



# End-to-end Multichannel Speaker-Attributed ASR: Speaker Guided Decoder and Input Feature Analysis

Can Cui, Imran Ahamad Sheikh, Mostafa Sadeghi, Emmanuel Vincent

## ► To cite this version:

Can Cui, Imran Ahamad Sheikh, Mostafa Sadeghi, Emmanuel Vincent. End-to-end Multichannel Speaker-Attributed ASR: Speaker Guided Decoder and Input Feature Analysis. 2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU 2023), Dec 2023, Taipei, Taiwan. 10.1109/ASRU57964.2023.10389729 . hal-04235774

**HAL Id: hal-04235774**

**<https://inria.hal.science/hal-04235774>**

Submitted on 11 Oct 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# END-TO-END MULTICHANNEL SPEAKER-ATTRIBUTED ASR: SPEAKER GUIDED DECODER AND INPUT FEATURE ANALYSIS

Can Cui<sup>1,2</sup>, Imran Sheikh<sup>2</sup>, Mostafa Sadeghi<sup>1</sup>, Emmanuel Vincent<sup>1</sup>

<sup>1</sup>Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

<sup>2</sup>Vivoka, Metz, France

## ABSTRACT

We present an end-to-end multichannel speaker-attributed automatic speech recognition (MC-SA-ASR) system that combines a Conformer-based encoder with multi-frame cross-channel attention and a speaker-attributed Transformer-based decoder. To the best of our knowledge, this is the first model that efficiently integrates ASR and speaker identification modules in a multichannel setting. On simulated mixtures of LibriSpeech data, our system reduces the word error rate (WER) by up to 12% and 16% relative compared to previously proposed single-channel and multichannel approaches, respectively. Furthermore, we investigate the impact of different input features, including multichannel magnitude and phase information, on the ASR performance. Finally, our experiments on the AMI corpus confirm the effectiveness of our system for real-world multichannel meeting transcription.

**Index Terms**— Speaker-attributed ASR, cross-channel attention, multichannel ASR, phase features, AMI

## 1. INTRODUCTION

Automatic processing of multi-party (a.k.a. multi-speaker) speech recordings, such as meetings, requires multi-speaker automatic speech recognition (ASR) and diarization systems. The main challenges faced in these scenarios include overlapping speech, reverberation caused by distant microphones, and background noise. End-to-end multi-speaker ASR and diarization systems for single-channel [1–4] and multichannel recordings [5–8] have recently emerged, demonstrating promising results on meeting transcription tasks.

The earlier approaches for end-to-end multi-speaker ASR and diarization for single-channel recordings lacked information sharing between the ASR and the speaker verification modules [2] and/or required a varying number of speaker encoder (or attention) modules based on the number of speakers [1, 3]. The authors of [4] proposed an end-to-end single-channel speaker-number invariant Transformer-based speaker-attributed ASR (SA-ASR) system based on serialized output training (SOT). This system shares both speech and speaker representations across the multi-speaker ASR and speaker diarization tasks. Similar to their single-channel counterparts, the ASR and speaker identification

modules in end-to-end multichannel ASR and diarization approaches [5–7] do not share speech and speaker representations. Interestingly, the approach of multichannel word-level diarization with SOT (MC-WD-SOT) in [8] performs a fusion of ASR and speaker information. It uses multi-frame cross-channel attention (MFCCA) [9] in the ASR encoder to integrate information from different channels and hidden-layer embeddings from the ASR decoder to assist speaker identification. However, the ASR and the diarization modules in this approach use different types of attention for multichannel fusion, leading to an increase in the number of model parameters. Moreover, the communication of ASR and speaker information is not bidirectional, since the speaker information is not reciprocally exploited by ASR.

Multichannel SA-ASR (MC-SA-ASR) can exploit spatial information which is generally advantageous for localizing different speakers. However, the predominant approaches in end-to-end multichannel ASR use Mel filterbank features as input and discard phase information [9, 10]. More recently, the “all-in-one” model in [11] uses interchannel phase difference (IPD) [12–14] and spatial features, wherein the later require video information to localize the speakers. In [15], magnitude and phase information from each channel were separately passed through linear layers and then concatenated to form the input for multichannel ASR. To our knowledge, there is currently no research comparing and discussing the impact of different input features on MC-SA-ASR.

In this paper, we target end-to-end MC-SA-ASR, and investigate the benefit of i) leveraging speaker information for ASR and ii) exploiting phase information. To do so, we propose an MC-SA-ASR system that tightly couples a Conformer-based encoder with MFCCA, a speaker encoder and a speaker-attributed Transformer-based decoder. We explore the use of phase information as input. We conduct extensive experiments on simulated mixtures of LibriSpeech data as well as on the AMI corpus.

The rest of the paper is organized as follows. Section 2 presents the related works. In Section 3, we introduce our proposed MC-SA-ASR system and the different features and feature encoding methods. Section 4 presents our experimental setup and results on simulated data, as well as on real meeting data. Finally, Section 5 provides a conclusion.

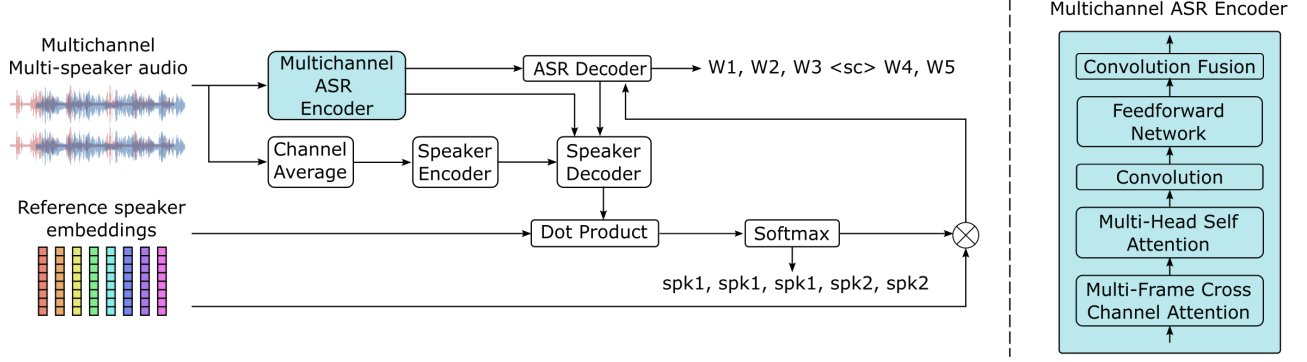


Fig. 1. Overview of the proposed MC-SA-ASR system (left) and its encoder (right).

## 2. RELATED WORKS

### 2.1. End-to-end single-channel SA-ASR

A Transformer-based end-to-end single-channel SA-ASR (SC-SA-ASR) system was proposed in [4]. Following the SOT principle, the output is the concatenation of all speakers' sentences, where each token is associated with one speaker ID and distinct speakers are separated by a <sc> token. The inputs to the model consist of an acoustic feature sequence  $X \in \mathbb{R}^{L \times A}$  where  $L$  is the sequence length and  $A$  the feature dimension, and a matrix  $S \in \mathbb{R}^{E \times K}$  of  $E$ -dimensional reference speaker embeddings obtained from enrollment data, each corresponding to one speaker  $k$  out of  $K$ . The feature sequence  $X$  is fed to the speaker encoder and the ASR encoder:

$$H^{\text{spk}} = \text{SpeakerEncoder}(X) \in \mathbb{R}^{T \times D}, \quad (1)$$

$$H^{\text{asr}} = \text{ASREncoder}(X) \in \mathbb{R}^{T \times D}. \quad (2)$$

The resulting embeddings  $H^{\text{spk}}$ ,  $H^{\text{asr}}$  and the  $n-1$  previous ASR output tokens  $\hat{y}_{[1:n-1]}$  are given to the speaker decoder to obtain a speaker posterior  $\hat{s}_n$  and a speaker profile  $\bar{s}_n$  associated with the  $n$ -th output token as follows:

$$q_n = \text{SpeakerDecoder}(\hat{y}_{[1:n-1]}, H^{\text{asr}}, H^{\text{spk}}) \in \mathbb{R}^E, \quad (3)$$

$$\hat{s}_n = \text{softmax}(S^\top q_n) \in \mathbb{R}^K, \quad (4)$$

$$\bar{s}_n = \sum S \hat{s}_n \in \mathbb{R}^E, \quad (5)$$

where  $(\cdot)^\top$  denotes matrix transposition. The ASR embedding  $H^{\text{asr}}$  and the speaker profile  $\bar{s}_n$  are provided to the ASR decoder to generate the  $n$ -th ASR output token:

$$\hat{y}_n = \text{ASRDecoder}(\hat{y}_{[1:n-1]}, H^{\text{asr}}, \bar{s}_n). \quad (6)$$

Given the ground truth token and speaker sequences, the training objective is to maximize the joint probability

$$P(Y, S | X, S) = \prod_{n=1}^N P(\hat{y}_n | \hat{y}_{[1:n-1]}, \hat{s}_{[1:n]}, X, S) \times P(\hat{s}_n | \hat{y}_{[1:n-1]}, \hat{s}_{[1:n-1]}, X, S). \quad (7)$$

### 2.2. Conformer-based multichannel ASR encoder

The Conformer-based multichannel ASR encoder is based on MFCCA [9], which is an attention mechanism that combines cross-channel and temporal context information. The  $h$ -th MFCCA head is computed as

$$Q_h = X W_h^q + (b_h^q)^\top \in \mathbb{R}^{T \times C \times D}, \quad (8)$$

$$K_h = \underline{X} W_h^k + (b_h^k)^\top \in \mathbb{R}^{T \times (2F+1)C \times D}, \quad (9)$$

$$V_h = \underline{X} W_h^v + (b_h^v)^\top \in \mathbb{R}^{T \times (2F+1)C \times D}, \quad (10)$$

$$H_h = \text{softmax}\left(\frac{Q_h(K_h)^\top}{\sqrt{D}}\right) V_h \in \mathbb{R}^{T \times C \times D}, \quad (11)$$

where  $C$  represents the number of channels,  $W_h^q$ ,  $W_h^k$ ,  $b_h^q$  and  $b_h^k$  are learnable parameters, and  $\underline{X} = [X^0, \dots, X^t, \dots, X^T]$  with  $X^t = [X^{t-F}, \dots, X^t, \dots, X^{t+F}] \in \mathbb{R}^{(2F+1)C \times D}$  the concatenation of  $F$  context frames at each time step  $t$ .

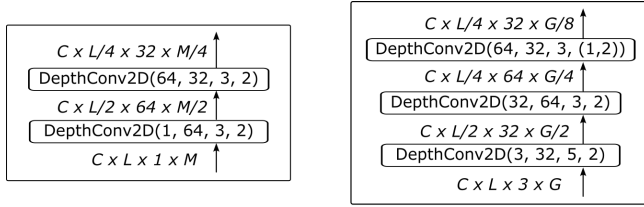
## 3. PROPOSED METHOD

The proposed MC-SA-ASR system is illustrated in Fig. 1. It draws inspiration from the end-to-end SC-SA-ASR model in Section 2.1, and uses reference speaker embeddings to guide the ASR decoder as in (6). However, instead of the single-channel encoder in Section 2.1, our system uses a modified version of the Conformer-based multichannel ASR encoder in Section 2.2. The ASR and speaker decoders remain the same as in the SC-SA-ASR model. In this section, we describe the elements that are specific to the proposed system.

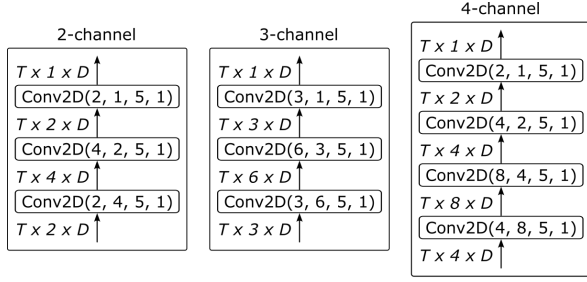
### 3.1. Input features for multichannel ASR

Multichannel speech signals contain spatial information that could be advantageous for discriminating speakers. Therefore, we investigate whether incorporating phase information can help our MC-SA-ASR model to generate more accurate multi-speaker transcriptions.

We consider two alternative sets of input features. On the one hand, we compute  $M$ -dimensional log Mel filterbank features from the STFT magnitude only. On the other hand, we concatenate the STFT magnitude and the cosine and sine of



**Fig. 2.** Depth-wise 2D-convolution feature extraction (input dimension, output dimension, kernel size, stride) for Mel filterbank (left) and magnitude+phase (right) features.



**Fig. 3.** Multichannel convolution fusion extended to 2, 3 and 4 input channels.

the phase, each with  $G$  dimension, into a  $3 \times G$ -dimensional representation, that is called the magnitude+phase feature. These features then undergo a specific processing, which is illustrated in Fig. 2 and described hereafter.

For the Mel filterbank, we apply depthwise separable convolution on each microphone. In the input array,  $C$  represents the number of microphones and  $L$  is the audio length. After two layers of 2-dimensional convolution, the output has a dimension of  $C \times T \times A$ , where  $T = L/4$  since each layer performs a sub-sampling of factor 2 over the time dimension, and  $A = 32 \times M/4$ , which is the output feature dimension.

The magnitude+phase features are processed using three layers of depthwise separable convolution. In the first layer, convolutional operations are used to fuse information across magnitude, cosine and sine values. The next two layers are similar to the ones used for the Mel filterbank, resulting in features with a dimension of  $C \times T \times A$ , similar to the Mel filterbank features, but with  $A = 32 \times G/8$ .

Finally, for both the Mel filterbank and magnitude+phase features, the output array of dimension  $C \times T \times A$  is passed to a linear layer yielding a representation array of dimension  $C \times T \times D$ , where  $D$  is the model dimension which is the same for both input features.

### 3.2. Convolution fusion

Convolution fusion serves as the output layer of the multichannel ASR encoder (see Fig. 1). It combines the representations corresponding to the multiple input channels. We extend the convolution fusion from [9] to support 2, 3, and 4 channel input as illustrated in Fig. 3.

### 3.3. Speaker encoder

Mel filterbank features are first averaged across all channels and then fed to our speaker embedding model. We use an x-vector speaker embedding model based on the emphasized channel attention, propagation and aggregation in TDNN (ECAPA-TDNN) [16]. In order to align the dimension of the x-vectors with our model architecture, we replace the final (average pooling) layer by a linear layer.

## 4. EXPERIMENTS

### 4.1. Evaluation on mixtures of LibriSpeech data

#### 4.1.1. Dataset and metrics

We simulate a multi-speaker scenario using the LibriSpeech dataset [17]. The train-960, dev-clean and test-clean subsets are used to generate our train, dev and test sets, respectively. We assume a linear 2- to 4-microphone array with an aperture of 10 cm. Room impulse responses (RIRs) aperture are generated by the image source method (ISM) using the gpuRIR toolkit [18]. The length, width and the height of the room are randomly drawn between 3 and 8 m and between 2.4 and 3 m, respectively. The array centers and speaker positions are randomly sampled with the constraint that the array center are at most 0.5 m away from the room center, the height in the range of 0.6 to 0.8 m, and the speakers should be at least 0.5 m away from the walls. The RT60 value is fixed in the range [0.4, 1]. The number of speakers in each sentence is randomly chosen between 1 and 3. The speaker embeddings directory includes 8 speakers, and the speaker IDs are also randomly generated. In order to simulate multi-speaker scenarios, the second and third speakers signals are delayed relative to the previous speaker. This realistic choice also guarantees the first-in first-out (FIFO) principle behind SOT [19]. The sentences of each speaker are concatenated together and separated from the next speaker using the <sc> token.

We use the word error rate (WER) to evaluate the ASR task, and the token-level speaker error rate (T-SER) and sentence-level speaker error rate (S-SER) [20] to evaluate the speaker prediction task. In order to calculate the S-SER, a single speaker is assigned to each hypothesized sentence by taking a max over token-level speaker prediction counts within the respective sentences segmented by <sc>.

#### 4.1.2. Baseline

We choose two baseline systems to study the impact of a multichannel encoder that effectively utilizes speaker information. The first baseline is the end-to-end SC-SA-ASR system proposed in [4]. It helps us to compare the impact of a multichannel MFCCA-based ASR encoder on far-field speech. Our implementation of the SC-SA-ASR system uses ECAPA-TDNN speaker embeddings. The second baseline is the MFCCA-based multichannel ASR (MC-ASR) model

**Table 1.** WER (%), sentence-level SER (S-SER) (%) and token-level SER (T-SER) (%) on the simulated multichannel multi-speaker LibriSpeech test set. Results are grouped by the number of speakers in the simulated mixture. The ‘1,2,3-speaker mix’ column shows the results obtained on a test set containing mixtures of 1, 2, and 3 speakers.

| System                 | Input features | # Channels            | 1-speaker   |             |              | 2-speaker mix |             |              | 3-speaker mix |             |              | 1,2,3-speaker mix |              |       | #Param. |
|------------------------|----------------|-----------------------|-------------|-------------|--------------|---------------|-------------|--------------|---------------|-------------|--------------|-------------------|--------------|-------|---------|
|                        |                |                       | WER         | S-SER       | T-SER        | WER           | S-SER       | T-SER        | WER           | S-SER       | T-SER        | WER               | S-SER        | T-SER |         |
| <u>Baseline Models</u> |                |                       |             |             |              |               |             |              |               |             |              |                   |              |       |         |
| SC-SA-ASR [4]          | Mel filterbank | 1                     | 7.15        | 1.64        | 3.13         | 14.68         | 3.23        | 5.42         | 20.39         | <b>5.59</b> | 7.67         | 16.81             | <b>3.95</b>  | 6.18  | 61.1M   |
|                        |                | 2                     | 8.52        | -           | -            | 16.76         | -           | -            | 22.87         | -           | -            | 18.21             | -            | -     |         |
|                        |                | 3                     | 8.29        | -           | -            | 16.54         | -           | -            | 22.01         | -           | -            | 18.03             | -            | -     | 27.8M   |
| MC-ASR [9]             | Mel filterbank | 4                     | 8.11        | -           | -            | 16.43         | -           | -            | 21.76         | -           | -            | 17.84             | -            | -     |         |
|                        |                | <u>Proposed Model</u> |             |             |              |               |             |              |               |             |              |                   |              |       |         |
|                        |                | Mel filterbank        | 2           | 6.64        | 1.15         | 2.84          | 14.21       | 3.19         | 5.14          | 19.03       | 5.92         | 7.34              | <b>15.41</b> | 4.34  | 6.46    |
| 3                      | <b>6.62</b>    |                       | <b>1.11</b> | <b>2.73</b> | 14.24        | 3.15          | 5.56        | 18.92        | 6.11          | 7.29        | 15.25        | 4.12              | 6.16         |       |         |
| 4                      | 7.17           |                       | 1.41        | 2.95        | 14.09        | <b>3.06</b>   | <b>5.10</b> | 18.70        | 6.06          | <b>6.93</b> | 14.77        | 4.05              | 6.40         |       |         |
| Magnitude+phase        | 2              | 7.24                  | 1.53        | 3.30        | 15.42        | 3.85          | 6.22        | 20.34        | 7.06          | 7.98        | 16.76        | 4.87              | 7.29         |       |         |
|                        | 3              | 6.86                  | 1.78        | 3.39        | <b>13.69</b> | 3.72          | 5.39        | 18.14        | 6.86          | 7.83        | <b>15.04</b> | 4.08              | 6.28         |       |         |
|                        | 4              | 6.91                  | 1.86        | 3.43        | 13.97        | 3.10          | 5.85        | <b>18.03</b> | 6.69          | 7.26        | <b>14.69</b> | 4.12              | <b>5.85</b>  |       |         |

proposed in [9]. The MC-ASR model has been extended to perform speaker identification in the MC-WD-SOT model presented in [8]. However, the MC-WD-SOT model does not incorporate any speaker information in the ASR decoder. Hence, the performance of the ASR module in the MC-WD-SOT model is not expected to be better than that of the MC-ASR model. Thus, in favor of a simpler implementation, we choose the MC-ASR model as the second baseline.

#### 4.1.3. Model and training setup

We compute STFT with a window length of 25 ms and a hop size of 10 ms. The magnitude information is used to generate 80-dimensional log Mel filterbank features. For the magnitude and phase features, we generate a  $3 \times 201$ -dimensional feature that includes magnitude, cosine, and sine phases. According to the convolution process in Fig. 2, the convolutional feature extractor produces features  $A$  of size 640 and 832 for the Mel filterbank features and phase-wise features, respectively. The speaker encoder uses 80-dimensional log Mel filterbank features averaged across all channels.

For all the models (SC-SA-ASR, MC-ASR and MC-SA-ASR) in our experiments, the Conformer-based encoder has 12 layers, and the Transformer-based decoder has 1 layer. The speaker decoder in SC-MC-ASR and MC-SA-ASR has 2 layers. All multi-head attention mechanisms have 4 heads, the model dimension  $D$  is set to 256, and the size of the feedforward layer is 2048. Following [9], the context frame length  $F$  of MFCCA in MC-ASR and SA-MC-ASR is set to 2. Our text tokenizer is a SentencePiece model [21] with a vocabulary of 5000 tokens. The speaker embedding model is an ECAPA-TDNN model pre-trained on the VoxCeleb1 [22] and VoxCeleb2 [23] training data, generating 192 dimension embeddings. In each reference speaker embedding matrix  $S$ , the number of speakers  $K$  is set to 8, and the embedding for each speaker is derived from two random enrollment sentences.

Our experiments were implemented using the SpeechBrain toolkit [24]. All the models were trained until conver-

gence. The ASR modules in SC-SA-ASR and MC-SA-ASR were pre-trained for 80 epochs by setting  $S$  and  $H^{\text{spk}}$  to 0. We utilized the Adam optimizer with a learning rate of  $5 \times 10^{-4}$  during the pre-training process. Subsequently, the ASR and speaker modules of the SC-SA-ASR model were fine-tuned for 60 epochs using a learning rate of  $2.5 \times 10^{-4}$ . The MC-SA-ASR model was fine-tuned for 120 epochs using a learning rate of  $1.5 \times 10^{-5}$ , after the initial training of 60 epochs. The weight of the speaker loss was set to 0.1, following prior work [20]. The MC-ASR model was trained for 140 epochs with a learning rate of  $5 \times 10^{-4}$ . For all the experiments, the global batch size (batch size  $\times$  number of GPUs  $\times$  gradient accumulation factor) is fixed to 160.

#### 4.1.4. Results and discussion

Table 1 presents the test results of the baseline models (SC-SA-ASR and MC-ASR) and the proposed model (MC-SA-ASR). First of all, by comparing the results of SC-SA-ASR and MC-SA-ASR, we can conclude that using a multichannel encoder to process multi-microphone speech information improves the speech recognition performance. Specifically, the MC-SA-ASR model with Mel filterbank input features achieves a WER of 14.77% in the 4-channel scenario, that is a 12% relative reduction compared to the SC-SA-ASR model (16.81%). On average, the WER of the MC-SA-ASR models with 2, 3, and 4 channels (15.14%) is reduced by 10% relative compared to the SC-SA-ASR model. Secondly, the proposed MC-SA-ASR model, which incorporates speaker information into the ASR decoder, obtains a 16% relative reduction in WER compared to MC-ASR (18.03%). This suggests that leveraging speaker information can improve the performance of multi-speaker ASR in a multichannel setting.

The test WERs obtained by the proposed MC-SA-ASR model with Mel filterbank input features and with magnitude+phase input features do not exhibit significant differences. However, interestingly, in the 2-channel scenarios, Mel filterbank features (15.41%) outperformed magnitude+phase

(16.76%) with a 8% relative lower WER. Conversely, in the test results of the 3- and 4-channel scenarios, magnitude+phase features achieved a slight relative reduction of 1.4% and 0.5% in WER, respectively. This leads us to speculate that phase features may result in better ASR performance in models with a larger number of channels. However, further testing is required to validate this conclusion. Moreover, in the test results of the 3-channel scenarios, magnitude+phase features exhibited a relative reduction of 4% in WER on both 2 and 3-speaker-mixed datasets compared to Mel filterbank features. From this, we can conclude that in multi-speaker multichannel ASR, phase features perform better on chunks with a larger number of speakers. This can be explained by the fact that as the number of speakers in a room increases, the positional information of the speakers has a greater impact on ASR performance.

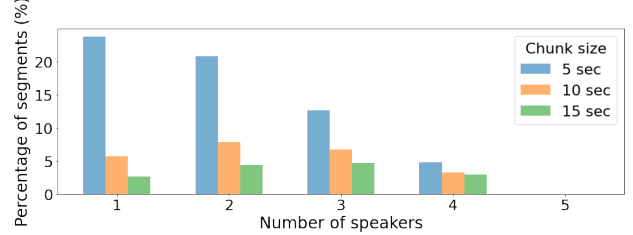
## 4.2. Evaluation on AMI

We validate the effectiveness of the proposed MC-SA-ASR model on real-world data by fine-tuning and testing our pre-trained model on the AMI meeting corpus [25].

### 4.2.1. Corpus preparation

The AMI multiple distant microphone (MDM) corpus consists of approximately 100 hours of 8- to 16-channel audio recordings of 3 to 5 participants in meetings. The data is annotated in terms of ASR and diarization, with the start and end timestamps for each sentence. In order to process the meeting files, which typically consist of approximately 1 hour of content, we have to divide each meeting into smaller segments. We adopt a segmentation approach inspired by “utterance groups” [26] which works as follows. (a) Segment each meeting using a chunk size of  $b$  seconds and a hop size of  $o$  seconds. (b) If the start/end time of a segment falls within a region involving more than one speaker, it is adjusted to be two seconds outside the overlap region. (c) If the start/end time of a segment falls within a word, it is adjusted to align with the start/end of that word. The benefits of segmenting in this manner are twofold. Firstly, the utilization of a hop size allows for an increased number of training samples. Secondly, segmenting outside the speaker overlap regions respects the FIFO training approach.

We conducted experiments with chunk size of 5, 10 or 15 s, with the hop size set to 2 s. Figure 4 illustrates the distribution of the number of segments and speakers in the datasets generated using different chunk sizes. Table 2 presents some statistics of the training, development, and test sets generated with a chunk size of 5 s. It is observed that the probability of a chunk containing multiple speakers increases as the chunk size increases. To evaluate the model’s performance on datasets with varying numbers of speakers, we combined all the test sets segmented at 5, 10, and 15 s. We then divided them into four different test sets based on the number of



**Fig. 4.** Percentage of segments containing a given number of speakers for different chunk sizes on the AMI corpus.

**Table 2.** AMI statistics after segmentation (chunks size of 5 seconds). The average duration is in seconds (s), and the total duration is in hours (h).

| Training set    |        |        |        |        |      |         |
|-----------------|--------|--------|--------|--------|------|---------|
| # speakers      | 1      | 2      | 3      | 4      | 5    | total   |
| # segments      | 34719  | 30304  | 18492  | 6900   | 10   | 90425   |
| Avg. dur. (s)   | 5.62   | 6.48   | 7.74   | 8.93   | 9.43 | 6.59    |
| Total dur. (h)  | 54.22  | 54.569 | 39.77  | 17.11  | 0.02 | 165.71  |
| # words         | 444656 | 592651 | 520644 | 252017 | 434  | 1810402 |
| Development set |        |        |        |        |      |         |
| # speakers      | 1      | 2      | 3      | 4      |      | total   |
| # segments      | 4104   | 3357   | 2362   | 988    |      | 10811   |
| Avg. dur. (s)   | 5.55   | 6.38   | 7.45   | 8.45   |      | 6.49    |
| Total dur. (h)  | 6.33   | 5.95   | 4.88   | 2.31   |      | 19.50   |
| # words         | 52730  | 65072  | 65575  | 35803  |      | 219180  |
| Test set        |        |        |        |        |      |         |
| # speakers      | 1      | 2      | 3      | 4      |      | total   |
| # segments      | 4211   | 3035   | 2129   | 858    |      | 10233   |
| Avg. dur. (s)   | 5.57   | 6.44   | 7.90   | 9.34   |      | 6.63    |
| Total dur. (h)  | 6.52   | 5.43   | 4.67   | 2.22   |      | 18.86   |
| # words         | 52765  | 59624  | 64769  | 33500  |      | 210658  |

speakers, which include datasets with 1, 2, 3 and 4 speakers, respectively. The number of segments for each speaker count is as follows: 5,737 segments with 1 speaker, 4,911 segments with 2 speakers, 3,986 segments with 3 speakers, and 1,936 segments with 4 speakers.

### 4.2.2. Fine-tune settings

We fine-tune the pre-trained SC-SA-ASR and MC-SA-ASR models on the AMI MDM datasets using the Full-corpus-ASR partitions. SC-SA-ASR utilizes only the 1st channel of Array 1<sup>1</sup> of the train-dev-test splits; while MC-SA-ASR utilizes the 1st and 5th channels for the 2-channel model. The datasets with chunk sizes of 5, 10, and 15 s undergo fine-tuning for 40 and 90 epochs, respectively. In each case, the first half of all training epochs updates the ASR module, while the second half jointly updates the ASR and speake modules. All fine-tuning steps employ the Adam optimizer with a learning rate of  $1 \times 10^{-4}$ , and a global batch size of 160.

### 4.2.3. Results and discussion

Table 3 presents the test results of SC-SA-ASR and MC-SA-ASR on datasets segmented using different chunk sizes.

<sup>1</sup>Few meetings have two arrays, each array consisting of 8 microphones.

**Table 3.** WER (%), sentence-level SER (S-SER) (%) and token-level SER (T-SER) (%) of models adapted to AMI when chunk size (5, 10 or 15 s) is consistent across train, dev and test splits. Our proposed MC-SA-ASR uses 2 channels. Size represents # of model parameters in millions.

| System        | 5 s          |              |              | 10 s         |              |              | 15 s         |              |              |
|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|               | WER          | S-SER        | T-SER        | WER          | S-SER        | T-SER        | WER          | S-SER        | T-SER        |
| SC-SA-ASR [4] | 46.03        | <b>33.86</b> | <b>28.54</b> | 46.54        | 40.88        | 28.95        | 48.78        | 45.93        | 30.87        |
| MC-SA-ASR     | <b>45.15</b> | 34.14        | 29.08        | <b>45.97</b> | <b>39.93</b> | <b>28.00</b> | <b>48.49</b> | <b>44.58</b> | <b>28.56</b> |

**Table 4.** WER (%) and token-level SER (T-SER) (%) of models adapted to AMI when chunk sizes are different across train and test splits. Rows 5 s, 10 s and 15 s refer to the chunk size in train-dev splits. The test set consists of 5, 10 and 15 s chunks. MC-SA-ASR uses 2 channels.

| System        | 1-spkr       |              | 2-spkr mix   |              | 3-spkr mix   |              | 4-spkr mix   |              |
|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|               | WER          | T-SER        | WER          | T-SER        | WER          | T-SER        | WER          | T-SER        |
| SC-SA-ASR [4] |              |              |              |              |              |              |              |              |
| 5 s           | 28.10        | <b>13.37</b> | 42.03        | 25.53        | 54.04        | 35.65        | 67.27        | 43.17        |
| 10 s          | 31.84        | 17.25        | 43.93        | 27.88        | 54.20        | 35.82        | 67.92        | 43.49        |
| 15 s          | 48.38        | 32.38        | 60.73        | 43.12        | 64.39        | 44.53        | 73.68        | 49.40        |
| MC-SA-ASR     |              |              |              |              |              |              |              |              |
| 5 s           | <b>27.68</b> | 13.43        | <b>39.54</b> | <b>24.83</b> | <b>52.76</b> | 35.70        | <b>64.72</b> | <b>41.24</b> |
| 10 s          | 31.81        | 17.52        | 43.77        | 27.18        | 53.92        | <b>34.99</b> | 66.54        | 42.80        |
| 15 s          | 48.69        | 31.79        | 60.34        | 42.48        | 63.66        | 44.19        | 72.69        | 48.36        |

MC-SA-ASR consistently outperforms SC-SA-ASR across all datasets. Particularly, with a chunk size of 5 s, 2-channel MC-SA-ASR achieves a relative 2% reduction in WER compared to SC-SA-ASR. We would also like to highlight that our proposed MC-SA-ASR model exhibits a 16% relative reduction in model size compared to the SC-SA-ASR model which has 61M parameters. This reduction comes by replacing the first (feedforward) layer of the original Conformer by a smaller-sized MFCCA.

We also compare the performance of models trained on different chunk sizes on all the test sets. Table 4 presents the results, divided into 4 subsets based on the number of speakers. We can observe that 2-channel MC-SA-ASR consistently achieves a lower WER than SC-SA-ASR. Particularly, the 5-second model demonstrates a 6% relative reduction in WER compared to SC-SA-ASR on the 2-speaker test set. Moreover, we observe that models trained on smaller chunk sizes perform better. In MC-SA-ASR, the 5-second model exhibits relative reductions in WER of 43%, 34%, 17%, and 11% on the test sets with 1, 2, 3 and 4 speakers compared to the 15-second model, respectively (note that the SER exhibits a similar trend). This might be explained by the fact that, according to Fig. 4, when decreasing the number of speakers, both the total number and the proportion of 5-second segments in the training set increase compared to 15-second segments.

A comparison of S-SER versus T-SER performances in Table 3 shows that S-SER is much higher than T-SER on the AMI test set, for both SC-SA-ASR and MC-SA-ASR sys-

**Table 5.** Speaker counting accuracy (%) on the total of 5, 10 and 15 seconds test chunks of AMI for models trained on chunks of 5 seconds.

| System        | # Speakers | Estimated # speakers |              |              |              |      |
|---------------|------------|----------------------|--------------|--------------|--------------|------|
|               |            | 1                    | 2            | 3            | 4            | >4   |
| SC-SA-ASR [4] | 1          | 91.75                | 7.53         | 0.64         | 0.06         | 0.00 |
|               | 2          | 11.26                | 76.82        | 10.73        | 1.03         | 0.14 |
|               | 3          | 1.75                 | 36.67        | 50.12        | 8.83         | 2.60 |
|               | 4          | 0.36                 | 21.02        | 51.65        | 19.31        | 7.64 |
| MC-SA-ASR     | 1          | <b>92.40</b>         | 6.98         | 0.48         | 0.06         | 0.05 |
|               | 2          | 11.57                | <b>77.05</b> | 10.71        | 0.54         | 0.10 |
|               | 3          | 1.80                 | 37.18        | <b>50.74</b> | 8.50         | 1.75 |
|               | 4          | 0.82                 | 18.59        | 53.87        | <b>20.04</b> | 6.66 |

tems, as opposed to lower S-SER and higher T-SER values on the LibriSpeech test set in Table 1. The reason is that the prediction of speaker change markers <sc> is a more challenging task on the AMI test set. As a further analysis, we evaluate the performance of the two systems in terms of speaker counting task on the AMI test set. Table 5 presents the speaker counting accuracy obtained by counting the occurrences of <sc> tokens in the ASR output, on datasets with different numbers of speakers. We observe that the MC-SA-ASR system is consistently outperforming the SC-SA-ASR system. However, the accuracy decreases by 60% relative from the 3-speaker test set to the 4-speaker set for MC-SA-ASR (61% for SC-SA-ASR). Furthermore, for scenarios involving 2, 3, and 4 speakers, the majority of errors originate from underestimating the number of speakers.

## 5. CONCLUSION

In this paper, we have introduced an end-to-end MC-SA-ASR system that combines a Conformer-based encoder with multi-frame cross-channel attention and a speaker-attributed Transformer-based decoder. Experimental results demonstrate that, on simulated data, our approach achieves relative reductions in WER of up to 12% and 16% compared to existing single-channel and multichannel methods, respectively. We also studied the impact of using Mel filterbank vs. magnitude+phase features on MC-SA-ASR. On real-world data, our model achieves a relative reduction in WER of up to 6% compared to SC-SA-ASR. However, it still has limitations in accurately determining the number of speakers in scenarios involving three or more participants. Future research can focus on improving this aspect.

## 6. ACKNOWLEDGMENTS

Experiments presented in this paper were carried out using (a) the Grid’5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see <https://www.grid5000.fr>), and (b) HPC resources from GENCI-IDRIS (Grant 2023-[AD011013881]).

## 7. REFERENCES

- [1] Pengcheng Guo, Xuankai Chang, Shinji Watanabe, and Lei Xie, “Multi-speaker ASR combining non-autoregressive Conformer CTC and conditional speaker chain,” in *INTERSPEECH 2021*, 2021, pp. 1401–1405.
- [2] Liang Lu, Naoyuki Kanda, Jinyu Li, and Yifan Gong, “Streaming multi-talker speech recognition with joint speaker identification,” in *INTERSPEECH 2021*, 2021, pp. 1782–1782.
- [3] Ilya Sklyar, Anna Piunova, and Yulan Liu, “Streaming multi-speaker ASR with RNN-T,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6903–6907.
- [4] Naoyuki Kanda, Guoli Ye, Yashesh Gaur, Xiaofei Wang, Zhong Meng, Zhuo Chen, and Takuya Yoshioka, “End-to-end speaker-attributed ASR with Transformer,” in *INTERSPEECH 2021*, 2021, pp. 4413–4417.
- [5] Xuankai Chang, Wangyou Zhang, Yanmin Qian, Jonathan Le Roux, and Shinji Watanabe, “Mimo-speech: End-to-end multi-channel multi-speaker speech recognition,” in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 237–244.
- [6] Xuankai Chang, Wangyou Zhang, Yanmin Qian, Jonathan Le Roux, and Shinji Watanabe, “End-to-end multi-speaker speech recognition with Transformer,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6134–6138.
- [7] Robin Scheibler, Wangyou Zhang, Xuankai Chang, Shinji Watanabe, and Yanmin Qian, “End-to-end multi-speaker ASR with independent vector analysis,” in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 496–501.
- [8] Mohan Shi, Jie Zhang, Zhihao Du, Fan Yu, Shiliang Zhang, and Li-Rong Dai, “A comparative study on multi-channel speaker-attributed automatic speech recognition in multi-party meetings,” *arXiv e-prints*, pp. arXiv–2211, 2022.
- [9] Fan Yu, Shiliang Zhang, Pengcheng Guo, Yuhao Liang, Zhihao Du, Yuxiao Lin, and Lei Xie, “MFCCA: Multi-frame cross-channel attention for multi-speaker ASR in multi-party meeting scenario,” in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 144–151.
- [10] Tuo Zhao, Yunxin Zhao, Shaojun Wang, and Mei Han, “Unet++-based multi-channel speech dereverberation and distant speech recognition,” in *12th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2021, pp. 1–5.
- [11] Yiwen Shao, Shi-Xiong Zhang, and Dong Yu, “Multi-channel multi-speaker ASR using 3D spatial feature,” in *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6067–6071.
- [12] Zhuo Chen, Xiong Xiao, Takuya Yoshioka, Hakan Erdogan, Jinyu Li, and Yifan Gong, “Multi-channel overlapped speech recognition with location guided speech extraction network,” in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 558–565.
- [13] Rongzhi Gu, Shi-Xiong Zhang, Yong Xu, Lianwu Chen, Yuexian Zou, and Dong Yu, “Multi-modal multi-channel target speech separation,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 3, pp. 530–541, 2020.
- [14] Zhuohuang Zhang, Yong Xu, Meng Yu, Shi-Xiong Zhang, Lianwu Chen, and Dong Yu, “ADL-MVDR: All deep learning MVDR beamformer for target speech separation,” in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6089–6093.
- [15] Feng-Ju Chang, Martin Radfar, Athanasios Mouchtaris, Brian King, and Siegfried Kunzmann, “End-to-end multi-channel Transformer for speech recognition,” in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5884–5888.
- [16] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, “ECAPA-TDNN: emphasized channel attention, propagation and aggregation in TDNN based speaker verification,” in *Interspeech 2020*, 2020, pp. 3830–3834.
- [17] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.
- [18] David Diaz-Guerra, Antonio Miguel, and Jose R. Beltran, “gpuRIR: A Python library for room impulse response simulation with GPU acceleration,” *Multimedia Tools and Applications*, vol. 80, pp. 5653–5671, 2021.
- [19] Naoyuki Kanda, Yashesh Gaur, Xiaofei Wang, Zhong Meng, and Takuya Yoshioka, “Serialized output training for end-to-end overlapped speech recognition,” in *Interspeech 2020*, 2020, pp. 2797–2801.



- [20] Naoyuki Kanda, Yashesh Gaur, Xiaofei Wang, Zhong Meng, Zhuo Chen, Tianyan Zhou, and Takuya Yoshioka, “Joint speaker counting, speech recognition, and speaker identification for overlapped speech of any number of speakers,” in *Interspeech 2020*, 2020, pp. 36–40.
- [21] Taku Kudo and John Richardson, “Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing,” in *2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 66–71.
- [22] Joon Son Chung, Arsha Nagrani, Ernesto Coto, Weidi Xie, Mitchell McLaren, Douglas A. Reynolds, and Andrew Zisserman, “VoxSRC 2019: The first VoxCeleb speaker recognition challenge,” *arXiv preprint arXiv:1912.02522*, 2019.
- [23] Arsha Nagrani, Joon Son Chung, Jaesung Huh, Andrew Brown, Ernesto Coto, Weidi Xie, Mitchell McLaren, Douglas A. Reynolds, and Andrew Zisserman, “VoxSRC 2020: The second VoxCeleb speaker recognition challenge,” *arXiv e-prints*, pp. arXiv–2012, 2020.
- [24] Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio, “SpeechBrain: A general-purpose speech toolkit,” 2021, arXiv:2106.04624.
- [25] Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al., “The AMI meeting corpus: A pre-announcement,” in *International Workshop on Machine Learning for Multimodal Interaction*, 2005, pp. 28–39.
- [26] Naoyuki Kanda, Guoli Ye, Yu Wu, Yashesh Gaur, Xiaofei Wang, Zhong Meng, Zhuo Chen, and Takuya Yoshioka, “Large-Scale Pre-Training of End-to-End Multi-Talker ASR for Meeting Transcription with Single Distant Microphone,” in *Interspeech 2021*, 2021, pp. 3430–3434.