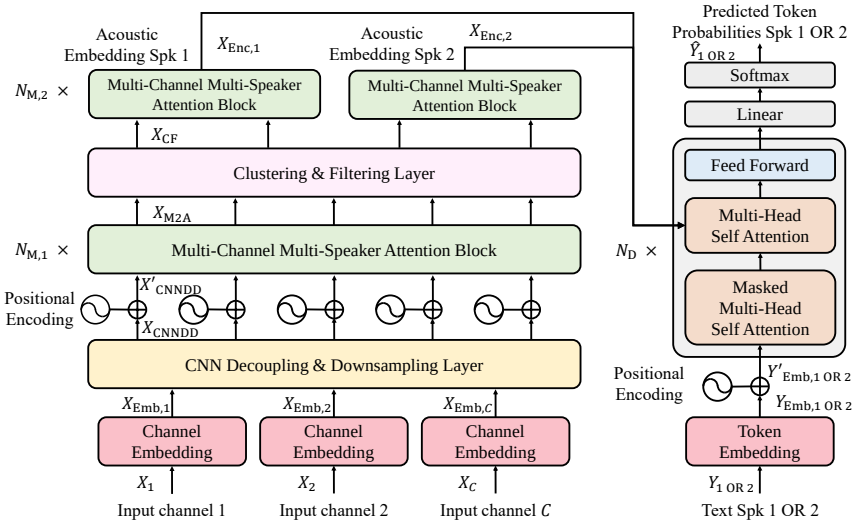


Figure 1: *Multi-channel multi-speaker transformer. $C$ is the number of input channels. $N_M$ and $N_D$ represent the numbers of M2A blocks and decoder blocks respectively. Assume that there are two speakers in the input audio.*

beddings into text separately. Thus, we will focus on each component of the encoder, which is the core of the M2Former.

### 2.1. Channel Embedding

In order to obtain suitable input features for network learning, we add a channel embedding layer before the encoder following [10]. Specifically, we use the magnitude $X^{\text{mag}}$ and phase features $X^{\text{pha}}$ of STFT as the input features $X \in \mathbb{R}^{C\times T\times 3F}$. Then we use linear projection layer to obtain the embeddings $X_{\text{Emb}} \in \mathbb{R}^{C\times T\times D}$ (for simplicity, we ignore the bias):

$$X_{\text{Emb},i} = [X_i^{\text{mag}} W^{\text{mag}}, X_i^{\text{pha}} W^{\text{pha}}] \cdot W^{\text{Emb}} \tag{1}$$

Here $X_i$ denotes the $i$th channel of X, $X_{\text{Emb},i}$ denotes the $i$th channel of $X_{\text{Emb}}$, $W$ denotes linear projection layer, and $[\ldots,\ldots]$ denotes feature concatenation.

### 2.2. CNN Decoupling and Downsampling Layer

Inspired by [15–19], a 2D CNN module is utilized to decouple the input multi-channel features $X_{\text{Emb}}$ into $X_{\text{CNNDD}}$ with more channels for more discriminative representations. Besides, motivated by the sparse distribution of the spectrum, we use the CNNs to downsample the input features on both frequency and time dimensions for computation efficiency. Thus we obtain $X_{\text{CNNDD}} \in \mathbb{R}^{C'\times T'\times D'}$ where $C' > C$, $T' < T$, and $D' < D$.

### 2.3. Multi-Channel Multi-Speaker Attention Block

In order to directly encode the contextual relationship for each speaker by utilizing the intra-channel and cross-channel information, we proposed the M2A. Following [10, 11], the M2A consists of an intra-channel attention layer (Figure 2a) and a cross-channel attention layer (Figure 2c).

As described in the introduction, each output channel of the CNN is expected to have a high probability of containing information corresponding to only one speaker (or noise). So we can first utilize the self-attention in each channel in the same way as the commonly used transformer based single-channel ASR encoder does [10, 11, 26] (Figure 2a). It can learn the contextual continuity information within a channel.

Then we propose a cross-channel attention layer to utilize both the continuity information between frames and the spatial information between channels. As shown in Figure 2b, the MCT [10,11] combines channels using learnable weights $P$ (for simplicity, we don't introduce the specific design of the $P$). In such ways, channels belonging to different speakers would interfere with each other. This is because there are many permutations of the correspondence between channels and sound sources, which is difficult to be modeled with fixed parameters. To avoid such problem, we propose the M2A cross-channel attention (Figure 2c). Let $X_c, X'_c \in \mathbb{R}^{T'\times D'}$ denote the $c$-th channel's input and output features of the cross-channel attention layers respectively. Then $X'$ are computed as:

$$\begin{aligned}
\hat{X}_c &= \sum_i z_{ci} \cdot X_i \\
Q_c &= X_c W^q + \mathbf{1} \cdot (\mathbf{b}^q)^{\mathrm{T}} \\
K_c &= \hat{X}_c W^k + \mathbf{1} \cdot (\mathbf{b}^k)^{\mathrm{T}} \\
V_c &= \hat{X}_c W^v + \mathbf{1} \cdot (\mathbf{b}^v)^{\mathrm{T}} \\
X'_c &= \text{softmax}\left(Q_c K_c^{\mathrm{T}} / \sqrt{d_k}\right) V_c
\end{aligned} \tag{2}$$

Where $W^*, \mathbf{b}^*$ are learnable weights and bias, $d_k$ is the scaling factor. We omit the multihead mechanism here for simplicity. The $z_{ci}$ is the $c$-th row $i$-th column element of the inter-channel similarity matrix $Z \in \mathbb{R}^{C'\times C'}$, which reflects the similarity between the channel $c$ and $i$:

$$Z = \text{softmax}\left(1/T \textstyle\sum_t X_t X_t^{\mathrm{T}} / \sqrt{d_k}\right) \tag{3}$$

Since the signals of different sound sources are usually uncorrelated, combining based on similarity can alleviate the interference mentioned above. After the last M2A block in the encoder (Figure 1), we average the channels corresponding to the same speaker to obtain a single-channel output for each speaker.

### 2.4. Clustering and Filtering Layer

The clustering and filtering (CF) layer is shown in the figure 3. First, we conduct spectral clustering [22] with the inter-channel similarity matrix $Z$ generated by the nearest M2A block to assign a label to each channel. After that, we calculate the inter-frame similarity difference (IFSD) values [23] for each output

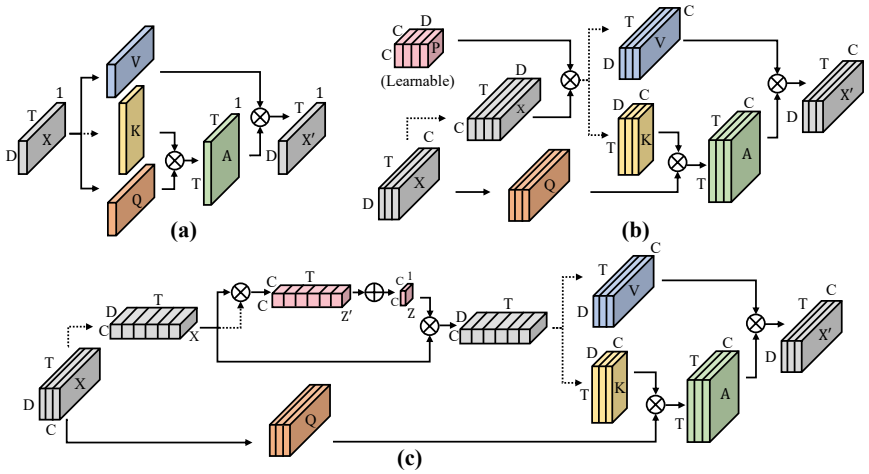


Figure 2: *Multi-channel attention blocks:* ***(a)****Intra-channel attention.* ***(b)*** *MCT's cross-channel attention.* ***(c)*** *M2A's cross-channel attention.* $\bigotimes$, $\bigoplus$ *and dashed lines denote matrix multiplication, averaging on time dimension and matrix transpose respectively.*

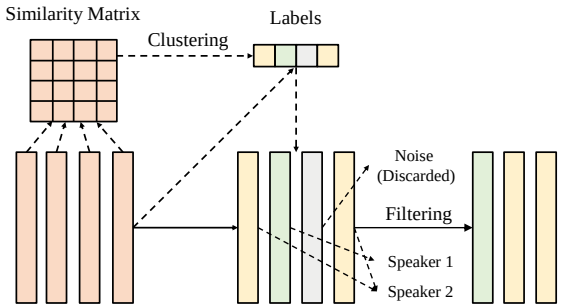


Figure 3: *Clustering and filtering layer. Gradients are propagated only in the solid line. Assume that there are 4 channels and 2 speakers, and the gray label get the lowest IFSD value.*

channel, and average the values belonging to the same label:

$$\mathrm{IFSD}\,(X) = \frac{1}{T}\sum_{t}\left[\hat{\mathbf{x}}_t^{\mathrm{T}}\cdot\hat{\mathbf{x}}_{t+1} - \alpha\cdot\hat{\mathbf{x}}_t^{\mathrm{T}}\cdot\hat{\mathbf{x}}_{t+\tau}\right] \qquad (4)$$

Where $X \in \mathbb{R}^{D\times T}$ is the input features, $\hat{\mathbf{x}}_t \in \mathbb{R}^{D}$ is the normalized $t$-th frame of $X$, and $\alpha$ and $\tau$ are hyperparameters. Finally we keep $n$ labels with highest IFSD values, and discard the others ($n$ is the number of the speakers). The rationale is that IFSD values represent the probabilities that channels are dominated by speech rather than noise. Thus the preserved labels correspond to the speech sources, while the others correspond to the noise.

As shown in Figure 1, we also try to use M2A between channels belonging to the same cluster, which get better results.

# 3. Experiments

## 3.1. Datasets

To evaluate our proposed model, we conduct several experiments on the SMS-WSJ [24] Dataset. SMS-WSJ is a six-channel multi-speaker far-field dataset. The SMS-WSJ has three versions of 2, 3 and 4 simultaneous speakers, and we denote them as SMS-2, SMS-3 and SMS-4. In this paper, we conduct experiments mainly on the SMS-2, while the other two datasets are only used for validation. The train set, valid set, and test set of SMS-2 consist of 33561, 982, and 1332 utterances respectively, for a total of 93.3 hours. SMS-3 and SMS-4 share the same original data division configuration with SMS-2.

## 3.2. Configurations of Baselines and the Proposed Model

### *3.2.1. Separation Frontends of Baselines*

To demonstrate the performance of our method, three representative multi-channel separation systems are used as the frontends in our baselines. We use STFT features with 25 ms frame length and 10 ms frame shift as the raw features. The first input channel is used as the reference channel.

**Multi-Channel Deep Clustering (MC-DPCL) [3]** : the MC-DPCL uses a 3 layer bLSTM with 300 hidden units as the embedding extractor. It takes the magnitude and phase features generated from STFT features as inputs. To make it so it can be trained end-to-end, we utilize soft k-means method as in [27].

**Dual-Path RNN with Transform-Average-Concatenate (DPRNN-TAC) [6]**: 6 DPRNN blocks are used in the system with 128 hidden units. The feature dimension and segment size of the DPRNN-TAC are 64 and 24. It takes the time domain signals as input directly.

**Neural Beamformers (NB) [4]**: we use a 3 layer bLSTM with 300 hidden units with two linear projection layers as the mask estimators for two speakers respectively.

### *3.2.2. ASR Backend of Baselines*

The ASR backend of the baselines is a unified transformer-based single-channel AED model [26] which is trained end-to-end with the frontend systems using PIT. There are 12 blocks in the encoder, 6 blocks in the decoder, 4 heads for multi-head attention whose attention dimension is 256, and 1024 units for the position-wise feed forward.

### *3.2.3. The Proposed Model*

The 2D CNN module in the CNNDD has eight CNN layers with 6, 6, 10, 10, 20, 20, 40, 40 output channels. The kernel sizes of the CNNs are all $3 \times 3$ with zero paddings on both dimensions. The strides of the first layer and second layer are [2,2] and [2,1], while the other layers' are all 1. $\alpha$ of IFSD is set to 5.3 according to the results on the valid set. To make the M2Former have the comparable size, we set the sum of $N_{\mathrm{M},1}$ and $N_{\mathrm{M},2}$ to 6, and the $N_{\mathrm{D}}$ to 6, the attention dimension to 256, the heads of the multi-head attention to 4, and the feed forward units to 1024.

Table 1: *Word error rate (WER%) of experiments on proposed model and baseline models.*

| SMS-2 | cv_dev93 | test_eval92 | MEAN |
|---|---|---|---|
| MC-DPCL [3] | 34.7 | 39.6 | 37.2 |
| DPRNN-TAC [6] | 22.3 | 25.1 | 23.7 |
| NB [4] | 18.4 | 20.7 | 19.6 |
| M2Former | **16.6** | **18.9** | **17.8** |

Table 2: *Ablation Experiments on the SMS-2 (WER%).[-] means the module is removed during experiment).*

| SMS-2 | cv_dev93 | test_eval92 | MEAN |
|---|---|---|---|
| [-] CNNDD | 35.5 | 40.1 | 37.8 |
| [-] $M2A_1$ | 24.2 | 28.7 | 26.5 |
| [-] $M2A_2$ | 18.3 | 21.0 | 19.7 |
| [-] IFSD | 17.9 | 20.8 | 19.4 |
| Complete Model | **16.6** | **18.9** | **17.8** |

#### *3.2.4. Loss Function of M2Former and the Baselines*

Following [28], the loss function $\mathcal{L}$ consists of the CTC loss [29] of the encoder and the attention loss of the decoder. The CTC loss is used to determine the best permutation with PIT.

$$\begin{aligned}\mathcal{L} &= \sum\nolimits_i [\lambda \mathcal{L}_{\text{ctc}}(X_{\text{Enc},i}, Y_{\hat{\pi}(i)}) + (1-\lambda)\mathcal{L}_{\text{att}}(\hat{Y}_i, Y_{\hat{\pi}(i)})] \\ \hat{\pi} &= \text{argmin}_{\pi\in\mathcal{P}}(\sum\nolimits_i \mathcal{L}_{\text{ctc}}(X_{\text{Enc},i}, Y_{\pi(i)}))\end{aligned} \tag{5}$$

Where $i = 1, 2, ..., N$ represents the speaker id, and $\mathcal{P}, Y, \hat{Y}$, $X_{\text{Enc}}$ represents all possible permutations, the ground truth tokens, predicted tokens and outputs of encoder respectively.

### 3.3. Experimental Results

#### *3.3.1. Comparison between M2Former and the Baselines*

We conduct experiments on the SMS-2 dataset to compare the performance between the M2Former and the baselines (Table 1). Our proposed model achieves the best results on the SMS-2. Specifically, it outperforms the NB, DPRNN-TAC and MC-DPCL based end-to-end systems by 9.2%, 24.9%, and 52.2% respectively, in terms of relative word error rate reduction.

#### *3.3.2. Ablation Experiments of the Modules of the Encoder*

We conduct several ablation experiments to verify the effectiveness of each module in the encoder (Table 2). For the experiments of the CNNDD, we replace the original CNNs with two $3 \times 3$ 2D CNNs whose channel dimensions are 40 and strides are [2,2] and [2,1]. For the experiments of $M2A_1$, we just set the $N_{\text{M},2}$ to 6 and $N_{\text{M},1}$ to 0. The setup method of $M2A_2$ experiment is similar to that of the $M2A_1$, except we add a linear layer after the CF layer to smooth out the distortion introduced by the CF layer. For the IFSD experiments, we set the number of clusters to 2 and discard no channels during the CF layer.

We can find that the CNNDD contributes the most to the overall performance as it can decouple the inputs into outputs corresponding to different facets. The $M2A_1$ and the $M2A_2$ are useful. The $M2A_1$ can encode acoustic embeddings without being disturbed by other speakers. And the $M2A_2$ could eliminate the influence of the distortion caused by CF and the possible residual noise while encoding. The IFSD method also contributes to the noise cancellation.

Table 3: *WER% of experiments on the M2A and the MCT.*

| SMS-2 | cv_dev93 | test_eval92 | MEAN |
|---|---|---|---|
| s-CNN + MCT | 34.8 | 38.1 | 36.5 |
| s-CNN + STA [30, 31] | 31.4 | 35.6 | 33.6 |
| s-CNN + $M2A_1$ | 40.2 | 47.7 | 44.0 |
| CNNDD + MCT | 21.2 | 24.7 | 23.0 |
| CNNDD + STA | 19.7 | 22.8 | 21.3 |
| CNNDD + $M2A_1$ | 18.3 | 21.0 | 19.7 |

Table 4: *Performance of M2Former with different numbers of speakers on the SMS-WSJ datasets (WER%). The second column denotes whether the number of speakers is known in advance during inference.*

| | Known | cv_dev93 | test_eval92 | MEAN |
|---|---|---|---|---|
| SMS-2 | ✓ | 16.6 | 18.9 | 17.8 |
| | ✗ | 16.9 | 19.3 | 18.1 |
| SMS-3 | ✓ | 28.5 | 32.7 | 30.6 |
| | ✗ | 30.8 | 34.5 | 32.7 |
| SMS-4 | ✓ | 33.4 | 37.1 | 35.3 |
| | ✗ | 37.9 | 41.8 | 39.9 |

#### *3.3.3. Experimental comparison between the M2A and MCT*

To further explore the performance of M2A, we conduct experiments to compare the M2A with MCT and spatial-temporal attention (STA) [30, 31] (Table 3). All 3 models use 6 attention blocks in the encoder. As the MCT can only be used with fixed number of input channels, we don't utilize attention blocks after the CF layer. We also consider the cases when replacing the CNNDD with the single CNN (s-CNN) mentioned in 3.3.2.

Comparing line 1 with 4 (or 2 with 5 and 3 with 6) in table 3, the importance of the CNNDD is demonstrated again. Comparing line 1, 2 and 3, we can find that without the decoupling of the CNNDD, the M2A can be more susceptible to the interference between sources within each channel, while the MCT and STA can learn some speaker-related information implicitly. Comparing line 4, 5 and 6, it can be found that the CNNDD-M2A can better avoid interference between channels corresponding to different sound sources, and achieves a 14.3% WER reduction relative to the CNNDD-MCT.

#### *3.3.4. Performance with Different Numbers of Speakers*

We conduct a series of experiments to explore the performance of M2Former with different numbers of speakers (Table 4). The eigengap heuristic method [32] are used to determine the number of speakers for the case where the number is unknown.

By comparing each two rows belonging to the same dataset, we can find that the number of speakers could be relatively accurately estimated from the decoupled channels. This shows that the model has learned to decouple features to different channels according to sound sources. Besides, although the model has not been trained with more than two speakers, it works okay on the SMS-3 and SMS-4. This may show that the model could be easily applied to the situation of more speakers.

## 4. Conclusion

We proposed an end-to-end multi-channel multi-speaker transformer for speech recognition. By using CNN decoupling and downsampling layer, multi-channel multi-speaker attention block, and clustering and filtering layer, the encoder can encode speaker-wise acoustic features directly from the mixture input. The experiments shows that the proposed model outperforms the separation-recognition form baselines and the MCT.

# 5. References


[1] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 241–245.

[2] L. Drude and R. Haeb-Umbach, "Tight integration of spatial and spectral features for bss with deep clustering embeddings." in *Interspeech*, 2017, pp. 2650–2654.

[3] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5.

[4] X. Chang, W. Zhang, Y. Qian, J. Le Roux, and S. Watanabe, "Mimo-speech: End-to-end multi-channel multi-speaker speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 237–244.

[5] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[6] Y. Luo, Z. Chen, N. Mesgarani, and T. Yoshioka, "End-to-end microphone permutation and number invariant multi-channel speech separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6394–6398.

[7] R. Gu, S.-X. Zhang, L. Chen, Y. Xu, M. Yu, D. Su, Y. Zou, and D. Yu, "Enhancing end-to-end multi-channel speech separation via spatial feature learning," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7319–7323.

[8] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 21–25.

[9] N. Kanda, Y. Gaur, X. Wang, Z. Meng, and T. Yoshioka, "Serialized output training for end-to-end overlapped speech recognition," *arXiv preprint arXiv:2003.12687*, 2020.

[10] F.-J. Chang, M. Radfar, A. Mouchtaris, B. King, and S. Kunzmann, "End-to-end multi-channel transformer for speech recognition," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5884–5888.

[11] F.-J. Chang, M. Radfar, A. Mouchtaris, and M. Omologo, "Multi-channel transformer transducer for speech recognition," *arXiv preprint arXiv:2108.12953*, 2021.

[12] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 196–200.

[13] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel end-to-end speech recognition," in *International conference on machine learning*. PMLR, 2017, pp. 2632–2641.

[14] K. Kumatani, W. Minhua, S. Sundaram, N. Ström, and B. Hoffmeister, "Multi-geometry spatial acoustic modeling for distant speech recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6635–6639.

[15] T. N. Sainath, R. J. Weiss, K. W. Wilson, A. Narayanan, M. Bacchiani, and A. Senior, "Speaker location and microphone spacing invariant acoustic modeling from raw multichannel waveforms," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 30–36.

[16] Y. Qian, M. Bi, T. Tan, and K. Yu, "Very deep convolutional neural networks for noise robust speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2263–2276, 2016.

[17] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1570–1584, 2018.

[18] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, "Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement," *arXiv preprint arXiv:2008.00264*, 2020.

[19] T. Park, K. Kumatani, M. Wu, and S. Sundaram, "Robust multi-channel speech recognition using frequency aligned network," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6859–6863.

[20] D. Wei, B. Zhou, A. Torrabla, and W. Freeman, "Understanding intra-class knowledge inside cnn," *arXiv preprint arXiv:1507.02379*, 2015.

[21] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5188–5196.

[22] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 14, 2001.

[23] Y. Guo, Y. Chen, G. Cheng, P. Zhang, and Y. Yan, "Far-field speech recognition based on complex-valued neural networks and inter-frame similarity difference method," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021, pp. 1003–1010.

[24] L. Drude, J. Heitkaemper, C. Boeddeker, and R. Haeb-Umbach, "Sms-wsj: Database, performance measures, and baseline recipe for multi-channel source separation and recognition," *arXiv preprint arXiv:1910.13934*, 2019.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[26] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5884–5888.

[27] Y.-J. Lu, X. Chang, C. Li, W. Zhang, S. Cornell, Z. Ni, Y. Masuyama, B. Yan, R. Scheibler, Z.-Q. Wang *et al.*, "Espnet-se++: Speech enhancement for robust speech recognition, translation, and understanding," *arXiv preprint arXiv:2207.09514*, 2022.

[28] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[29] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[30] D. Wang, T. Yoshioka, Z. Chen, X. Wang, T. Zhou, and Z. Meng, "Continuous speech separation with ad hoc microphone arrays," in *2021 29th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 1100–1104.

[31] S. Horiguchi, Y. Takashima, P. Garcia, S. Watanabe, and Y. Kawaguchi, "Multi-channel end-to-end neural diarization with distributed microphones," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7332–7336.

[32] M. Bolla, *Relations between spectral and classification properties of multigraphs*. DIMACS, Center for Discrete Mathematics and Theoretical Computer Science, 1991.