This is the Pre-Published Version.

The following publication L. -j. Zhao and M. -W. Mak, "Channel Interdependence Enhanced Speaker Embeddings for Far-Field Speaker Verification," 2021 12th International Symposium on Chinese Spoken Language Processing (ISCSLP), Hong Kong, 2021 is available at https://doi.org/10.1109/ISCSLP49672.2021.9362108.

Channel Interdependence Enhanced Speaker Embeddings for Far-Field Speaker Verification

Ling-jun Zhao and Man-Wai Mak

Dept. of Electronic and Information Engineering,
The Hong Kong Polytechnic University, Hong Kong SAR, China
{lingjzhao, enmwmak}@polyu.edu.hk

Abstract

Recognizing speakers from a distance using far-field microphones is difficult because of the environmental noise and reverberation distortion. In this work, we tackle these problems by strengthening the frame-level processing and feature aggregation of x-vector networks. Specifically, we restructure the dilated convolutional layers into Res2Net blocks to generate multi-scale frame-level features. To exploit the relationship between the channels, we introduce squeeze-and-excitation (SE) units to rescale the channels' activations and investigate the best places to put these SE units in the Res2Net blocks. Based on the hypothesis that layers at different depth contain speaker information at different granularity levels, multi-block feature aggregation is introduced to propagate and aggregate the features at various depths. To optimally weight the channels and frames during feature aggregation, we propose a channel-dependent attention mechanism. Combining all of these enhancements leads to a network architecture called channel-interdependence enhanced Res2Net (CE-Res2Net). Results show that the proposed network achieves a relative improvement of about 16% in EER and 17% in minDCF on the VOiCES 2019 Challenge's evaluation set.

Index Terms: Far-field speaker verification; speaker embedding; Res2Net; Squeeze-and-excitation; channel-dependent attention

1. Introduction

Speaker verification (SV) using far-field microphones has long been a challenging problem because of the environmental noise and uncontrollable reverberation [1]. Due to the difference in microphone characteristics, there exists a domain mismatch between near-field microphone speech and far-field microphone speech. The mismatch could cause SV systems that are trained on near-field microphone speech to perform poorly on far-field microphone speech. Therefore, it is essential to develop a resilient speaker-embedding method to overcome this kind of domain mismatch.

Recently, the use of DNNs to create a speaker embedding space that captures most of the speaker characteristics has become an active research area [2, 3]. A promising approach to creating speaker embedding is to train a time-delay neural network (TDNN) to capture the frame-level features, followed by a statistics pooling process to aggregate the frame-level features to obtain the utterance-level features (embeddings). Speaker verification is then accomplished by comparing the embeddings of two utterances to determine whether they are spoken by the same speaker or not. The approach is known as x-vector [4], which consistently achieves state-of-the-art performance in speaker recognition evaluations [5, 6, 7].

It has been shown that adding residual connections between the frame-level layers enhances speaker embeddings [8, 9]. Leveraged on the idea of x-vector, a recent trend in speaker embedding is to enhance the capability of the frame-level processing by replacing the TDNN with ResNet [10, 11] and DenseNet [12, 13]. In [14], it was found that enhancing the multi-scale temporal features at a granular level by the Res2Net blocks [15] and recalibrating the channel-wise feature responses by the squeeze-and-excitation (SE) units [16] can improve performance. However, there is a lack of study on the impact of the variants of SE and integration strategy, i.e., the best placement of the SE units in the Res2Net blocks. Considering that multi-scale representations can be obtained from a Res2Net block, we argue that the placement of SE units has an impact on the channel interdependence.

This work aims to fill this gap by performing an ablation study to access the influence of the SE unit's placement on speaker embeddings. Also, an SE variant is proposed. This new SE unit incorporates a statistics pooling mechanism into the squeeze operation, which is expected to generate better channel-wise statistics. Motivated by the merits of SE-based Res2Net blocks (SE-Res2blocks) and multi-block feature aggregation, we propose the channel-interdependence enhanced Res2Net (CE-Res2Net) for tackling the far-field SV problem. In addition to the enhancement in network architecture, we also investigate a channel-dependent attentive pooling mechanism, which weights individual frames based on the channel attention scores. We compare different network designs and demonstrate the effectiveness of our *CE-Res2Net* system on the VOiCES Challenge 2019 dataset [17], or Voices19 for short.

2. Channel-Interdependence Enhanced Embedding

This section details the main components of the proposed CE-Res2Net for enhancing the TDNN architecture and the attentive pooling layer. The configuration of the CE-Res2Net is shown in Fig. 1.

2.1. SE-based Res2block

To investigate the relationship between the channels, we introduce a 1D-SE unit, which recalibrates the channel activations based on channel dependence and the global information about the channels. The squeeze operation in the SE unit uses global

This work was in part supported by the Huawei Technologies Co., Ltd, Project No. YBN2019095008 and National Natural Science Foundation of China (NSFC), Grant No. 61971371.

©2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

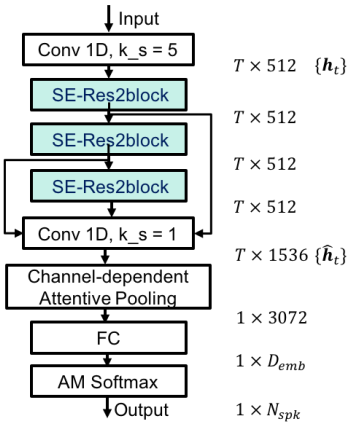

Figure 1: *Illustration of the CE-Res2Net topology. k_s denotes the kernel size. $\boldsymbol{h}_t$ $(t = 1, \ldots, T)$ denotes the frame-level features. $\hat{\boldsymbol{h}}_t$ denotes the last frame-level convolutional layer's output. T is the utterance length. D_{emb} denotes the dimension of speaker embeddings. N_{spk} is the number of training speakers.*

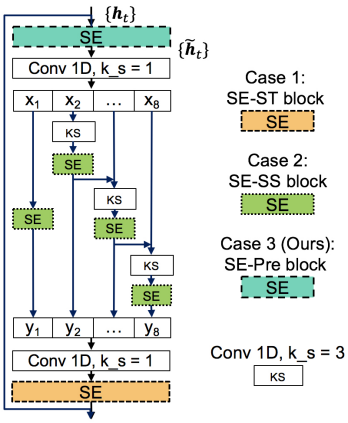

Figure 2: *Variants of an SE-based Res2block. "SE" denotes the standard SE unit or the statistics pooling based SE (SPSE) unit. $\tilde{\boldsymbol{h}}_t$ denotes the rescaled frame-level features by "SE". In this example, the channels after the "Conv 1D" are split into 8 channel groups.*

average pooling to obtain the channel-wise statistics:

$$\boldsymbol{u} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{h}_t, \quad \boldsymbol{u} \in \mathbb{R}^C, \tag{1}$$

where $\boldsymbol{h}_t$ $(t = 1, \ldots, T)$ denote the frame-level features at the last convolutional layer and C is the number of channels. After that, the excitation operator uses the aggregated information $\boldsymbol{u}$ to compute a set of channel weights:

$$\boldsymbol{s} = S(\boldsymbol{W}_2 f(\boldsymbol{W}_1 \boldsymbol{u})), \quad \boldsymbol{s} \in \mathbb{R}^C, \tag{2}$$

where S is a sigmoid function, f is a leaky ReLU function, $\boldsymbol{W}_1 \in \mathbb{R}^{\frac{C}{r}\times C}$, $\boldsymbol{W}_2 \in \mathbb{R}^{C\times\frac{C}{r}}$, and $r \ll C$ is a constant for reducing the dimension of $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$. In this work, r was set to 2. For simplicity, all the bias terms are omitted. The output $\tilde{\boldsymbol{h}}_t$ of the unit is obtained by rescaling the features in each channel with the weights in $\boldsymbol{s}$:

$$\tilde{\boldsymbol{h}}_t = \boldsymbol{s} \odot \boldsymbol{h}_t, \tag{3}$$

where $\odot$ represents the Hadamard product.

Based on the standard SE in [16], we propose a variant of SE named statistics pooling-based SE (SPSE), which incorporates a statistics pooling [18] mechanism into the squeeze operator. In details, the statistics pooling calculates a mean vector $\boldsymbol{u}$ as in Eq. 1 and a standard deviation vector $\boldsymbol{\sigma}$ over $\boldsymbol{h}_t$ as follows:

$$\boldsymbol{\sigma} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{h}_t \odot \boldsymbol{h}_t - \boldsymbol{u} \odot \boldsymbol{u}}. \tag{4}$$

In addition to the mean vector $\boldsymbol{u}$, we consider that the standard deviation $\boldsymbol{\sigma}$ is also important for describing channel-wise statistics because it contains information on temporal variability. The $\boldsymbol{u}$ and $\boldsymbol{\sigma}$ are concatenated and passed to the excitation operation in Eq. 2, i.e., replacing $\boldsymbol{u}$ by $\left[\boldsymbol{u}^\mathsf{T}, \boldsymbol{\sigma}^\mathsf{T}\right]^\mathsf{T}$. To ensure that the SE unit and the SPSE unit have the same number of parameters, the value r for $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ in Eq. 2 was set to 4.

Res2block [15], a recently proposed architecture, uses hierarchical residual-like connections within each residual block to extract multi-scale features. Due to the evidence that integrating the SE units into the standard ResNet could improve performance [16], we integrate the SE unit into the 1D Res2block. The benefit of the integration will be demonstrated in Section 4.

To further explore the impact of the placement of the SE unit within the Res2Net, three kinds of SE-Res2block are proposed, as shown in Fig. 2. For simplicity, we use "SE" in this figure to represent the standard SE unit or the SPSE unit. Similar to [15, 16], the most straightforward way to integrate the SE unit is putting it after all the convolutional operations (conv-ops) in the Res2block, which is called standard SE (SE-ST) as shown in "Case 1" of Fig. 2. In this block, the SE unit recalibrates the channel activations based on the multi-scale features obtained from the conv-ops in the Res2block. In scale-specific SE (SE-SS), "Case 2" of Fig. 2, an SE unit is placed after each scale-specific branch. In this case, each SE unit models the interdependence within the channel group based on the global information of the scale-specific branch. In SE-Pre, "Case 3" of Fig. 2, an SE unit is placed before the conv-ops. It first rescales the channel activations and then passes them to the conv-ops to learn multi-scale features. For comparison, the number of parameters in the three types of SE-Res2blocks is the same. The performance of the SPSE unit and the impact of the placement of the SE unit are revealed in Section 4.

2.2. Channel-dependent Self-attention

Self-attentive pooling [19, 20] assigns a weight or a scalar score e_t for each frame-level feature vector through a trainable layer. However, this kind of mechanism assumes that all channels are of equal importance. To explore the importance of individual channel, we introduce channel-dependent attention, which calculates a scalar score $e_{t,c}$ $(c = 1, \ldots, C)$ for each frame-level vector $\hat{\boldsymbol{h}}_t$ at the last convolutional layer's output in Fig. 1:

$$e_{t,c} = \boldsymbol{v}_c^\mathsf{T} f\left(\boldsymbol{W}\hat{\boldsymbol{h}}_t\right), \tag{5}$$

where $\boldsymbol{W} \in \mathbb{R}^{\frac{C}{d}\times C}$ and $\boldsymbol{v}_c \in \mathbb{R}^{\frac{C}{d}\times 1}$ are parameters to calculate the attention weights. $d \ll C$ is set to reduce the number of parameters and to avoid overfitting. In this work, $d = 4$. All the bias terms are omitted for simplicity. The $e_{t,c}$ is then

normalized channel-wise across time by a softmax layer:

$$w_{t,c} = \frac{\exp\left(e_{t,c}\right)}{\sum_{\tau=1}^{T} \exp\left(e_{\tau,c}\right)}. \quad (6)$$

Given a set of channel-dependent weights $w_{t,c}$, we can obtain the weighted average of channel c:

$$\hat{\mu}_c = \sum_{t=1}^{T} w_{t,c} \hat{h}_{t,c}. \quad (7)$$

The weighted mean vector is $\hat{\boldsymbol{\mu}} = [\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_C]^{\mathsf{T}}$. Similar to the self-attentive pooling, the weighted standard deviation vector $\hat{\boldsymbol{\sigma}}$ can be computed:

$$\hat{\sigma}_c = \sqrt{\frac{1}{T} \sum_{t=1}^{T} w_{t,c} \hat{h}_{t,c}^2 - \hat{\mu}_c^2}, \quad c = 1, \ldots, C. \quad (8)$$

By concatenating the weighted mean vector $\hat{\boldsymbol{\mu}}$ and the weighted standard deviation vector $\hat{\boldsymbol{\sigma}}$, the output of the channel-dependent attention is obtained.

2.3. Multi-block Feature Aggregation

In the original x-vector framework, only the last frame-level layer is connected to the statistics pooling layer. It is conjectured that the speaker-specific information is more prominence at the top TDNN layer than those near the input. However, there is no in-depth investigation on the amount of speaker information in different layers. It is suspected that the shallow features (features near the input) could also contribute to speaker embeddings. To investigate the impact of shallow features, we introduce a multi-block aggregation (MA) method that concatenates the output feature maps of all the SE-Res2blocks, as shown in Fig. 1. A dense layer is followed to aggregate the concatenated information and then pass the features to the statistics pooling layer. In this way, features from all stages of the blocks are propagated into the dense layer, which will contribute directly to the attentive pooling layer.

3. Experimental Setup

3.1. Data Preparation

We used the VOiCES Challenge 2019 (Voices19) for performance evaluation. The dataset is based on the Voices Obscured in Complex Environmental Settings (VOiCES) corpus [21]. The speech samples were recorded by 12 microphones placed at different positions of four rooms with distinct reverberation conditions and noise distractions. Distractor noises – including babble, music, and television sound – were played in the background. Voices19 includes a development (Dev) set and an evaluation (Eval) set; each of them comprises an enrollment subset and a test subset.

The training data consist of Speakers in the Wild (SITW) [22], VoxCeleb1 [23], and VoxCeleb2 [24]. We followed the data augmentation procedure in Kaldi to create the augmented data, using the noise, babble, and music files in MUSAN [25] as the noise sources and the RIR dataset [26] as the room impulse responses for producing the reverberant speech. A random subset with one million augmented utterances was combined with the original utterances to create an augmented training set with around 2.2 million utterances. Utterances with number of frames less than 400 and speakers with fewer than 8 utterances were excluded from this set.

3.2. Acoustic Features and Speaker Embedding

30-dimensional mel-frequency cepstral coefficients (MFCCs) were extracted from 16-kHz audio signals using a 25-ms window with 10-ms frame shift. The mel-scale filter bank covers frequency range 20–7600Hz. After that, cepstral mean normalization (CMN) over a 3-second sliding window was applied. The Kaldi's voice activity detector (VAD) was used to remove non-speech. For Voices19 speech files, a dereverberation method called weighted prediction error (WPE) [27] was applied to improve signal quality. No dereverberation was applied to VoxCeleb files.

The configuration of our system is illustrated in Fig. 1. The dilation of the Conv1D layers is 2, 3, and 4 for the three SE-Res2blocks, respectively. The dimension of speaker embedding, D_{emb}, is 192 and the number of training speakers, N_{spk}, is 7602. Each of the convolutional layers in all systems is connected to a leaky ReLU layer, followed by batch normalization. Speaker embeddings were extracted from the affine layer's output after the statistics pooling layer.

All models were trained using the Adam optimizer with an initial learning rate of 0.001 and linearly reduced to 0.0001 for 3 epochs. Training samples were randomly shuffled before each epoch. The mini-batch size is 64. Simple softmax was first used to investigate the importance of individual components in the proposed CE-Rese2Net. AM-softmax [28] with a scale factor of 30 and a penalty margin of 0.2 was later used as the output activation function for all models. To avoid overfitting, L2-regularization with weight equals to 2×10^{-5} was applied.

3.3. Scoring and Evaluation

The PLDA model training and PLDA scoring were realized by Kaldi. Before that, length normalization, centering, and LDA were applied to reduce feature dimension and enhance speaker discrimination. No PLDA adaptation was applied. Symmetric score normalization (S-norm) [29] was performed to normalize the scores of the Eval-test set, where the cohort was selected from the embeddings of the Dev-test set. All systems were evaluated on the Dev-test set and Eval-test set. The performance metrics consist of equal error rate (EER) and minimum detection cost function (minDCF), where P_{target} is 0.01.

3.4. Advanced Baseline Systems

Extended TDNN (E-TDNN) and ResNet-based systems [30, 31] were used as the two strong baselines for comparisons. An enhanced attentive pooling mechanism, named multi-head self-attention (MHA) [32], was incorporated into the two baselines to produce better embeddings.

One of our baseline systems is the MHA-based E-TDNN x-vector system (MHA-ETDNN). The architecture of E-TDNN is introduced in [30] and described in great detail there. This network includes four 1-dimensional (1D) dilated convolutional layers with interleaving dense layers. After two dense layers, a statistics pooling layer is added to calculate the mean and standard deviation over the last frame-level representations. Different from the original E-TDNN, we used MHA in the statistics pooling layer, which has shown to improve performance [32]. Before the softmax output layer, two dense layers were added and the speaker embeddings were extracted from the linear affine outputs in the first dense layer. We replaced the softmax in [30] by the additive margin softmax (AM-softmax) [28].

Another baseline system, MHA-based ResNet (MHA-ResNet), is a variant of the standard TDNN x-vector system [4].

Table 1: *Importance of different components in the proposed CE-Res2Net. Boldface values are the best results. SE-Res2blocks employed here are all SE-ST blocks described in Section 2.1 and shown in Fig. 2. Network outputs were normalized by the softmax function. MA: multi-block aggregation. MHA: multi-head attention. SE: squeeze-and-excitation. CE-Res2Net: Fig. 1 with channel-dependent attention. CE-Res2Net w/o SE: only removing SE units from CE-Res2Net. CE-Res2Net w/o MA: only removing MA from CE-Res2Net. CE-Res2Net w/ MHA: only replacing channel-dependent attention in CE-Res2Net by MHA.*

Systems	**Development set**		**Evaluation set**	
	EER(%)	**MinDCF**	**EER(%)**	**MinDCF**
Kaldi x-vector	3.61	0.3785	7.78	0.5624
CE-ResNet	2.57	0.3072	6.54	0.5297
CE-Res2Net	**2.31**	0.2685	6.15	**0.4918**
CE-Res2Net w/o MA	2.50	0.2829	6.23	0.5206
CE-Res2Net w/o SE	2.52	0.2834	**6.05**	0.4943
CE-Res2Net w/ MHA	2.34	**0.2670**	6.52	0.5265

Table 2: *Performance of the proposed CE-Res2Net with different variants of SE-Res2blocks on the Dev and Eval set of Voices19. All the variants of SE-Res2blocks are described in Section 2.1 and shown in Fig. 2. Network outputs were normalized by the additive margin softmax (AM-softmax).*

System	**Development set**		**Evaluation set**	
	EER(%)	**MinDCF**	**EER(%)**	**MinDCF**
SE-ST	2.18	0.2281	5.96	0.4296
SE-SS	2.12	0.2792	6.06	0.4763
SE-Pre	2.02	**0.2108**	**5.72**	**0.4225**
SPSE-Pre	**1.93**	0.2165	5.77	0.4232

The second to the forth TDNN layers were replaced by three residual blocks [10]. Each residual block consists of two 1D convolutions with kernel size equal to 1 and one 1D dilated convolution with kernel size equal to 3. The dilation of the three residual blocks was set to 2, 3 and 4, respectively. Same as the MHA-TDNN, we used MHA as the statistics pooling module to produce the fixed-length representation from variable-length speech segments. After that, one dense layer together with an AM-softmax layer were added. Speaker embeddings were extracted from the linear activations in the dense layer.

4. Results

The impacts of individual components in the proposed CE-Res2Net are given in Table 1. The results demonstrate that the CE-Res2Net significantly outperforms the Kaldi x-vector baseline [4]. To demonstrate the importance of aggregating all the SE-Res2blocks, we removed the multi-block aggregation (MA) and used the output of the last frame-level dense layer for the attentive pooling operation. Compared with the results of CE-Res2Net without MA, aggregation could obtain improvement of about 8% in EER and 5% in minDCF on the Dev set and 6% in minDCF on the Eval set. To explore the impact of channel-dependent attention, we used multi-head attention (MHA) to replace it. The results (CE-Res2Net w/ MHA) demonstrate the superiority of channel-dependent attention, which is about 6% better in EER and 7% better in minDCF on the Eval set. Compared with CE-Res2Net without any SE units, incorporating the SE units could reduce the EER by 8% and reduce the minDCF by 5% on the Dev set. The results confirm the advantage of rescaling the channel activations by the SE units. In the CE-ResNet in Table 1, the Res2blocks are replaced by the standard ResNet blocks, which degrades the performance by about 9% in EER and 12% in minDCF on the Dev set and degrades the performance by about 6% in EER and 7% in minDCF on the Eval set. Results in Table 1 confirm that the CE-Res2Net can benefit from the SE units, channel-dependent attention, Res2blocks, and MA.

Table 3: *Performance of CE-Res2Net and baseline systems (Section 3.4) on the Dev and Eval set of Voices19. SE-Res2blocks employed in the CE-Res2Net are SE-Pre blocks described in Section 2.1. Network outputs were normalized by the additive margin softmax (AM-softmax).*

Systems	**Params**	**Development set**		**Evaluation set**	
		EER (%)	**MinDCF**	**EER (%)**	**MinDCF**
MHA-ETDNN	11.4M	2.67	0.3028	6.81	0.5084
MHA-ResNet	10.2M	2.10	0.2628	6.11	0.4565
CE-Res2Net	9.5M	**2.02**	**0.2108**	**5.72**	**0.4225**

The performance of the standard SE (SE-ST) block and its variants introduced in Section 2.1 are reported in Table 2. The SE-SS block, where each split convolutional operation is followed by an SE unit, leads to performance drop in both Dev and Eval sets. The SE-Pre block, in which the SE unit is placed before the conv-ops, results in relative reduction of 5% in EER and 8% in minDCF on the Dev set and 4% reduction in EER and 2% reduction in minDCF on the Eval set. SPSE-Pre, in which statistics pooling is introduced in the SE-Pre block, achieves performance comparable to SE-Pre. These results clearly confirm that the placements of SE units have influence on performance. In particular, SE-Pre and SPSE-Pre, i.e., placing the SE unit before conv-ops in a Res2Net block, lead to better performance.

The results of the proposed CE-Res2Net and the two baselines described in Section 3.4 are summarized in Table 3. SE-Pre blocks are employed in the CE-Res2Net in this table because of their good performance. It is obvious that the proposed CE-Res2Net significantly improves the performance. Also, it has the least number of parameters compared with the two baselines. In detail, compared with MHA-ResNet, the CE-Res2Net obtains a relatively improvement of about 4% in EER and 20% in minDCF on the Dev set and 6% in EER and 7% in minDCF on the Eval set; compared with MHA-ETDNN, the CE-Res2Net obtains a relatively improvement of about 24% in EER and 30% in minDCF on the Dev set and 16% in EER and 17% in minDCF on the Eval set.

5. Conclusions

In this paper, we presented a novel speaker embedding extractor, CE-Res2Net, for speaker verification. Unlike the original TDNN in x-vector networks, we focused on the interdependence between channels by introducing SE units and channel-dependent attention into the network. We investigated different placements of the SE block and found that it is better to put it before the Res2Net block. With the Res2blocks, multi-block feature aggregation, and channel-dependent attentive pooling, the CE-Res2Net can significantly improves the performance of far-field speaker verification.

6. References

[1] Q. Jin, T. Schultz, and A. Waibel, "Far-field speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2023–2032, 2007.

[2] P. S. Nidadavolu, S. Kataria, J. Villalba, P. Garcia-Perera, and N. Dehak, "Unsupervised feature enhancement for speaker verification," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7599–7603.

[3] D. Cai, W. Cai, and M. Li, "Within-sample variability-invariant loss for robust speaker recognition under noisy environments," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6469–6473.

[4] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[5] L. Burget, O. Novotny, and O. Glembek, "Analysis of BUT submission in far-field scenarios of VOiCES 2019 challenge," in *Proc. Interspeech*, 2019.

[6] J. Villalba, N. Chen, D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, J. Borgstrom, F. Richardson, S. Shon, F. Grondin *et al.*, "State-of-the-art speaker recognition for telephone and video speech: The JHU-MIT submission for NIST SRE18." in *Proc. Interspeech*, 2019, pp. 1488–1492.

[7] K. A. Lee, H. Yamamoto, K. Okabe, Q. Wang, L. Guo, T. Koshinaka, J. Zhang, and K. Shinoda, "The NEC-TT 2018 speaker verification system." in *Proc. Interspeech*, 2019, pp. 4355–4359.

[8] S. Novoselov, A. Shulipa, I. Kremnev, A. Kozlov, and V. Shchemelinin, "On deep speaker embeddings for text-independent speaker recognition," *arXiv preprint arXiv:1804.10080*, 2018.

[9] A. Gusev, V. Volokhov, T. Andzhukaev, S. Novoselov, G. Lavrentyeva, M. Volkova, A. Gazizullina, A. Shulipa, A. Gorlanov, A. Avdeeva *et al.*, "Deep speaker embeddings for far-field speaker recognition on short utterances," *arXiv preprint arXiv:2002.06033*, 2020.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[11] N. Li, D. Tuo, D. Su, Z. Li, and D. Yu, "Deep discriminative embeddings for duration robust speaker verification." in *Proc. Interspeech*, 2018, pp. 2262–2266.

[12] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[13] W. Lin, M. W. Mak, and L. Yi, "Learning mixture representation for deep speaker embedding using attention," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 210–214.

[14] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[15] S. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. H. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence, doi:10.1109/TPAMI.2019.2938758*, 30 Aug. 2019.

[16] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[17] M. K. Nandwana, J. Van Hout, M. McLaren, C. Richey, A. Lawson, and M. A. Barrios, "The VOiCES from a distance challenge 2019 evaluation plan," *arXiv preprint arXiv:1902.10828*, 2019.

[18] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification." in *Proc. Interspeech*, 2017, pp. 999–1003.

[19] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," *arXiv preprint arXiv:1803.10963*, 2018.

[20] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification." in *Proc. Interspeech*, 2018, pp. 3573–3577.

[21] C. Richey, M. A. Barrios, Z. Armstrong, C. Bartels, H. Franco, M. Graciarena, A. Lawson, M. K. Nandwana, A. Stauffer, J. van Hout *et al.*, "Voices obscured in complex environmental settings (VOiCES) corpus," *arXiv preprint arXiv:1804.05053*, 2018.

[22] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The speakers in the wild (SITW) speaker recognition database." in *Proc. Interspeech*, 2016, pp. 818–822.

[23] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[24] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[25] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[26] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5220–5224.

[27] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717–1731, 2010.

[28] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[29] P. Matejka, O. Novotnỳ, O. Plchot, L. Burget, M. D. Sánchez, and J. Cernockỳ, "Analysis of score normalization in multilingual speaker recognition." in *Proc. Interspeech*, 2017, pp. 1567–1571.

[30] D. Garcia-Romero, A. McCree, D. Snyder, and G. Sell, "JHU-HLTCOE system for the Voxsrc speaker recognition challenge," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7559–7563.

[31] D. Cai, X. Qin, W. Cai, and M. Li, "The DKU system for the speaker recognition task of the 2019 VOiCES from a distance challenge," *arXiv preprint arXiv:1907.02194*, 2019.

[32] P. Safari and J. Hernando, "Self multi-head attention for speaker recognition," *arXiv preprint arXiv:1906.09890*, 2019.