Multi-channel extension of pre-trained models for speaker verification

Ladislav Mošner, Romain Serizel, Lukáš Burget, Oldřich Plchot, Emmanuel Vincent, Junyi Peng, Jan Černocký

Multi-Channel Extension of Pre-trained Models for Speaker Verification

Ladislav Mošner[1], Romain Serizel[2], Lukáš Burget[1], Oldřich Plchot[1], Emmanuel Vincent[2], Junyi Peng[1], Jan Černocký[1]

[1]Brno University of Technology, Faculty of Information Technology, Speech@FIT, Czechia
[2]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

{imosner, iplchot}@fit.vutbr.cz, romain.serizel@loria.fr, emmanuel.vincent@inria.fr

Abstract

In this work, we focus on designing a multi-channel speech processing system based on large pre-trained models. These models are typically trained for single-channel scenarios via self-supervised learning (SSL). A common approach to using the SSL models with microphone array data is to prepend it with a multi-channel speech enhancement. The downside is that spatial information can be leveraged only by the pre-processing stage, and enhancement errors get propagated to the SSL model. We aim to alleviate the issue by designing METRO, a Multi-channel ExTension of pRe-trained mOdels. It interleaves per-channel processing with cross-channel information exchange, eventually fusing channels into one. While our approach is general, here we focus on multi-channel speaker verification. Our experiments on the MultiSV corpus show noteworthy improvements over the best-published results on the dataset.

Index Terms: multi-channel speaker verification, pre-trained models

1. Introduction

Recognition of speech or speakers from far-field microphones is hindered by adverse conditions caused by reverberation and noise. A common practice to alleviate the harmful effect of such distortions is to leverage multiple microphones (channels) providing spatial information. This helps to discriminate between desired and undesired sources occupying different spatial locations. In the same vein, various hands-free commercial devices comprise microphone arrays. As speaker verification (SV) is essential to personalization or enhanced authentication, multi-channel SV is an important field of study.

Current approaches to multi-channel SV fall into three categories: using multi-channel pre-processing [1–5], training tailored architectures from scratch [6, 7], or extending single-channel models [8–10]. Multi-channel pre-processing is the most common. Despite its greater interpretability and modularity, it is limited by the loose correlation between the enhancement and downstream objectives [11]. Joint fine-tuning [5, 12] improves performance, but only partly alleviates this limitation. The mentioned tailored architectures, by contrast, require substantial volumes of training data. Moreover, they are data-specific since it is the network itself that models channel interdependencies. As such, they cannot accommodate a variable number of channels. Existing studies [6, 7] consider circular arrays with a fixed number of sensors and limited radius variations. Our approach falls in the third category of designs which extend single-channel models. They usually start from a strong single-channel speaker embedding extractor which processes channels independently up to a specific layer. Subsequently, modules that fuse information across channels are introduced.

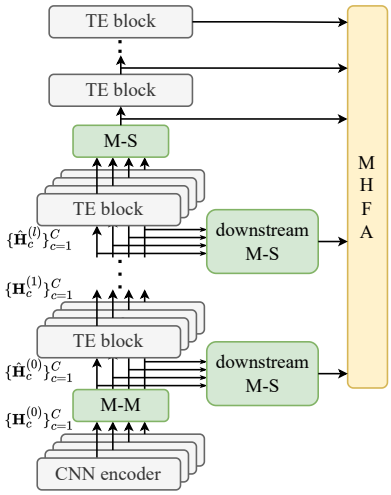

Figure 1: *Designed multi-channel extension of SSL models (*METRO*) for speaker verification. TE stands for Transformer encoder, MHFA is multi-head factorized attentive pooling [13].*

The information can be fused at the utterance level [8, 9] or at the frame level [10], where the latter was shown to be superior.

Self-supervised learning (SSL) has also attracted significant attention in speech signal processing [14–16]. It allows for training models on large-scale audio data without labels. SSL models are typically trained on clean data, which makes them less suitable for signals corrupted by room conditions. Recently, studies extending them to different flavours of data have been published [16, 17]. For instance, WavLM [16], among other modifications over HuBERT [15], was designed to be noise-robust. The resulting general representations of speech were shown to be useful for many tasks (including SV) in the SUPERB benchmark [18], where the SSL model is frozen. Recent studies [13,16,19,20] have demonstrated an impressive SV performance when it is fine-tuned with the downstream models.

Commonly available SSL models are single-channel by design. Inspired by the robust performance of models leveraging pre-trained speech representations, attempts to utilize them in multi-channel settings have emerged. All the top CHiME-7 Challenge systems [21], as well as [22,23], are based on cascading multi-channel beamforming/enhancement, an SSL model, and (end-to-end) single-channel automatic speech recognition. A few works have explicitly extended the SSL framework to multi-channel models. Spatial HuBERT [24] is an SSL model built upon WavLM. It extends a CNN encoder to its multi-channel version. Multi-channel AV-wav2vec2 [25] is an SSL model for multi-channel, multi-modal data, which fuses channels and modalities before the Transformer encoder blocks.

None of the mentioned models was publicly released.

In this paper, we design METRO, a general approach to extend single-channel SSL models to multi-channel data. Despite focusing on SV in this paper, our design is applicable to other speech tasks. It replicates the original blocks to perform parallel processing of channels up to a certain level. Then, the channels are fused, and the original SSL architecture continues. Importantly, we introduce new modules after each parallel block to promote information exchange across channels. We stress the importance of properly initializing these new modules to avoid corrupting the forwarded data. Our approach achieves the best-published results on the MultiSV corpus [26]. By building upon various concepts, we design a holistic solution for multi-channel SV: 1) Inspired by its robust speech representations, we employ WavLM [16] as a backbone for our model. 2) Among various approaches to using SSL features for SV, we selected multi-head factorized attentive pooling (MHFA) [13] as it is lightweight yet competitive. 3) We adopt the idea of extending a single-channel model while fusing channel-wise information at the frame level [10]. 4) Multi-channel architectures interleaving cross-channel and temporal per-channel processing have become successful in speech separation [27, 28] and diarization [29]. We reach the same structure by introducing new modules to SSL models (see Figure 1). Code is available at `https://github.com/BUTSpeechFIT/Wespeaker_MC_SSL`.

2. Method

To extend SSL models to multi-channel data, we replicate the original blocks (CNN encoder, Transformer encoder blocks) to allow for parallel processing. The same transformation is thus applied to all channels due to weight sharing. As shown in [30], global information exchange provides clear benefits in speech enhancement/separation tasks. Following this idea, we append modules that promote cross-channel information fusion to the replicated original blocks. We introduce two types of such modules: *M-M* and *M-S*. While both have multi-channel input, M-M has a multi-channel output and M-S provides a single-channel output. As displayed in Figure 1, M-M modules are inserted after each set of blocks until a specific layer. After the integration of the M-S module, the original single-channel structure follows. The resulting architecture resembles that of [27, 28], interleaving temporal per-channel processing with cross-channel processing. Our architecture, however, differs in that it builds upon the SSL model and eventually fuses parallel branches into a single one, which is computationally efficient and provides empirically corroborated performance improvements.

To perform speaker embedding extraction for SV, we employ a lightweight downstream model: multi-head factorized attentive pooling (MHFA) [13]. It was designed such that the outputs of CNN encoder and Transformer encoder blocks form the input to MHFA. In our multi-channel model, we propose to use M-M or M-S outputs instead. While the M-S outputs do not require special treatment, the M-M outputs need to be compressed to a single channel via a *downstream M-S*. This approach allows for using the MHFA without modifications.

2.1. M-M Modules

The M-M modules allow information exchange between channels. We base their design on approaches from the literature [29, 30]. We note that we also experimented with cross-channel attention [27] but found it inferior compared to the presented alternatives. As it will follow from the description, utilized designs are agnostic to the number of channels.

Let $\mathbf{H}_c \in \mathbb{R}^{T\times D}$ be the output of the CNN or Transformer encoder block corresponding to the c-th channel, where T and D represent the number of time frames and the feature dimensionality, respectively. Despite layer dependency, we omit the layer index $(\cdot)^{(l)}$ (see Figure 1) to ease notation. Let us define $f_{\mathrm{FC}}(\mathbf{x};\alpha) := \sigma(\mathbf{x}\mathbf{W}+\mathbf{b})$, where $\alpha = \{\mathbf{W},\mathbf{b}\}$ and $\sigma(\cdot)$ is a suitable non-linearity. Moreover, let $f_{\mathrm{MA}}(\mathbf{X},\mathbf{Y};\beta) := \left[\Big\|_{h=1}^{H} f_{\mathrm{AT}}\left(\mathbf{X}\mathbf{W}_Q^{(h)}, \mathbf{X}\mathbf{W}_K^{(h)}, \mathbf{Y}\mathbf{W}_V^{(h)}\right)\right]\mathbf{W}_O$ be the H-head attention, where $f_{\mathrm{AT}}(\mathbf{Q},\mathbf{K},\mathbf{V})$ is the dot-product attention defined in Eq. (1) of [31], the operator $||$ concatenates its arguments along the last dimension, and $\beta = \{\mathbf{W}_Q^{(h)}, \mathbf{W}_K^{(h)}, \mathbf{W}_V^{(h)}\}_{h=1}^{H} \cup \{\mathbf{W}_O\}$. The input matrix $\mathbf{X}$ defines queries and keys, hence attention weights.

TAC The transform-average-concatenate (TAC) module presented in [30] averages frame-level per-channel representations into a single-channel representation in a $\bar{d}$-dimensional space parameterized by θ. The resulting global representation

$$\mathbf{T} = f_{\mathrm{FC}}\left(\frac{1}{C}\sum_{c=1}^{C} f_{\mathrm{FC}}(\mathbf{H}_c;\theta);\phi\right) \tag{1}$$

is merged with the per-channel one as

$$\bar{\mathbf{T}}_c = \mathrm{LN}\left(f_{\mathrm{FC}}(\mathbf{H}_c||\mathbf{T};\psi)\right), \tag{2}$$

with $\mathrm{LN}(\cdot)$ denoting layer-norm. The output is $\hat{\mathbf{H}}_c = \mathbf{H}_c + \bar{\mathbf{T}}_c$.

COATT Co-attention was introduced in diarization [29] to extend conventional end-to-end diarization (EEND-EDA [32]) to multiple channels. We modified the original co-attention module with two inputs and two outputs (single-channel summary and multi-channel representations) so that it can be used as the M-M module. Our version (see Figure 2) performs similar high-level operations as TAC (summarization of channels, concatenation with per-channel representations, augmentation of original channels). The difference is that both summary and per-channel representations are contextualized through cross-frame co-attention, providing the ability to tackle misalignment in time.

Given a multi-channel input $\{\mathbf{H}_c\}_{c=1}^{C}$, we obtain channels summary embeddings by $\mathbf{S} = \mathrm{LN}\left(\left[\frac{1}{C}\sum_c \mathbf{H}_c\right]\mathbf{W}_S\right)$, where $\mathbf{W}_S \in \mathbb{R}^{D\times d}$ reduces dimensionality to d. Per-channel representations are given as $\mathbf{M}_c = \mathrm{LN}\left(\mathbf{H}_c\mathbf{W}_M\right) \in \mathbb{R}^{T\times d'}$. In the next step, cross-frame co-attention is performed. To this end,

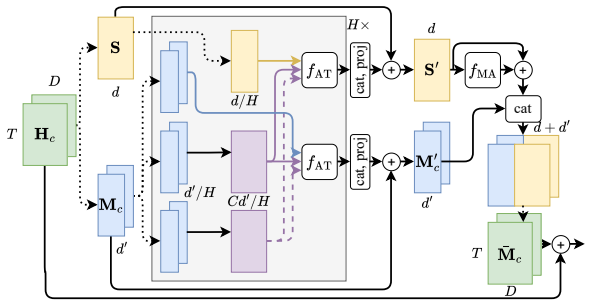

Figure 2: *Scheme of the modified cross-frame co-attention. Dotted lines transform the data, "*cat*" is an abbreviation for concatenation and "*proj*" for projection.*

per-channel representations are concatenated: $\mathbf{M} = \mathop{\Vert}_{c=1}^{C} \mathbf{M}_c \in \mathbb{R}^{T \times Cd'}$. The co-attention is then formulated as

$$\begin{aligned} \mathbf{M}'_c &= \mathrm{LN}\left(f_{\mathrm{MA}}(\mathbf{M}, \mathbf{M}_c; \xi) + \mathbf{M}_c\right), \\ \mathbf{S}' &= \mathrm{LN}\left(f_{\mathrm{MA}}(\mathbf{M}, \mathbf{S}; \xi') + \mathbf{S}\right). \end{aligned} \tag{3}$$

Both query $\mathcal{W}_Q^{(h)}$ and key $\mathcal{W}_K^{(h)}$ transformation matrices are shared by ξ and ξ' leading to the same attention matrix. Both $\mathcal{W}_Q^{(h)}$ and $\mathcal{W}_K^{(h)}$ are block-diagonal matrices composed of C matrices $\mathbf{W}_Q^{(h)} \in \mathbb{R}^{d' \times d'/H}$ and $\mathbf{W}_K^{(h)} \in \mathbb{R}^{d' \times d'/H}$, respectively. To link the description and Figure 2, we note the equivalence $\mathbf{M}\mathcal{W}_Q^{(h)} = \mathop{\Vert}_{c} \mathbf{M}_c \mathbf{W}_Q^{(h)}$. Next, cross-frame self-attention is applied to single-channel representations: $\bar{\mathbf{S}} = \mathrm{LN}\left(f_{\mathrm{MA}}(\mathbf{S}', \mathbf{S}'; \omega) + \mathbf{S}'\right)$. Finally, $\bar{\mathbf{S}}$ is concatenated with each channel of $\mathbf{M}'_c$ and linear projection takes the result to a D-dimensional space: $\bar{\mathbf{M}}_c = \left(\mathbf{M}'_c || \bar{\mathbf{S}}\right) \mathbf{W}_F$, $\mathbf{W}_F \in \mathbb{R}^{(d+d') \times D}$. The output is $\hat{\mathbf{H}}_c = \mathbf{H}_c + \bar{\mathbf{M}}_c$.

2.2. M-S Modules

The M-S modules fuse multi-channel representations into single-channel ones. Since mean pooling is sufficient in the architecture with similar high-level structure in [27], we opted for a weighted average for the final M-S module. The channel weights are jointly optimized with the network. We found that they converge to the same values, hence mean pooling suffices if agnosticity to the number of channels is of concern.

The downstream M-S modules are implemented either as a weighted average or the take-first approach. The latter simply uses the first (out of C) channel as its output.

3. Experiments

3.1. Setup

Pre-trained model Without loss of generality, we limit our experimentation to WavLM Base+ [16]. The reason is twofold. First, WavLM-based models provide strong results on the SUPERB benchmark. Second, it represents a good trade-off between size and performance in SV [13].

Training strategy We take a two-stage training approach. In the first stage, the MHFA downstream model with 64 heads is appended to a single-channel pre-trained WavLM Base+. We follow the joint optimization scheme [13], where the SSL model and MHFA are updated with a learning rate of 2e-5 and 1e-3, respectively. The model is trained on 3 s segments for 15 epochs, and the learning rates are decayed by a factor of 0.95 after each epoch. We use the additive angular margin loss (AAM, $s = 30$, $m = 0.2$) as an optimization objective [33]. We argue that access to a comprehensive multi-channel corpus could render this stage unnecessary. However, we empirically found it helpful.

In the second stage, we introduce the M-M and M-S into the architecture and jointly train the model on speaker-labeled multi-channel data with the same objective for 12 epochs. Newly incorporated weights are updated with a learning rate of 1e-3, while others with 2e-5. It follows from Figure 1 that the M-M modules in the backbone increasingly corrupt information flowing through the network if initialized randomly. To prevent the deeper layers from receiving polluted representations, we initialize the M-M modules so that $\hat{\mathbf{H}}_c \approx \mathbf{H}_c$. In TAC, we achieve it by setting a multiplicative parameter built in layer-norm LN in (2) to 1e-2. In COATT, values of $\mathbf{W}_F$ are drawn from $\mathcal{U}(-\sqrt{\frac{10^{-4}}{d+d'}}, \sqrt{\frac{10^{-4}}{d+d'}})$.The final M-S and downstream M-S are initialized to weigh channels equally.

By experimentation on the dev. set, dimensions $\bar{d}$, d and d' were set to 960, 128, and 32, respectively. Number of heads H is 8.

Training data To adapt the SSL model and MHFA to speaker recognition in the first training stage, we use VoxCeleb2 dev [34] augmented on the fly with MUSAN noise and reverberation. This dataset also serves for the baseline ResNet34 (with squeeze-and-excitation [35]) embedding extractor training.

The speakers in the multi-channel dataset must be a subset of those in the single-channel one to avoid architectural changes between stages. Therefore, we base the simulated multi-channel corpus on a speaker-balanced subset of VoxCeleb2 dev with recordings exceeding 15 dB (1,012 h). Apart from speech, we also introduce one source of corruption per utterance. Noise recordings are drawn from the following datasets: FMA large, WHAM! noise, MUSAN, CHiME-3, and FSD50K[1]. We made sure to remove speech and noise recordings overlapping with evaluation data. Given pairs of speech and noise data, simulation of 4-channel ad-hoc microphone arrays is performed using the image source method. Reverberation time RT60 is uniformly drawn from [0.2, 1] s. The mixing ratio ranges from 3 to 20 dB. To reflect evaluation data properties, we incorporate simulated cardioid and omnidirectional polar patterns.

Evaluation data We use retransmitted read speech data of MultiSV for evaluation [26]. Specifically, we follow the MRE and MRE_hard protocols that define verification trials with 4-channel enrollment and test parts. Reverberation and noise are present in all enrollment and test segments except for the MRE enrollment (which is noise-free). We used the development subset to select hyperparameters, so results on it are optimistic.

Metrics Scores for trials are computed as the cosine similarity of embeddings. We use the equal error rate (EER) and minimum detection cost function (mDCF) to assess the SV performance. Following [26], we set the target trial probability to 0.01.

3.2. Towards a Multi-Channel Extension of the SSL Models

To put the embedding extractors into the context of SV, we first analyze the performance on VoxCeleb1-O. The ResNet34 extractor yields 1.08% EER and 0.086 mDCF. A first-stage-training single-channel extractor (*SC MHFA*) provides 0.96% EER and 0.129 mDCF. In row A of Table 1, we show results on the MultiSV data when a random channel per microphone array is selected and the embedding is extracted with SC MHFA. It can be seen as a baseline for systems building on top of it.

Next, we present intuitive ways to extend SSL models to multi-channel data and justify the benefits of information fusion and, thus, our design. Following [8], a straightforward approach is to extract per-channel embeddings with SC MHFA and average them (row F of Table 1). The downside is that the model was trained on VoxCeleb2 dev. Therefore, a domain shift between training and evaluation data exists. To provide the ability to adapt to multi-channel data, we devise another baseline (row G). The SSL model processes channels in parallel, and downstream M-S implemented by weighted average provides input to the MHFA (the only updated part). This baseline provides clear benefits, especially for the eval. sets of MRE and MRE_hard. In row H, we present the performance of a strong baseline — a model with the same architecture where we allow fine-tuning of the SSL model (using learning rates according to Section 3.1).

[1] https://github.com/BUTSpeechFIT/MultiSV/blob/main/training/metadata/MultiSV2_train.zip.

Table 1: *Experimental results on MultiSV. Results with parentheses are in the format* mean (std)*, where the moments were computed on four runs with different seeds. Downstream M-S is abbreviated as d. M-S. The final layer with the M-M module is layer 4.*

	Model	params. [M]	MRE dev. EER	MRE dev. mDCF	MRE eval. EER	MRE eval. mDCF	MRE_hard dev. EER	MRE_hard dev. mDCF	MRE_hard eval. EER	MRE_hard eval. mDCF
A	SC MHFA	96.68	1.77	0.187	5.12	0.406	2.11	0.198	7.87	0.567
B	FaSNet + SC MHFA	99.44	1.95	0.181	6.69	0.447	2.27	0.188	10.44	0.644
C	TasN-BF + SC MHFA	97.88	1.33	0.135	3.03	0.293	1.31	0.138	3.72	0.334
D	TasN-BF + ResNet34	19.66	1.00	0.090	2.88	0.274	1.19	0.095	3.62	0.332
E	+ joint fine-tuning (ft.)	19.66	1.04	0.110	2.44	0.203	1.05	0.115	2.80	0.249
F	SC MHFA embed. avg.	96.68	1.07	0.125	2.48	0.277	1.40	0.134	4.57	0.430
G	SC MHFA parallel fixed	96.68	1.44	0.119	2.16	0.228	1.55	0.125	3.15	0.342
H	SC MHFA parallel trained	96.68	0.90 (0.04)	0.099	1.82 (0.06)	0.184	1.11 (0.02)	0.108	2.30 (0.06)	0.283
I	METRO w/ TAC	112.37	0.77 (0.03)	0.096	1.52 (0.03)	0.187	1.01 (0.06)	0.108	1.85 (0.11)	0.272
J	+ wavg. d. M-S	112.37	0.84 (0.04)	0.087	1.57 (0.06)	0.178	1.09 (0.03)	0.102	1.87 (0.04)	0.263
K	METRO w/ COATT	98.44	0.80 (0.02)	0.095	1.65 (0.07)	0.181	1.02 (0.04)	0.107	1.85 (0.02)	0.269
L	+ wavg. d. M-S	98.44	0.76 (0.07)	0.084	1.50 (0.04)	0.162	0.94 (0.07)	0.095	1.79 (0.06)	0.240

Table 2: *Comparison with published results on MultiSV.*

Model	params. [M]	MRE eval. EER	MRE eval. mDCF	MRE_hard eval. EER	MRE_hard eval. mDCF
Mask predictor [26]	17.16	3.91	0.355	5.37	0.518
TasN-BF [26]	15.16	3.71	0.364	4.61	0.482
TasN-BF + ResNet34 ft.	19.66	2.44	0.203	2.80	0.249
Diff-Filter [5]	–	3.07	–	3.19	–
METRO (L in Table 1)	98.44	**1.50**	**0.162**	**1.79**	**0.240**

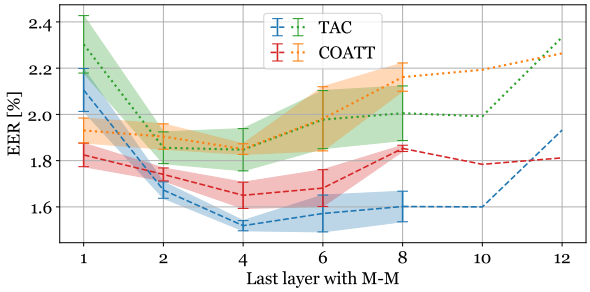

Figure 3: *Performance dependence on the last layer where the M-M module is inserted (with the take-first downstream M-S). Red and blue colors correspond to MRE eval., green and orange to MRE_hard eval. Shaded areas depict standard deviations obtained on four runs.*

Since multi-channel pre-processing is a common approach to multi-channel SV, we also compare the abovementioned baselines with cascaded models. Multi-channel pre-processing implemented by FaSNet [36] was trained on the training part of MultiSV. Despite an audible reduction of distractors in recordings, non-linear distortions introduced by the network result in a domain shift, which degrades the performance compared to SC MHFA (row B). A reimplementation of the approach with beamforming (with weights estimated using Conv-TasNet [12]) followed by our ResNet34 (row D) tends to outperform row F, while neither do adapt embedding extraction on multi-channel data. A version of D adapted on multi-channel data is in row E.

Rows I–L present results achieved with the proposed models. The M-M modules were incorporated according to Figure 1, where the last layer with M-M was selected based on Section 3.3. In rows I and K, the downstream M-S modules follow the take-first approach. The weighted average approach to the downstream M-S provides results in rows J and L. It is worth noting that the baseline H is more computationally expensive than any METRO instance as all channels are forwarded through the whole backbone. With the take-first approach to downstream M-S, TAC and COATT perform on par, while COATT requires much fewer parameters. Downstream M-S implemented by weighted average improves only mDCF in METRO with TAC. However, COATT consistently benefits from it, providing the best results overall.

3.3. Layer Dependence

We empirically found that keeping parallel processing throughout the whole architecture is not necessarily optimal. We hypothesize that keeping parallel processing at the beginning is beneficial because representations that are closer to the signal level provide complementary and spatial information. The fusion of abstract representations extracted farther within the model does not provide improvements.

In Figure 3, we present ablation results where we insert the M-M modules up to the layer denoted on the x-axis. While the decision on the last (4th) layer, where the M-M is inserted in our final models, was made on the dev. part, we note that trends are similar on dev. (which we do not show) and eval. subsets.

3.4. Comparison with Other Works

In Table 2, we compare METRO with published results on MultiSV. We note that all the studies use multi-channel preprocessing. The TasN-BF + ResNet34 ft. corresponds to row E in Table 1. While METRO provides the best results, model size may be a limiting factor in some use cases.

4. Conclusions

This paper presents METRO, a Multi-channel ExTension of pRe-trained mOdels for SV, which outperforms published results on MultiSV. It replicates the original blocks of the SSL models to perform parallel processing. They are followed by modules providing channel information exchange. Channels are eventually fused to one, and single-channel processing follows. We used MHFA to extract speaker embeddings given per-layer representations.

Even though we experimented with WavLM Base+, our approach is general. Therefore, we intend to explore other architectures and expect gains from large ones. In this paper, we presented METRO in the context of SV. We plan experimentation with other tasks (such as ASR) in future work.

5. Acknowledgements

The work was supported by Czech Ministry of Interior projects Nos. VJ01010108 "ROZKAZ". Ladislav's internship in INRIA Nancy was supported by the French Embassy in Prague as part of the Joseph Fourier prize. Computing on IT4I supercomputer was supported by the Ministry of Education, Youth and Sports of the Czech Republic through e-INFRA CZ (ID:90254).

6. References

[1] H. Taherian, Z.-Q. Wang, J. Chang, and D. Wang, "Robust Speaker Recognition Based on Single-Channel and Multi-Channel Speech Enhancement," *IEEE/ACM TASLP*, vol. 28, 2020.

[2] L. Mošner, P. Matějka, O. Novotný, and J. H. Černocký, "Dereverberation and Beamforming in Far-Field Speaker Recognition," in *IEEE ICASSP*, 2018.

[3] J.-Y. Yang and J.-H. Chang, "Joint Optimization of Neural Acoustic Beamforming and Dereverberation with x-Vectors for Robust Speaker Verification," in *Proc. Interspeech*, 2019.

[4] S. Dowerah, R. Serizel, D. Jouvet, M. Mohammadamini, and D. Matrouf, "Joint Optimization of Diffusion Probabilistic-Based Multichannel Speech Enhancement with Far-Field Speaker Verification," in *IEEE Spoken Language Technology Workshop*, 2023.

[5] S. Dowerah, A. Kulkarni, R. Serizel, and D. Jouvet, "Self-supervised Learning with Diffusion-based Multichannel Speech Enhancement for Speaker Verification under Noisy Conditions," in *Proc. Interspeech*, 2023.

[6] D. Cai, X. Qin, and M. Li, "Multi-Channel Training for End-to-End Speaker Recognition Under Reverberant and Noisy Environment," in *Proc. Interspeech*, 2019.

[7] X. Qin, D. Cai, and M. Li, "Robust Multi-Channel Far-Field Speaker Verification Under Different In-Domain Data Availability Scenarios," *IEEE/ACM TASLP*, vol. 31, 2023.

[8] D. Cai and M. Li, "Embedding Aggregation for Far-Field Speaker Verification with Distributed Microphone Arrays," in *IEEE Spoken Language Technology Workshop*, 2021.

[9] C. Liang, J. Chen, S. Guan, and X.-L. Zhang, "Attention-based Multichannel Speaker Verification with Ad-hoc Microphone Arrays," in *Proc. APSIPA Annual Summit and Conference*, 2021.

[10] C. Liang, Y. Chen, J. Yao, and X.-L. Zhang, "Multi-Channel Far-Field Speaker Verification with Large-Scale Ad-hoc Microphone Arrays," in *Proc. Interspeech*, 2022.

[11] S. Shon, H. Tang, and J. Glass, "VoiceID Loss: Speech Enhancement for Speaker Verification," in *Proc. Interspeech*, 2019.

[12] L. Mošner, O. Plchot, L. Burget, and J. H. Černocký, "Multi-Channel Speaker Verification with Conv-Tasnet Based Beamformer," in *IEEE ICASSP*, 2022.

[13] J. Peng, O. Plchot, T. Stafylakis, L. Mošner, L. Burget, and J. Černocký, "An Attention-Based Backend Allowing Efficient Fine-Tuning of Transformer Models for Speaker Verification," in *IEEE Spoken Language Technology Workshop*, 2023.

[14] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020.

[15] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units," *IEEE/ACM TASLP*, vol. 29, 2021.

[16] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda *et al.*, "WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing," *IEEE J-STSP*, vol. 16, no. 6, 2022.

[17] M. Fazel-Zarandi and W.-N. Hsu, "Cocktail Hubert: Generalized Self-Supervised Pre-Training for Mixture and Single-Source Speech," in *IEEE ICASSP*, 2023.

[18] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi *et al.*, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. Interspeech*, 2021.

[19] Z. Chen, S. Chen, Y. Wu, Y. Qian, C. Wang, S. Liu, Y. Qian, and M. Zeng, "Large-Scale Self-Supervised Speech Representation Learning for Automatic Speaker Verification," in *IEEE ICASSP*, 2022.

[20] S. Novoselov, G. Lavrentyeva, A. Avdeeva, V. Volokhov, N. Khmelev, A. Akulov, and P. Leonteva, "On the Robustness of wav2vec 2.0 Based Speaker Recognition Systems," in *Proc. Interspeech*, 2023.

[21] S. Cornell, M. S. Wiesner, S. Watanabe, D. Raj, X. Chang, P. Garcia, Y. Masuyam, Z.-Q. Wang *et al.*, "The CHiME-7 DASR Challenge: Distant Meeting Transcription with Multiple Devices in Diverse Scenarios," in *Proc. 7th International Workshop on Speech Processing in Everyday Environments (CHiME 2023)*, 2023.

[22] Y. Masuyama, X. Chang, S. Cornell, S. Watanabe, and N. Ono, "End-to-End Integration of Speech Recognition, Dereverberation, Beamforming, and Self-Supervised Learning Representation," in *IEEE Spoken Language Technology Workshop*, 2023.

[23] Y. Masuyama, X. Chang, W. Zhang, S. Cornell, Z.-Q. Wang, N. Ono, Y. Qian, and S. Watanabe, "Exploring the Integration of Speech Separation and Recognition with Self-Supervised Learning Representation," in *IEEE WASPAA*, 2023.

[24] A. Dimitriadis, S. Pan, V. Sethu, and B. Ahmed, "Spatial HuBERT: Self-supervised Spatial Speech Representation Learning for a Single Talker from Multi-channel Audio," *arXiv preprint arXiv:2310.10922*, 2023.

[25] Q. Zhu, J. Zhang, Y. Gu, Y. Hu, and L. Dai, "Multichannel AV-wav2vec2: A Framework for Learning Multichannel Multi-Modal Speech Representation," in *Proc. AAAI*, 2024.

[26] L. Mošner, O. Plchot, L. Burget, and J. H. Černocký, "MultiSV: Dataset for Far-Field Multi-Channel Speaker Verification," in *IEEE ICASSP*, 2022.

[27] D. Wang, Z. Chen, and T. Yoshioka, "Neural Speech Separation Using Spatially Distributed Microphones," in *Proc. Interspeech*, 2020.

[28] D. Wang, T. Yoshioka, Z. Chen, X. Wang, T. Zhou, and Z. Meng, "Continuous Speech Separation with Ad Hoc Microphone Arrays," in *EUSIPCO*, 2021.

[29] S. Horiguchi, Y. Takashima, P. García, S. Watanabe, and Y. Kawaguchi, "Multi-Channel End-To-End Neural Diarization with Distributed Microphones," in *IEEE ICASSP*, 2022.

[30] Y. Luo, Z. Chen, N. Mesgarani, and T. Yoshioka, "End-to-end Microphone Permutation and Number Invariant Multi-channel Speech Separation," in *IEEE ICASSP*, 2020.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[32] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and P. García, "Encoder-Decoder Based Attractors for End-to-End Neural Diarization," *IEEE/ACM TASLP*, vol. 30, 2022.

[33] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *IEEE/CVF CVPR*, 2019.

[34] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," in *Proc. Interspeech*, 2018.

[35] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *IEEE/CVF CVPR*, June 2018.

[36] Y. Luo, C. Han, N. Mesgarani, E. Ceolini, and S.-C. Liu, "FaSNet: Low-Latency Adaptive Beamforming for Multi-Microphone Audio Processing," in *IEEE Automatic Speech Recognition and Understanding Workshop*, 2019.