Generalization Ability Improvement of Speaker Representation and Anti-Interference for Speaker Verification

Qian-Bei Hong, Chung-Hsien Wu, *Senior Member, IEEE*, and Hsin-Min Wang, *Senior Member, IEEE*

***Abstract*—The ability to generalize to mismatches between training and testing conditions and resist interference from other speakers is crucial for the performance of speaker verification. In this paper, we propose two novel approaches to improve the generalization ability to deal with the mismatched recorded scenarios and languages in test conditions and to reduce the influence of interference from other speakers on the similarity measurement of two speaker embeddings. First, parent embedding learning (PEL) is used for model training, which exploits the generalization ability of the shared structure to improve the representation of speaker embeddings. Second, partial adaptive score normalization (PAS-Norm) is used to reduce the influence of interference from other speakers on embedding-based similarity measures. In the experiments, the speaker embedding models are trained using the VoxCeleb2 dataset, and the performance is evaluated on four other datasets under different conditions, including VoxCeleb1, Librispeech, SITW, and CN-Celeb datasets. In the experiments on VoxCeleb1, evaluation results considering a large number of verification speakers and identity restrictions show that the proposed PEL-based system reduces the EER by 6.0% and 4.9% in these two cases, respectively, compared to the state-of-the-art (SOTA) system. Furthermore, in the experiments evaluating speaker verification in mismatch conditions on SITW and CN-Celeb, the proposed PEL-based system also outperforms the SOTA system. In the language mismatched conditions, the EER is reduced by 8.3%. For the evaluation of the influence of interference from other speakers, the EER is significantly reduced by 24.4% when PAS-Norm is used instead of the baseline AS-Norm score normalization method.**

***Index Terms*—Speaker verification, parent embedding learning, partial adaptive score normalization**

I. Introduction

SPEAKER recognition has been an important task for artificial intelligence applications [1]-[3] for years. Traditional methods for speaker recognition typically use the Gaussian mixture model-universal background model (GMM-UBM) and i-vector to represent the short-term spectral characteristics of a speaker [4]-[6]. Villalba et al. [7] presented a framework based on the variational autoencoder paradigm to deal with the latent variables between i-vectors. Kinnunen et al. [8] adopted i-vector as the basic unit for voice conversion. In recent years, deep neural networks (DNNs) have been widely used to generate features for speaker recognition [9]-[14]. In these studies, DNN was applied to directly capture speaker characteristics and produce speaker embeddings as speaker representations. Many speaker verification (SV) systems have recently been developed that are based on x-vector features [15]-[16], with an architecture that includes frame-level and segment-level feature transformations. The frame-level feature transformation was based on the time-delay neural network (TDNN) structure [17]. It has been proven that using TDNN, speech features extracted from multi-frame signals with shift-invariance are more effective than those extracted from single-frame signals [18]. The segment-level feature transformation applies statistics pooling to aggregate variable-length frame-level features to obtain a fixed-dimensional vector. In [19], a statistics pooling TDNN was proposed to improve the x-vector learning ability by capturing more robust speaker characteristics.

Nowadays, many studies are devoted to improving SV performance through model design and loss correction. For model design, Tang et al. [20] integrated TDNN and long short-term memory (LSTM) to capture speaker information at different levels. Zhu et al. [21] proposed a self-attention mechanism for DNN-based embedding and computed the embedding as a weighted average of the speaker's frame-level features. For loss correction, Liu et al. [22] proposed the angular softmax (A-Softmax) loss to learn angularly discriminative features in convolutional neural networks (CNNs) for face recognition. Later, it was applied to SV in [23]. Wang et al. [24] proposed the additive margin softmax (AM-Softmax) loss, which simplifies the calculation cost of the A-Softmax loss and has better performance than the A-Softmax. Generally, most SV studies evaluate the model under specific conditions, such as clear pronunciation, high sound quality, and no interference. These studies often ignore the importance of model generalization ability, and models with poor generalization ability will degrade performance in complex real-world

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Contract No. MOST 108-2221-E-006-103-MY3.

Q.-B. Hong is with the Graduate Program of Multimedia Systems and Intelligent Computing, National Cheng Kung University and Academia Sinica, Taiwan (e-mail: qbhong75@gmail.com).

C.-H. Wu is with the Department of Computer Science and Information Engineering, National Cheng Kung University (e-mail: chunghsienwu@gmail.com).

H.-M. Wang is with the Institute of Information Science, Academia Sinica, Taiwan (e-mail: whm@iis.sinica.edu.tw).

environments. More specifically, the ability to generalize to differences in training and testing conditions is critical for SV performance. Therefore, several studies have tried to ensure the model's generalization ability. For example, Kye et al. [25] combined the episodic classification loss with the global classification loss to obtain well-matched and discriminative speaker embeddings. Wan et al. [26] proposed a metric learning-based loss to improve the generalization ability of the model. Kim et al. [27] applied the mean teacher network to produce highly accurate and stable representations to improve the generalization performance of open-set SV tasks.

From the perspective of neural networks, multi-task learning (MTL) [28] was proposed to deal with multiple tasks through a single shared structure. From this viewpoint, MTL has also been used to improve the model's generalization ability. For example, Zhang et al. [29]-[30] indicated that MTL can be regarded as the generalization of multi-label learning and multi-output regression, which can leverage useful information contained in different tasks to improve the generalization ability. However, if the training dataset does not have multiple types of labels, MTL cannot be used for model training. Meyerson et al. [31] proposed pseudo-task augmentation (PTA) to simulate MTL performance when only a single task is available for training. The idea of PTA is to construct multiple decoders to solve the same problem in multiple ways, which can achieve performance improvements similar to MTL. On the other hand, in practical applications, the recorded signal often contains various interferences, which reduce the quality of speaker embedding, especially the interference from other speakers, which is difficult to detect and remove. Therefore, the anti-interference performance of SV remains to be solved.

In this study, we aim to improve the model generalization ability to deal with mismatches of recorded scenarios and languages between training and testing conditions, and to improve the anti-interference ability from other speakers in embedding similarity measurement. To achieve the goal, we propose three new approaches.

1) *Parent Embedding Learning (PEL)*: We train the speaker identification model using two classifiers for the same training task to leverage the generalization ability of the shared structure to improve speaker embedding extraction.
2) *Residual Block PEL*: We add a shortcut connection or Res2Net structure between two consecutive segment-level layers to improve speaker embedding extraction.
3) *Partial Adaptive Score Normalization (PAS-Norm)*: We introduce a novel strategy in the similarity comparison of speaker embeddings to reduce the influence of interference from other speakers.

The remainder of this paper is organized as follows. A state-of-the-art (SOTA) system for SV is described in Section II. The PEL-based speaker embedding extraction method and the PAS-Norm process are presented in Section III. In Section IV, we introduce the datasets and provide the experimental results in detail. Finally, conclusions are given in Section V.

II. Speaker Embedding Learning For SV

One of the most popular speaker embedding learning systems is the x-vector model. In the x-vector model, first, each utterance is fed to the frame-level layers to obtain low-level features, where TDNN is commonly used to encode time-varying signals by operating with multiple interconnected layers. Second, as each utterance is recorded in a variable length of time, a pooling (such as statistics pooling) layer is applied to aggregate variable-length frame-level features into a fixed-dimensional segment-level feature vector. Then, the feature vector is passed through one or two fully-connected layers to obtain the non-linear features as the speaker embedding, which is used to train a speaker classifier. Generally, a speaker discriminative network is trained by a large amount of speech data from multiple speakers to learn the diversity of speech characteristics of different speakers. In SV, the SOTA method for speaker discrimination is based on the probabilistic linear discriminant analysis (PLDA) backend, which takes into account the intra-speaker variation and inter-speaker difference.

A. Extended TDNN

In the VoxCeleb Speaker Recognition Challenge (VoxSRC) 2019, the BUT system [32] and JHU-HLTCOE system [33] utilized the extended TDNN (E-TDNN) x-vector architecture [34] for speaker embedding extraction, and both won the top three in the challenge. The x-vector model is composed of frame-level and segment-level feature transformations. E-TDNN is applied to frame-level transformation, in which a fully-connected layer is added between two 1D convolutional layers for additional feature transformation.

B. ECAPA-TDNN

The ECAPA-TDNN architecture emphasizes channel attention, propagation, and aggregation in frame-level feature transformation [35]. First, instead of standard statistics pooling (global mean and standard deviation), ECAPA-TDNN utilizes channel- and context-dependent statistics pooling to aggregate all frame-level features to form a fixed dimensional vector. Second, the Res2Net module [36] and the squeeze-excitation (SE) block [37] are combined into SE-Res2Block, which may be beneficial for rescaling frame-level features, processing multi-scale features through hierarchical residual connections, and reducing the number of network parameters. Third, multi-layer feature aggregation (MFA) is introduced to combine the output features of all the SE-Res2Blocks for each frame. In VoxSRC 2020, the ECAPA-TDNN model won first place in the challenge. This architecture has become a standard configuration for speaker embedding extraction in many speaker-related tasks. In VoxSRC 2021, the IDLab team introduced the ECAPA CNN-TDNN [38] and ResNet-based models, which ranked third in the challenge.

C. Additive Margin Softmax Loss

The speaker classifier training for SV tasks is about how many distinguishing characteristics of speakers can be learned. The softmax function is most commonly used for probability prediction in classification tasks. In recent years, some studies

have tried to improve the performance of softmax loss by increasing the angular distance of the inter-class feature vectors and decreasing the angular distance of the intra-class feature vectors [22]-[24]. In [24], the cosine-based AM-Softmax loss is formulated as,

$$\mathcal{L}_{ams} = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{e^{s\cdot(cos\theta_{\Psi_n}-m)}}{e^{s\cdot(cos\theta_{\Psi_n}-m)}+\sum_{k=1,k\neq\Psi_n}^{K}e^{s\cdot cos\theta_k}} = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{e^{s\cdot(W_{\Psi_n}^{\mathrm{T}}\hbar_n-m)}}{e^{s\cdot(W_{\Psi_n}^{\mathrm{T}}\hbar_n-m)}+\sum_{k=1,k\neq\Psi_n}^{K}e^{sW_k^{\mathrm{T}}\hbar_n}}, \quad (1)$$

where θ_k is the angle between W_k and $\hbar_n$, $W_{\Psi_n} \in \mathbb{R}^{D\times 1}$ is the weight vector used to predict the likelihood of the target speaker Ψ_n of the n-th training sample, $\hbar_n \in \mathbb{R}^{D\times 1}$ is the speaker embedding, K is the number of training speakers, and N is the number of training samples in a mini-batch. In order to follow the cosine definition, L2 normalization is applied to the features and weights. For all classes, a scale value s is used to avoid the large loss caused by the smaller feature norms. For the target class, a margin value m is used for angular margin adjustment in the training stage. The AM-Softmax loss can effectively improve the classification performance by constraining the learned features to be discriminative on a hypersphere manifold.

D. Backend for Scoring

Cosine similarity and PLDA [40]-[41] are the most commonly used scoring methods in SV tasks, in which the cosine similarity score is normalized using adaptive score normalization (AS-Norm) [39] to obtain a more discriminative score as

$$score' = \frac{1}{2}\left(\frac{score-\mu(\mathcal{S}_1)}{\sigma(\mathcal{S}_1)}+\frac{score-\mu(\mathcal{S}_2)}{\sigma(\mathcal{S}_2)}\right), \quad (2)$$

where $\mu(\mathcal{S}_k)$ and $\sigma(\mathcal{S}_k)$ are the mean and standard deviation of scores calculated by matching ϕ_k against a given impostor cohort set.

III. Proposed Methods

As mentioned in the previous section, a robust embedding extractor must have the ability to distinguish the voices of different speakers. One method is to adjust the characteristics of the embedding model, such as using the AM-Softmax loss for model training to effectively increase the inter-class feature angular distance and decrease the intra-class feature angular distance. However, when applied to SV tasks, the pre-trained embedding model is always used to evaluate speech signals with mismatched conditions in practical applications. Therefore, the generalization ability of the embedding model is very important, but it is rarely studied. Moreover, the observation signal in the real-life environment often contains various interferences, which seriously affect the SV performance. Therefore, we propose a parent embedding learning (PEL) approach to improve the generalization ability of the embedding model and a partial adaptive score normalization (PAS-Norm) method to solve the weakness of SV under interference conditions.

A. Parent Embedding Learning (PEL)

MTL is one of the methods to improve the model's generalization ability. A standard architecture is to solve the problems of multiple tasks simultaneously by constraining the shared structure learning, thereby effectively improving the processing ability of the held-out data. However, most datasets do not meet the requirements of MTL, and the most important factor is the cost of labeling. In contrast, pseudo-task augmentation (PTA) [31] uses multiple decoders to perform a single task to approximate the MTL performance.

Classical MTL is based on the use of a shared low-level structure and multiple high-level decoders to solve multiple tasks. The multiple decoders that obtain corresponding results can be defined as

$$\hat{y}_{j,n} = \mathcal{D}_j(\mathcal{W}(u_n;\Theta_{\mathcal{W}});\Theta_{\mathcal{D}_j}), \quad (3)$$

where $\mathcal{W}$ denotes the shared structure with parameters $\Theta_{\mathcal{W}}$, u_n is the n-th input data, and the decoder $\mathcal{D}_j$ is used to predict the result of the j-th task. In PTA, all the J decoders are trained to solve the same task to simulate MTL performance.

Inspired by the better model generalization ability of MTL and PTA, we propose a parent embedding learning (PEL) method for the segment-level layer in the classification model to improve the generalization ability of the speaker embedding extractor. This study differs from MTL and PTA applied in the same layer because a neural network contains variant attributes between different hidden layers [55]-[56]. PEL leverages different attributes of different embedding layers to improve speaker discriminative ability. The proposed architecture is shown in Fig. 1. In addition to the original speaker classifier connected to the last fully-connected layer, we use the second speaker classifier to connect to the penultimate fully-connected layer. In other words, the last two fully-connected layers are trained by two separate speaker classifiers to produce two corresponding embeddings (i.e., Parent Embedding and Child Embedding). In our setting, Parent Embedding is forwarded not only to Speaker Classifier 1, but also to Speaker Classifier 2. Because Parent Embedding is related to the training of two classifiers, it should have better generalization ability as suggested by PTA.

Suppose two losses are obtained from Speaker Classifiers 1 and 2,

$$\mathcal{L}(y_n,\hat{y}_{1,n};\Theta_{\mathcal{D}_1}) = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{e^{s\cdot(\Theta_{\mathcal{D}_1,\Psi_n}^{\mathrm{T}}\hbar_{1,n}-m)}}{e^{s\cdot(\Theta_{\mathcal{D}_1,\Psi_n}^{\mathrm{T}}\hbar_{1,n}-m)}+\sum_{k=1,k\neq\Psi_n}^{K}e^{s\Theta_{\mathcal{D}_1,k}^{\mathrm{T}}\hbar_{1,n}}}, \quad (4)$$

$$\mathcal{L}(y_n,\hat{y}_{2,n};\Theta_{\mathcal{D}_2}) = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{e^{s\cdot(\Theta_{\mathcal{D}_2,\Psi_n}^{\mathrm{T}}\hbar_{2,n}-m)}}{e^{s\cdot(\Theta_{\mathcal{D}_2,\Psi_n}^{\mathrm{T}}\hbar_{2,n}-m)}+\sum_{k=1,k\neq\Psi_n}^{K}e^{s\Theta_{\mathcal{D}_2,k}^{\mathrm{T}}\hbar_{2,n}}}. \quad (5)$$

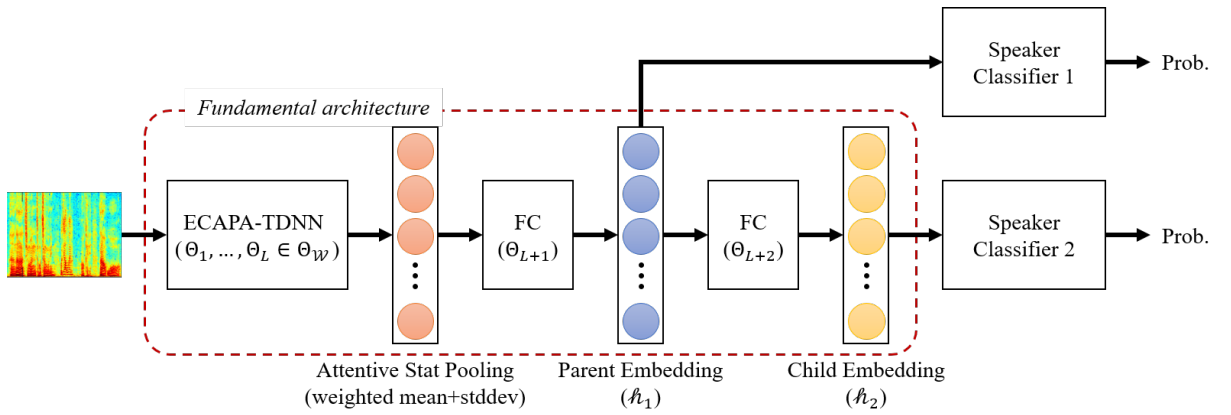
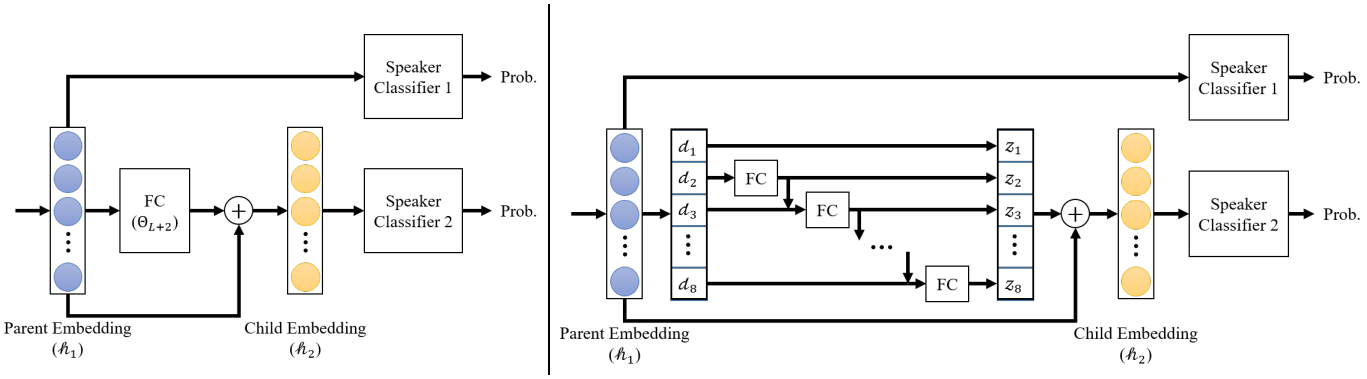


Fig. 1. The architecture of parent embedding learning. The fundamental architecture is the same as that of the baseline ECAPA-TDNN x-vector model, except that we use an additional speaker classifier for Parent Embedding training.



(a) Residual Connection PEL

(b) Res2Net PEL ($\ell = 8$)

Fig. 2. The segment layers of Parent Embedding and Child Embedding in different PEL structures.

Parent Embedding $\hbar_{1,n}$ is formed by the penultimate fully-connected layer with parameters Θ_{L+1}, optimized by gradient descent as

$$\Theta_{L+1} := \Theta_{L+1} - \eta \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{2} \nabla_{\Theta_{L+1}} (\mathcal{L}(y_n, \hat{y}_{j,n}; \Theta_{\mathcal{D}_j})) , \tag{6}$$

where η is the learning rate. For the n-th training sample, the gradient can be written as,

$$\sum_{j=1}^{2} \nabla_{\Theta_{L+1}} (\mathcal{L}(y_n, \hat{y}_{j,n}; \theta_{\mathcal{D}_j})) = \nabla_{\Theta_{L+1}} (\mathcal{L}(y_n, \hat{y}_{1,n}; \Theta_{\mathcal{D}_1})) + \nabla_{\hbar_{1,n}} (\mathcal{L}(y_n, \hat{y}_{2,n}; \Theta_{\mathcal{D}_2})) \cdot \nabla_{\Theta_{L+1}} (\hbar_{1,n}) . \tag{7}$$

On the other hand, Child Embedding $\hbar_{2,n}$ is formed by the last fully-connected layer with parameters Θ_{L+2}, optimized by

$$\Theta_{L+2} := \Theta_{L+2} - \eta \frac{1}{N} \sum_{n=1}^{N} \nabla_{\Theta_{L+2}} (\mathcal{L}(y_n, \hat{y}_{2,n}; \Theta_{\mathcal{D}_2})). \tag{8}$$

B. *Residual Block PEL*

Residual connection [42] not only effectively avoids the vanishing gradient problem in many deep neural networks, but also maintains the relationship of features in the interconnection. In the previous section, PEL aims to improve the generalization ability of Parent Embedding. As shown in Fig. 2(a), we can add a shortcut connection between Parent Embedding and Child Embedding to obtain different but closely-related representations, which may allow the model to learn more robust speaker embeddings. In this way, the Child Embedding layer is defined as follows.

$$\hbar_2 = a(\hbar_1; \Theta_{L+2}) + \hbar_1 , \tag{9}$$

where $a(\cdot)$ denotes a non-linear activation function. With the residual mapping structure, the output of the feedforward neural network is added to the original input (i.e., element-wise

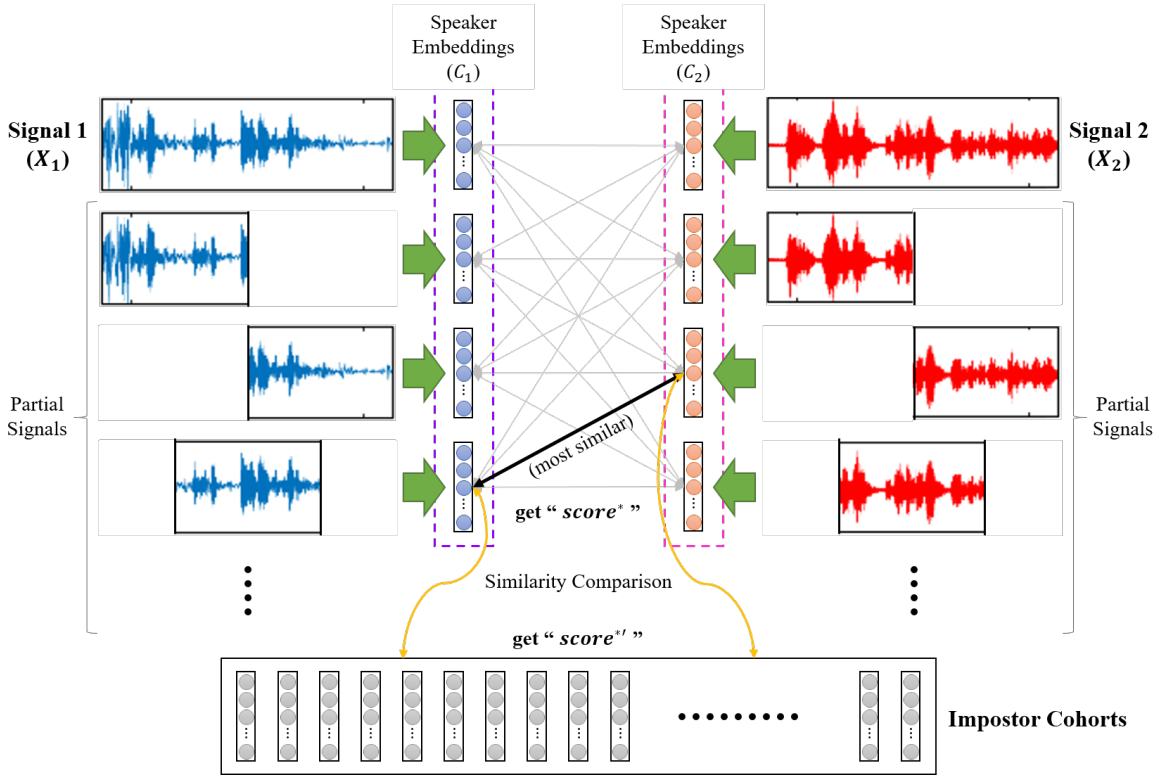


Fig. 3. The PAS-Norm process. There are two steps to obtain the similarity score of a pair of speech signals. The first step is to find the largest similarity score from all candidate embedding pairs (i.e., get "$score^*$""), the second step is to normalize the largest similarity score by the impostor cohort (i.e., get "$score^{*\prime}$").

addition) to form a closely-related output.

In recent studies, multi-scale feature extraction has largely replaced layer-wise feature extraction in ResNet [57]-[59]. Models based on Res2Net [36] exploit the multi-scale capability to increase the range of the receptive field and reduce the number of parameters. For PEL, we can simplify Res2Net from a convolutional operator to a fully-connected operator for embedding layer training, as shown in Fig. 2(b). The Child Embedding is computed by

$$z_q = \begin{cases} d_q & q = 1; \\ a(d_q; \Theta_{L+2}^q) & q = 2; \\ a(d_q + z_{q-1}; \Theta_{L+2}^q) & 2 < q \le \ell\,, \end{cases} \tag{10}$$

$$\hbar_2 = \begin{bmatrix} z_1 \\ \vdots \\ z_\ell \end{bmatrix} + \hbar_1\,, \tag{11}$$

where d_q is the q-th subset of $\hbar_1$, Θ_{L+2}^q denotes the parameters of the q-th fully-connected block in Res2Net, and ℓ is the scale dimension. ℓ was set to 8 in this study.

C. Partial Adaptive Score Normalization (PAS-Norm)

Most SV studies are conducted on speech data without interference, and good performance can usually be obtained. However, in a real-life environment, the recorded signal often contains various interferences, which greatly decrease the performance of SV. Therefore, how to improve the anti-interference ability of an SV system is a big challenge. As a result, we propose the partial adaptive score normalization (PAS-Norm) method to improve SV performance under interference conditions.

A time-domain speech signal $X = \{x_t : 1 \le t \le T\}$ can be decomposed into three component sets,

$$X_{target} = \{x_t : x_t \in O_{target}\}\,, \tag{12}$$
$$X_{nontarget} = \{x_t : x_t \in O_{nontarget}\}\,, \tag{13}$$
$$X_{overlap} = \{x_t : x_t \in O_{target} \cup O_{nontarget}\}\,, \tag{14}$$
$$X = \{X_{target}, X_{nontarget}, X_{overlap}\}\,, \tag{15}$$

where X_{target} denotes the signal from the target speaker O_{target}, $X_{nontarget}$ denotes the signal from the non-target speaker $O_{nontarget}$, and $X_{overlap}$ denotes the mixed signal from the target and non-target speakers. $X_{nontarget}$ may also cover non-speech components such as noise and silence. Ideally, removing $X_{nontarget}$ and $X_{overlap}$ from X can effectively avoid the influence of interference and improve the quality of speaker representation.

In practical applications, when the target speaker' voice is not fully collected, $X_{nontarget}$ and $X_{overlap}$ are difficult to distinguish from X_{target}. Therefore, in this study, we adopt a simple equal division method to divide the input signal into N segments (partial ranges). For example, when N=3, the input

signal is divided into the first half, the second half, and the middle half, as shown in Fig. 3. Given a pair of signals $[X_1, X_2]$ with interference, when the speaker embedding is extracted using the full signal, the quality of the speaker embedding will be corrupted by the interference, which in turn affects the similarity measure. Similarity comparison of partial signals may reduce the influence of interference.

Suppose $C_1 = \{\phi_1^{(0)}, \phi_1^{(1)}, \phi_1^{(2)}, \dots, \phi_1^{(N)}\}$ is the candidate embedding set of X_1 and $C_2 = \{\phi_2^{(0)}, \phi_2^{(1)}, \phi_2^{(2)}, \dots, \phi_2^{(N)}\}$ is the candidate embedding set of X_2, where $\phi_1^{(0)}$ and $\phi_2^{(0)}$ are the embeddings extracted from the full signals X_1 and X_2, and the remaining embeddings represent the N partial ranges of the signals. If $\phi_1^{(\cdot)}$ and $\phi_2^{(\cdot)}$ are from the same speaker, then

$$\mathcal{F}\left(\phi_1^{(\cdot)}, \phi_1^{(\cdot)}\right) \approx \mathcal{F}\left(\phi_2^{(\cdot)}, \phi_2^{(\cdot)}\right) \approx \mathcal{F}\left(\phi_1^{(\cdot)}, \phi_2^{(\cdot)}\right), \tag{16}$$

where $\mathcal{F}$ is a similarity comparison function. Otherwise, we will observe the relationship as,

$$\mathcal{F}\left(\phi_1^{(\cdot)}, \phi_1^{(\cdot)}\right) \approx \mathcal{F}\left(\phi_2^{(\cdot)}, \phi_2^{(\cdot)}\right) \gg \mathcal{F}\left(\phi_1^{(\cdot)}, \phi_2^{(\cdot)}\right). \tag{17}$$

When there is interference, we will see that most candidate embedding pairs have low similarity scores, even if the two speech signals are from the same speaker. In other words, only a few candidate embedding pairs corresponding to the unperturbed partial signals of the same speaker can get high similarity scores. Therefore, when the speech signal contains more than two speakers, the similarity comparison of partial signals reduces the possibility of being interfered with by other speakers, and the maximum similarity score of two candidate embeddings can be considered as a reliable similarity measure for speech signal pairs. In the proposed PAS-Norm process, the first step is to find the highest scoring candidate pair among all candidate pairs as the similarity comparison score of two speech signals, i.e.,

$$(i^*, j^*) = \arg\max_{(i,j)} \mathcal{F}\left(\phi_1^{(i)}, \phi_2^{(j)}\right), \tag{18}$$

$$score^* = \mathcal{F}\left(\phi_1^{(i^*)}, \phi_2^{(j^*)}\right). \tag{19}$$

Then, the second step is to normalize the similarity comparison score by the impostor trials as

$$\mathcal{S}_1^{(i^*)} = \begin{bmatrix} \mathcal{F}\left(\phi_1^{(i^*)}, \phi_{cohorts_1}^{(1)}\right) \\ \vdots \\ \mathcal{F}\left(\phi_1^{(i^*)}, \phi_{cohorts_1}^{(M)}\right) \end{bmatrix}, \tag{20}$$

$$\mathcal{S}_2^{(j^*)} = \begin{bmatrix} \mathcal{F}\left(\phi_2^{(j^*)}, \phi_{cohorts_2}^{(1)}\right) \\ \vdots \\ \mathcal{F}\left(\phi_2^{(j^*)}, \phi_{cohorts_2}^{(M)}\right) \end{bmatrix}, \tag{21}$$

$$score^{*\prime} = \frac{1}{2}\left(\frac{score^* - \mu(\mathcal{S}_1^{(i^*)})}{\sigma(\mathcal{S}_1^{(i^*)})} + \frac{score^* - \mu(\mathcal{S}_2^{(j^*)})}{\sigma(\mathcal{S}_2^{(j^*)})}\right), \tag{22}$$

where $\mathcal{S}_1^{(i^*)}$ and $\mathcal{S}_2^{(j^*)}$ are the scores of the top M most similar cohorts for $\phi_1^{(i^*)}$ and $\phi_2^{(j^*)}$, respectively. All terms in Eq. (22) depend on the representation pair $[\phi_1^{(i^*)}, \phi_2^{(j^*)}]$ with the least interference influence on the embedding, so the final score is more stable than that of AS-Norm and other commonly used normalization methods.

IV. Experimental Results

A. Datasets

In this study, the speaker embedding models were trained on the VoxCeleb2 dataset [43] and evaluated on four other datasets under different conditions, including the VoxCeleb1 dataset [44], the Librispeech dataset [45], the Speakers in the Wild (SITW) dataset [46], and the CN-Celeb dataset [47].

1) Training data

The VoxCeleb2 dataset contains over 1 million utterances (mostly English) collected from 6,112 speakers on YouTube, and is divided into two subsets for SV: *DEV* and *TEST*. The *DEV* set contains 1,092,009 utterances from 5,994 speakers and has no overlap with the speakers in the VoxCeleb1 dataset. In this study, the *DEV* set of the VoxCeleb2 dataset was used to train the embedding extraction models. We followed the popular data augmentation strategy of the Kaldi [48] recipe to generate extra samples for model training. In Kaldi /egs/voxceleb/v2/ recipe, the MUSAN dataset (babble, noise, music) [49] and the RIR dataset (reverb) [50] were combined with the original utterances to generate 4 extra samples for each utterance. It should be noted that only 1,000,000 samples were chosen at random from the 4,368,036 extra samples to train the model with the original 1,092,009 utterances in the Kaldi recipe strategy.

2) Testing data

- The VoxCeleb1 dataset contains 153,516 utterances collected from 1,251 speakers on YouTube. In this dataset, three trial lists formed by different conditions [51] are commonly used for SV evaluation experiments.
- The Librispeech dataset contains approximately 1,000 hours of read English speech. We manually generated a trial list using the train-clean-100 subset of 28,539 audio files from 251 speakers and evaluated the SV performance under clean speech conditions.
- The SITW dataset contains samples of about 300 people in different scenarios, and there are often multiple speakers in the recordings. The *EVAL* subset contains 2,883 audio files from 180 speakers, and is divided into four trial lists to evaluate SV under interference from other speakers.
- The CN-Celeb dataset contains 130,109 utterances from 1,000 Chinese speakers across 11 different genres. It can be used to evaluate SV in different genres in real-life settings, such as interviews, singing, and advertising.

B. Experimental Setup

In the experiments, all audio files were labeled with speaker identities, and silent intervals were removed using energy-based voice activity detection. The input feature vector contains

80-dimentional Mel-frequency cepstral coefficients (MFCCs) extracted based on a 25ms window with a stride of 10ms. The training data consists of randomly sampled 2 to 4 second segments from the training utterances. All models were trained with a cyclical learning rate [52] between 0.01 and 0.1. The duration was set to 130k steps for each cycle, and each model was trained for 10 cycles. The learning rate decayed by a factor of 0.75 per cycle. Early stopping was used to avoid overfitting. All models were trained by AM-Softmax with a margin value of 0.2 and a scale value of 30. Stochastic gradient descent (SGD) with a momentum of 0.9 was applied for model optimization. All weights used L2-regularization with a Lagrange multiplier of 2e-5 for weight decay. The mini-batch size was set to 128.

The E-TDNN x-vector architecture in [32] was replicated as the baseline. For the frame-level layers, each hidden layer had 1,024 output channels, except for the last hidden layer, which had 3,072 output channels. The statistics pooling layer had 6,144 output nodes (3,072 nodes for mean and 3,072 nodes for standard deviation). For the segment-level layers, the two consecutive fully-connected layers had 192 output channels (i.e., the dimension of the resulting speaker embedding was 192). The leaky rectified linear unit (Leaky ReLU) [53] activation function with a negative reserve ratio of 0.2 and batch normalization (BN) were applied to each transformation layer for non-linear mapping. The final softmax layer had 5,994 output channels because there were 5,994 training speakers. The baseline ECAPA-TDNN x-vector architecture was copied from [35]. Each SE-Res2Block had 1,024 output channels. The scale dimension in Res2Block was set to 8. The bottleneck dimension in SE-Block was set to 128. The two consecutive embeddings had 192 channels, which were also trained by predicting the probabilities of 5,994 speakers. In the proposed PEL-based models, an additional speaker classifier was used to train the first embedding on the E-TDNN and ECAPA-TDNN x-vector architectures (named E-TDNN-PEL and ECAPA-TDNN-PEL). To further improve speaker representation learning, we added a shortcut connection between the two consecutive embeddings of the ECAPA-TDNN-PEL x-vector architecture (named ECAPA-TDNN-ResPEL) and a Res2Net between the two consecutive embeddings of the ECAPA-TDNN-PEL x-vector architecture (named ECAPA-TDNN-Res2PEL).

Two similarity measures were used to compare a pair of embeddings in this study, including the cosine distance and the two-covariance PLDA score. For score normalization, the impostor cohorts were generated by averaging the speaker embeddings for each speaker in the VoxCeleb2 *DEV* set (i.e., the number of impostor cohorts was 5,994). For AS-Norm and the proposed PAS-Norm methods, the scores of the top 200 most similar cohorts were used for score normalization (i.e., $M = 200$). In PAS-Norm, three ranges were used to generate the partial signals (i.e., $N = 3$), including $1 \sim T/2$, $T/2 \sim T$, and $T/4 \sim 3T/4$. The training and testing datasets were prepared by the Kaldi toolkit, and all neural network models were implemented by the TensorFlow playform.

We reported SV results in term of equal error rate (EER) and minimum detection cost function (minDCF) with $p_{target} = 0.01$.

C. Joint Parent Embedding Learning

Firstly, we evaluated the effectiveness of PEL in model training. The E-TDNN and ECAPA-TDNN architectures were used, and the differences in embedding learning were compared. The baseline E-TDNN and ECAPA-TDNN architectures were trained with a single classification layer. The E-TDNN-PEL and ECAPA-TDNN-PEL architectures were trained with an additional classifier to obtain Parent Embedding, which was expected to have better speaker discriminative ability than the embeddings learned in E-TDNN and ECAPA-TDNN.

Table I shows the results of SV experiments on the VoxCeleb1-E (cleaned) and VoxCeleb1-H (cleaned) trial lists. All models compared are based on the E-TDNN architecture. E-TDNN-PTA was trained with two decoders connected to the Child Embedding layer. For the task involving a large number of verification speakers (i.e., the VoxCeleb1-E (cleaned) trial list), the proposed PEL-based embedding systems significantly outperformed the baseline systems. Compared with E-TDNN, E-TDNN-PEL reduced the EER by 7.6% (1.57% vs 1.7%). Compared with E-TDNN-PTA, E-TDNN-PEL reduced the EER by 6.5% (1.57% vs 1.68%). For the task with identity restrictions (i.e., the VoxCeleb1-H (cleaned) trial list), the proposed E-TDNN-PEL systems also outperformed the E-TDNN and E-TDNN-PTA systems and reduced the EER by 5.3% (2.67% vs 2.82%) and 3.6% (2.67% vs 2.77%), respectively. Obviously, using an additional speaker classifier connected to the penultimate fully-connected layer increases the representation ability, and PEL that captures different attributes at different layers effectively improves the SV performance, compared with PTA that captures different attributes at the same layer. On the other hand, in E-TDNN-PEL Plus, three separate speaker classifiers were respectively connected to the last three fully-connected layers for training. The experimental result showed that the Parent Embedding $\hbar_1$ of E-TDNN-PEL Plus obtained similar performance to the Parent Embedding $\hbar_1$ of E-TDNN-PEL. The reason may be that Parent Embedding $\hbar_1$ is mainly learned through Speaker Classifier 1 and the second fully-connected layer with parameters Θ_{L+2} (as shown in Fig. 1), so using more additional embedding layers in PEL does not improve its performance.

Table II shows the results of SV experiments using ECAPA-TDNN with PEL architectures. For evaluation on the VoxCeleb1-E (cleaned) trial list, compared with ECAPA-TDNN, ECAPA-TDNN-PEL reduced the EER by 6.0% (1.25% vs 1.33%). For evaluation on the VoxCeleb1-H (cleaned) trial list, ECAPA-TDNN-PEL achieved the best performance, and the EER was reduced by 4.9% (2.14% vs. 2.25%) compared with the ECAPA-TDNN systems. Furthermore, the PEL-based embedding systems were also superior to the baseline embedding systems in terms of minDCF, which means that the PEL-based embedding systems had better performance than the baseline embedding systems in achieving a low false positive rate. On the other hand, we can see that the baseline embedding systems achieved higher performance on Child Embedding (the traditionally commonly used layer), but the PEL-based

TABLE I
PERFORMANCE OF SPEAKER VERIFICATION EXPERIMENTS ON THE VOXCELEB1-E AND VOXCELEB1-H TRIAL LISTS USING E-TDNN WITH DIFFERENT SPEAKER EMBEDDING LEARNING PROCEDURES

System		Backend	VoxCeleb1-E (cleaned)		VoxCeleb1-H (cleaned)	
			EER (%)	minDCF	EER (%)	minDCF
E-TDNN	Parent Embedding $\hbar_1$	Cosine	2.20	0.2181	3.60	0.3045
		PLDA	1.73	0.2184	3.12	0.3445
	Child Embedding $\hbar_2$	Cosine	**1.70**	**0.1791**	**2.82**	**0.2582**
		PLDA	1.88	0.2263	3.29	0.3442
E-TDNN + PTA (the two decoders connected to the child embedding layer are trained by two separate speaker classifiers)	Parent Embedding $\hbar_1$	Cosine	2.41	0.2346	3.94	0.3111
		PLDA	1.71	0.2174	3.06	0.3374
	Child Embedding A $\hbar_{2A}$	Cosine	**1.68**	**0.1743**	2.77	0.2444
		PLDA	2.13	0.2331	3.62	0.3397
	Child Embedding B $\hbar_{2B}$	Cosine	1.68	0.1745	**2.77**	**0.2442**
		PLDA	1.89	0.2265	3.30	0.3433
E-TDNN-PEL (the last two fully-connected layers are trained by two separate speaker classifiers)	Parent Embedding $\hbar_1$	Cosine	1.59	**0.1639**	**2.67**	**0.2427**
		PLDA	**1.57**	0.1968	2.83	0.2960
	Child Embedding $\hbar_2$	Cosine	1.78	0.1856	2.95	0.2645
		PLDA	1.81	0.2074	3.17	0.3092
E-TDNN-PEL Plus (the last three fully-connected layers are trained by three separate speaker classifiers)	Parent Embedding $\hbar_1$	Cosine	1.59	**0.1657**	**2.69**	**0.2433**
		PLDA	**1.56**	0.1978	2.83	0.2967
	Child Embedding $\hbar_2$	Cosine	1.75	0.1878	2.94	0.2649
		PLDA	1.76	0.2108	3.21	0.3164
	Child Embedding $\hbar_3$	Cosine	1.91	0.2023	3.20	0.2952
		PLDA	1.94	0.2259	3.54	0.3385

TABLE II
PERFORMANCE OF SPEAKER VERIFICATION EXPERIMENTS ON THE VOXCELEB1-E AND VOXCELEB1-H TRIAL LISTS USING ECAPA-TDNN WITH PEL ARCHITECTURES

System		Backend	VoxCeleb1-E (cleaned)		VoxCeleb1-H (cleaned)	
			EER (%)	minDCF	EER (%)	minDCF
ECAPA-TDNN	Parent Embedding $\hbar_1$	Cosine	1.50	0.1509	2.51	0.2165
		PLDA	1.42	0.1777	2.59	0.2828
	Child Embedding $\hbar_2$	Cosine	**1.33**	**0.1398**	**2.25**	**0.2005**
		PLDA	1.67	0.1931	2.96	0.2969
ECAPA-TDNN-PEL (the last two fully-connected layers are trained by two separate speaker classifiers)	Parent Embedding $\hbar_1$	Cosine	**1.25**	**0.1383**	**2.14**	**0.1940**
		PLDA	1.31	0.1591	2.39	0.2480
	Child Embedding $\hbar_2$	Cosine	1.41	0.1510	2.44	0.2219
		PLDA	1.73	0.1811	3.03	0.2771

embedding systems achieved higher performance on Parent Embedding; and the baseline and PEL-based embedding systems achieved similar performance on Child Embedding, but PEL-based Parent Embedding was more effective than baseline Parent Embedding. Therefore, the results confirmed that the PEL-based method can effectively improve the representation learning ability of speaker embeddings.

D. Evaluation under Different Signal Conditions

Although previous experimental results have demonstrated that the proposed PEL-based embedding systems can achieve superior performance on the down-stream SV task on the VoxCeleb1 dataset, the performance in real-world environments may be disturbed in mismatched conditions. In this experiment, three datasets were used to evaluate the impact of mismatched conditions on the SV task, including Librispeech, SITW, and CN-Celeb. The Librispeech dataset was used to evaluate SV performance under clean conditions. Since Librispeech does not provide a trial list for SV evaluation, we selected audio files with IDs from 0001 to 0005 in audio Chapters to generate a simulated trial list by exhaustive pairing. Without pairing the same audio files, the selected 2,925 audio files yielded 8,552,700 trial pairs (i.e., $\#\text{pairs} = 2925^2 - 2925$). Table III shows the EER performance of SV on the trial lists of these three evaluation datasets. Evaluated on the Librispeech simulated trial list consisting of clean speech pairs, ECAPA-TDNN-PEL reduced the EER by 10.9% compared with ECAPA-TDNN (0.49% vs 0.55%). Evaluated on the SITW *EVAL* core-core trial list consisting of pairs without interference, ECAPA-TDNN-PEL reduced the EER by 17.9% compared with ECAPA-TDNN (1.42% vs 1.73%). Evaluated on the SITW *EVAL* core-multi and SITW *EVAL* assist-core trial lists that contain interference in one of the paired audios, compared with ECAPA-TDNN, ECAPA-TDNN-PEL reduced the EER by 6.9% (2.96% vs 3.18%) and 7.4% (3.12% vs 3.37%), respectively. Evaluated on the SITW *EVAL* assist-multi trial list, where both audios contain interference, ECAPA-TDNN-PEL reduced the EER by 7.6% compared with ECAPA-TDNN (3.91% vs 4.23%). Evaluated on the CN-Celeb trial list consisting of pairs with different genres and language mismatched conditions, ECAPA-TDNN-PEL reduced the EER by 8.3% compared with ECAPA-TDNN (12.22% vs 13.32%). Obviously, the ECAPA-TDNN-PEL embedding systems consistently outperformed the baseline ECAPA-TDNN embedding systems on all three datasets. From Table III, we can

TABLE III
EERs (%) OF SPEAKER VERIFICATION EXPERIMENTS ON THE TRIAL LISTS OF THREE EVALUATION DATASETS

Trial List		Backend	E-TDNN		E-TDNN-PEL		ECAPA-TDNN		ECAPA-TDNN-PEL	
			h_1	h_2	h_1	h_2	h_1	h_2	h_1	h_2
Librispeech		Cosine	0.83	**0.57**	**0.54**	0.65	0.60	**0.55**	**0.49**	0.57
		PLDA	0.81	0.79	0.68	0.72	0.70	0.75	0.64	0.65
SITW	core-core	Cosine	4.69	**2.53**	2.38	2.48	1.92	1.74	1.60	1.68
		PLDA	2.65	2.77	**2.16**	2.36	**1.73**	1.95	**1.42**	1.68
	core-multi	Cosine	9.11	**4.26**	3.64	3.90	3.91	**3.18**	**2.96**	3.23
		PLDA	4.90	5.39	**3.50**	3.73	3.81	4.14	3.02	3.56
	assist-core	Cosine	9.34	**4.41**	**3.88**	4.12	4.34	**3.37**	**3.12**	3.49
		PLDA	4.84	5.37	4.00	4.16	4.09	4.33	3.38	4.06
	assist-multi	Cosine	11.47	**5.22**	4.75	4.93	6.01	**4.23**	**3.91**	4.38
		PLDA	6.17	6.99	**4.73**	5.06	5.14	5.44	4.20	5.00
CN-Celeb		Cosine	15.91	**13.61**	13.63	14.47	15.58	**13.32**	**12.22**	13.11
		PLDA	14.81	14.70	**13.46**	14.05	13.52	13.74	12.84	13.57

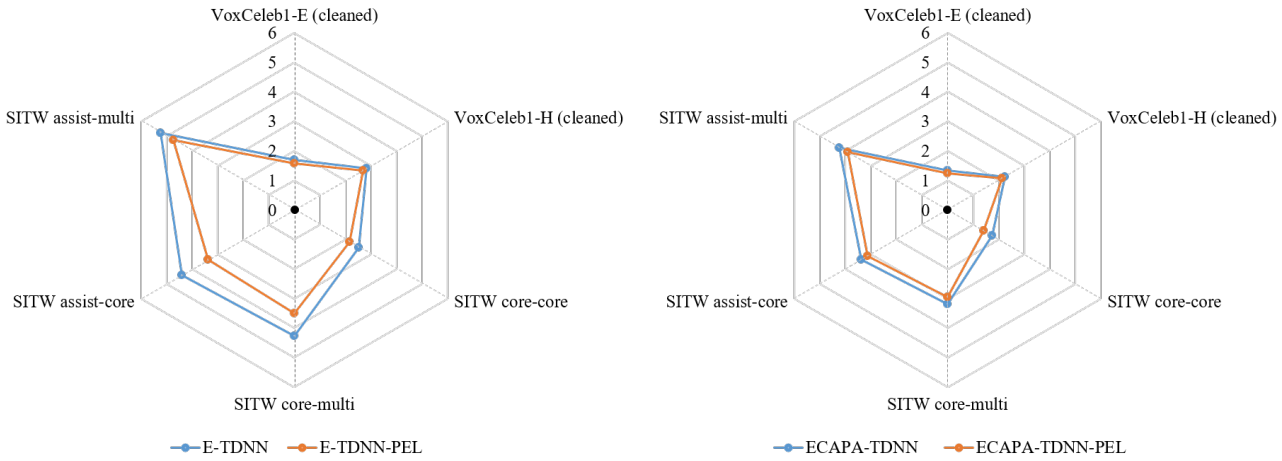


Fig. 4. EERs of PEL-based and baseline embedding systems evaluated on various datasets (based on the best EERs in each trial list).

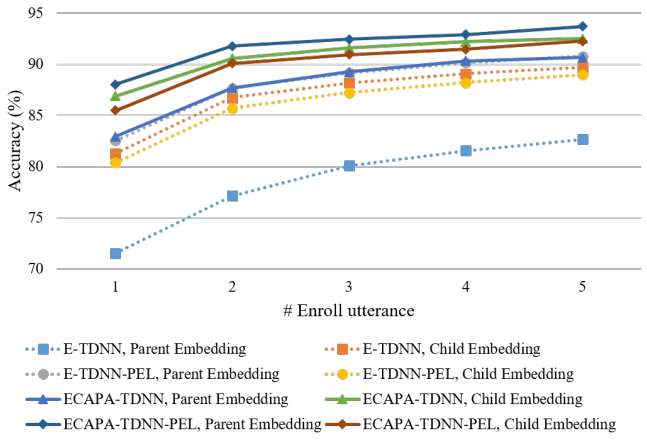


Fig. 5. Performance of speaker identification on varying numbers of enrollment utterances from 1,251 speakers on the VoxCeleb1 dataset.

also see that the E-TDNN-PEL embedding systems were superior to the baseline E-TDNN embedding systems on all three datasets. In addition, it is worth noting that the performance improvement of the PEL-based embedding systems over the baseline embedding systems in out-of-domain was greater than that in in-domain, as shown in Fig. 4. The results confirmed that PEL can effectively increase the generalization ability of speaker embedding.

E. Evaluation under Different Duration Conditions

In this experiment, we analyzed the effect of the number of enrollment utterances on the speaker identification task. We used the VoxCeleb1 speaker identification labels of 1,251 speakers (see iden_split.txt on the VoxCeleb1 webpage) for audio enrollment and evaluation. As shown in Fig. 5, as the number of enrollment utterances increases, the accuracy of all models continues to improve. ECAPA-TDNN-PEL (Parent Embedding) achieved the best performance regardless of the number of enrollment utterances, even if the speaker provided only one utterance for enrollment. The results showed that the PEL-based embeddings show reliable representation ability, and it can be applied to situations with less enrollment data to achieve minimal recognition errors.

F. Residual Block PEL

As described in Section III-B, different but closely-related Parent Embedding and Child Embedding can be learned by residual mapping. Furthermore, multi-scale feature extraction between Parent Embedding and Child Embedding can be implemented by the Res2Net structure. In this experiment, we

TABLE IV
PERFORMANCE OF PEL-BASED AND RESPEL-BASED EMBEDDING SYSTEMS ON THE VOXCELEB1-E AND VOXCELEB1-H TRIAL LISTS USING COSINE SIMILARITY

System	Input Feature (Dim.-Type)	#Aug.	#Aug. Samples	#Params	Loss Function		VoxCeleb1-E (cleaned) EER (%)	VoxCeleb1-E (cleaned) minDCF	VoxCeleb1-H (cleaned) EER (%)	VoxCeleb1-H (cleaned) minDCF
ECAPA-TDNN (C=1024) [35]	80-MFCCs	✓	6,552,054	14.7M	AAM-Softmax		1.12	0.1318	2.12	0.2101
MFA-TDNN (Standard) [61]	80-FBanks	✓	5,460,045	7.3M	AAM-Softmax		1.08	**0.1175**	2.05	0.1897
ARET-25 [63]	161-Spec.	×	0	12.2M	AAM-Softmax		1.52	-	2.61	-
ResNet18 [35]	80-MFCCs	✓	6,552,054	13.8M	AAM-Softmax		1.60	0.1789	2.88	0.2672
ResNet34 [35]	80-MFCCs	✓	6,552,054	23.9M	AAM-Softmax		1.33	0.1560	2.46	0.2288
ResNet160 [32]	40-FBanks	✓	5,000,000	-	AAM-Softmax		1.38	-	2.50	-
ResNet256 [32]	40-FBanks	✓	5,000,000	-	AAM-Softmax		1.35	-	2.48	-
Res2Net-26w8s [57]	80-FBanks	✓	-	9.3M	Softmax		1.47	0.1692	2.72	0.2717
H / ASP [60]	64-FBanks	×	0	8.0M	AP+Softmax		1.42	-	2.77	-
H / ASP [60]	64-FBanks	✓	-	8.0M	AP+Softmax		**1.07**	-	2.21	-
ECAPA-TDNN (Re-implemented)	80-MFCCs	✓	1,000,000	14.7M	AM-Softmax	$\hbar_1$	1.50	0.1509	2.51	0.2165
						$\hbar_2$	1.33	0.1398	2.25	0.2005
						concat	1.35	0.1404	2.30	0.2041
ECAPA-TDNN-PEL	80-MFCCs	✓	1,000,000	15.8M*	AM-Softmax	$\hbar_1$	1.25	0.1383	2.14	0.1940
						$\hbar_2$	1.41	0.1510	2.44	0.2219
						concat	1.30	0.1393	2.26	0.2042
ECAPA-TDNN-ResPEL	80-MFCCs	✓	1,000,000	15.8M*	AM-Softmax	$\hbar_1$	1.17	0.1223	**2.03**	0.1895
						$\hbar_2$	1.17	0.1234	2.05	0.1895
						concat	1.16	0.1229	2.04	0.1882
ECAPA-TDNN-Res2PEL	80-MFCCs	✓	1,000,000	15.8M*	AM-Softmax	$\hbar_1$	1.17	0.1260	2.05	**0.1882**
						$\hbar_2$	1.29	0.1423	2.29	2.2061
						concat	1.17	0.1259	2.05	0.1882

* Compared with ECAPA-TDNN, the proposed ECAPA-TDNN-PEL, ECAPA-TDNN-ResPEL and ECAPA-TDNN-Res2PEL models used an additional classifier with about 1.15M parameters (i.e., #Params $= 192 \times 5994$) for Parent Embedding learning. In the evaluation stage, the additional classifier was not used, and the total number of parameters remained the same as the ECAPA-TDNN model.

investigated whether the ECAPA-TDNN-PEL embedding system can be improved by introducing a residual connection or Res2Net structure between Parent Embedding and Child Embedding. As shown in Table IV, evaluated on the VoxCeleb1-E (cleaned) and VoxCeleb1-H (cleaned) trial lists, ECAPA-TDNN-ResPEL outperformed ECAPA-TDNN-PEL and reduced the EER by 6.4% (1.17% vs 1.25%) and 5.1% (2.03% vs 2.14%), respectively. The proposed systems were also compared with other SOTA systems in Table IV. Because the amount of data augmentation in [35] was about six times the amount used in this study, ECAPA-TDNN (Re-implemented) performed worse than ECAPA-TDNN (C=1024) [35]. Kwon et al. [60] revealed that data augmentation helps to significantly improve performance and reduce overfitting. But we can see that ECAPA-TDNN-ResPEL outperformed ECAPA-TDNN (C=1024) in most cases (e.g., EER: 2.03% vs 2.12% and minDCF: 0.1895% vs 0.2101% on the VoxCeleb1-H (cleaned) set), and both ECAPA-TDNN-PEL and ECAPA-TDNN-ResPEL always outperformed ResNet systems. Therefore, the experimental results showed the robust speaker representation ability of the PEL-based systems. Furthermore, ECAPA-TDNN-ResPEL performed slightly worse than the MFA-TDNN (Standard) [61] and H / ASP [60] systems on the VoxCeleb1-E (cleaned) trial list, but achieved the best performance on the VoxCeleb1-H (cleaned) trial list. In addition, ECAPA-TDNN-Res2PEL achieved similar performance to ECAPA-TDNN-ResPEL. Although ECAPA-TDNN-Res2PEL did not further improve SV performance over ECAPA-TDNN-ResPEL, it still achieved a significant performance improvement compared to conventional ECAPA-TDNN (Re-implemented). It is worth noting that the proposed Parent Embedding learning does not change the backbone structure and can be effortlessly combined with different embedding-based model architectures (e.g., E-TDNN, ECAPA-TDNN, and MFA-TDNN). On the other hand, an encouraging result was that Parent Embedding and Child Embedding of ECAPA-TDNN-ResPEL had similar performance, while Child Embedding was worse than Parent Embedding for ECAPA-TDNN-PEL. The results confirmed that adding a residual connection between Parent Embedding and Child Embedding can learn closely-related and more robust representations. Finally, we further analyzed the SV performance of the concatenated Parent Embedding $\hbar_1$ and Child Embedding $\hbar_2$. The results show that the concatenated embedding (cf. *concat* in Table IV) did not significantly improve the SV performance.

G. PAS-Norm for Anti-Interference

In previous experiments, although the proposed PEL-based embedding systems can achieve superior SV performance compared with the baseline systems, the performance dropped significantly when the testing data contained interference. In view of this, the proposed PAS-Norm method was dedicated to dealing with the influence of interference on SV. Table V shows the results of SV experiments using PAS-Norm with different partial score aggregations. As described in Section III-C, the largest similarity score of two candidate embeddings was selected to measure the normalization score. We compared the performance of PAS-Norm based on maximum, average, and minimum. The average-based PAS-Norm is similar to the AS-Norm over test time augmentation (TTA) [43]. From Table V, we can see that the maximum-based PAS-Norm performed

TABLE V
EERs (%) OF SPEAKER VERIFICATION EXPERIMENTS ON THE TRIAL LISTS OF VOXCELEB1, SITW AND CN-CELEB DATASETS WITH DIFFERENT PARTIAL SCORE AGGREGATIONS OF PAS-NORM USING PARENT EMBEDDING OF ECAPA-TDNN-RESPEL

Norm. Type	Aggr. Type	(w/o interference)			(w/ interference from other speakers)			(multi-genre)
		VoxCeleb1-E (cleaned)	VoxCeleb1-H (cleaned)	SITW core-core	SITW core-multi	SITW assist-core	SITW assist-multi	CN-Celeb
Original	-	1.31	2.37	1.74	3.27	3.39	4.29	13.91
AS-Norm	-	**1.17**	**2.03**	1.56	3.02	3.16	4.02	12.80
PAS-Norm	Average (similar to AS-Norm + TTA)	1.30	2.26	1.66	3.24	3.25	4.12	13.37
	Maximum	1.31	2.26	**1.48**	**2.51**	**2.49**	**3.04**	**12.58**
	Minimum	2.31	3.83	3.88	7.29	6.79	8.54	19.24

TABLE VI
EERs (%) OF SPEAKER VERIFICATION EXPERIMENTS ON THE TRIAL LISTS OF VOXCELEB1, SITW AND CN-CELEB DATASETS WITH DIFFERENT SCORE NORMALIZATION METHODS

System / Embedding	Norm. Type	(w/o interference)			(w/ interference from other speakers)			(multi-genre)
		VoxCeleb1-E (cleaned)	VoxCeleb1-H (cleaned)	SITW core-core	SITW core-multi	SITW assist-core	SITW assist-multi	CN-Celeb
ECAPA-TDNN-PEL / Parent Embedding $\hbar_1$	Original	1.38	2.43	1.83	3.16	3.20	4.09	13.12
	Z-Norm	1.35	2.39	1.63	3.02	3.35	4.10	13.12
	T-Norm	1.35	2.40	1.68	3.14	3.16	4.01	12.46
	S-Norm	1.31	2.32	**1.55**	3.04	3.18	3.98	12.68
	AZ-Norm	1.28	2.22	1.68	2.97	3.10	4.02	13.04
	AT-Norm	1.29	2.21	1.64	2.97	3.09	3.92	11.72
	AS-Norm	**1.25**	**2.14**	1.60	2.96	3.12	3.91	12.22
	PZ-Norm	1.55	2.74	1.67	2.58	2.85	3.37	13.18
	PT-Norm	1.58	2.75	1.76	2.71	2.71	3.29	12.36
	PS-Norm	1.50	2.63	1.59	2.54	2.69	3.27	12.61
	PAZ-Norm	1.44	2.46	1.76	2.58	**2.52**	3.22	13.08
	PAT-Norm	1.48	2.50	1.86	2.65	2.68	3.27	**11.34**
	PAS-Norm	1.39	2.36	1.65	**2.51**	2.55	**3.12**	12.04
ECAPA-TDNN-ResPEL / Parent Embedding $\hbar_1$	Original	1.31	2.37	1.74	3.27	3.39	4.29	13.91
	Z-Norm	1.27	2.31	1.49	3.00	3.46	4.28	13.65
	T-Norm	1.27	2.30	1.57	3.24	3.18	4.05	13.12
	S-Norm	1.23	2.21	1.45	3.08	3.27	4.14	13.31
	AZ-Norm	1.20	2.11	1.48	2.98	3.24	4.15	13.50
	AT-Norm	1.20	2.11	1.66	3.08	3.10	3.99	12.46
	AS-Norm	**1.17**	**2.03**	1.56	3.02	3.16	4.02	12.80
	PZ-Norm	1.48	2.66	1.53	**2.49**	2.83	3.32	13.75
	PT-Norm	1.49	2.66	1.75	2.74	2.55	3.22	13.16
	PS-Norm	1.41	2.52	**1.43**	2.53	2.65	3.22	13.25
	PAZ-Norm	1.35	2.35	1.53	2.51	2.64	3.25	13.48
	PAT-Norm	1.38	2.42	1.81	2.74	2.51	3.08	**12.13**
	PAS-Norm	1.31	2.26	1.48	2.51	**2.49**	**3.04**	12.58
ECAPA-TDNN-Res2PEL / Parent Embedding $\hbar_1$	Original	1.31	2.30	1.64	3.03	3.37	4.19	13.59
	Z-Norm	1.29	2.29	1.56	2.94	3.30	4.05	13.53
	T-Norm	1.28	2.28	1.61	3.01	3.36	4.04	12.98
	S-Norm	1.25	2.22	**1.52**	2.86	3.30	4.03	13.12
	AZ-Norm	1.21	2.11	1.53	2.89	3.18	4.03	13.44
	AT-Norm	1.20	2.12	1.60	2.87	3.23	3.97	12.11
	AS-Norm	**1.17**	**2.05**	1.54	2.80	3.21	4.03	12.60
	PZ-Norm	1.48	2.64	1.59	2.46	2.77	3.28	13.53
	PT-Norm	1.51	2.67	1.72	2.62	2.67	3.24	12.77
	PS-Norm	1.44	2.53	1.57	2.44	2.64	3.24	13.01
	PAZ-Norm	1.37	2.35	1.57	**2.40**	2.67	3.28	13.54
	PAT-Norm	1.39	2.40	1.77	2.65	2.64	**3.09**	**11.66**
	PAS-Norm	1.33	2.27	1.54	2.43	**2.58**	3.11	12.25

worse than AS-Norm when evaluated on two VoxCeleb1 trial lists consisting of pairs without interference, but achieved significant performance gains on other trial lists, especially those with pairs interfered by other speakers. The average-based PAS-Norm also had some effect, but performed worse than the AS-Norm, while the minimum-based PAS-Norm performed even worse than the baseline without normalization. The results show that the proposed maximum-based PAS-Norm with partial score aggregation can provide robust speaker discrimination and effectively improve SV under interference conditions and different genres.

Table VI shows the results of SV experiments on the VoxCeleb1, SITW and CN-Celeb trial lists with different score normalization methods. We compared our maximum-based

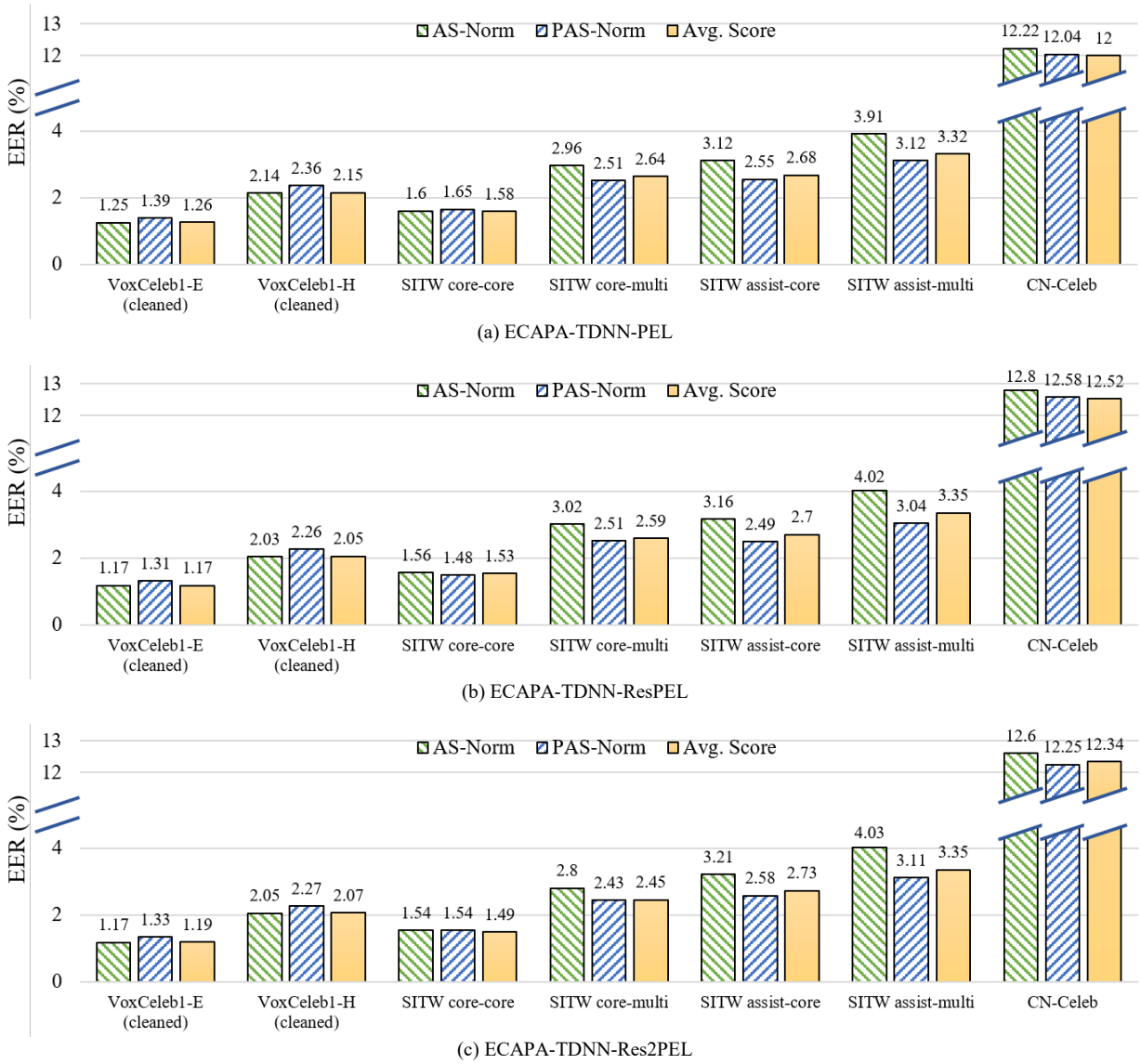


Fig. 6. EERs of average score fusion of AS-Norm and PAS-Norm evaluated with PEL-based Parent Embedding.

PAS-Norm method with several classical score normalization methods [54][62]. We also applied partial score aggregation to these normalization methods, yielding PZ-Norm, PT-Norm, PS-Norm, PAZ-Norm, and PAT-Norm. From Table VI, we can see that across all 12 test cases with interference or different genres, the best performing normalization method is always the normalization method with partial score aggregation. We can also see that PAS-Norm outperformed other score normalization methods in 5 out of 9 cases with interference, including ECAPA-TDNN-PEL Parent Embedding on the SITW *EVAL* core-multi (EER=2.51%) and SITW *EVAL* assist-multi (EER=3.12%) trial lists, ECAPA-TDNN-ResPEL Parent Embedding on the SITW *EVAL* assist-core (EER=2.49%) and SITW *EVAL* assist-multi (EER=3.04%) trial lists, and ECAPA-TDNN-Res2PEL Parent Embedding on the SITW *EVAL* assist-core (EER=2.58%) trial list. In the rest of the cases with interference, PAS-Norm performed the second best in 4 cases, including ECAPA-TDNN-PEL Parent Embedding on the SITW *EVAL* assist-core (EER=2.55%) trial list, ECAPA-TDNN-ResPEL Parent Embedding on the SITW *EVAL* core-multi (EER=2.51%) trial list, and ECAPA-TDNN-Res2PEL Parent Embedding on the SITW *EVAL* core-multi (EER=2.43%) and SITW *EVAL* assist-multi (EER=3.11%) trial lists. On the other hand, we can see from Table VI that AS-Norm performed the best in 6 out of 9 test cases without interference, while S-Norm performed the best on the SITW *EVAL* core-core trial list. We can also see that PAS-Norm performed only slightly worse than S-Norm on the SITW *EVAL* core-core trial list (1.65% vs 1.55%, 1.48% vs 1.45%, and 1.54% vs 1.52%). Obviously, the performance degradation caused by PAS-Norm on this dataset is relatively smaller than the performance gain provided by PAS-Norm on other datasets. Taken together, the results in Table VI confirm both partial score aggregation and PAS-Norm are effective in dealing with interference.

H. Integration of AS-Norm and PAS-Norm

As shown in the above section, PAS-Norm based on partial score aggregation outperformed AS-Norm on all trial lists with interference, but performed worse than AS-Norm on two VoxCeleb1 trial lists without interference. The major reason is that candidate embedding pairs corresponding to partial signals may have increased similarity scores due to the comparison of unvoiced segments, even if the two speech signals are from different speakers. In this experiment, we integrated AS-Norm and PAS-Norm by average score fusion to achieve stable results. The results of SV are shown in Fig. 6. Evaluated on two VoxCeleb1 trial lists, average score fusion effectively reduced the instability of PAS-Norm and achieved similar performance to AS-Norm. Evaluated on the SITW trial lists, average score fusion performed slightly worse than PAS-Norm, but much better than AS-Norm. Evaluated on the CN-Celeb trial list, average score fusion was superior to AS-Norm and PAS-Norm. In summary, the average score fusion of AS-Norm and PAS-Norm achieved stable results, demonstrating robust comparison capabilities in SV under recording conditions without or with interference (such as interference from other speakers or different genres).

V. CONCLUSIONS

A robust speaker embedding model must have good generalization ability and anti-interference ability to be applied to complex real-world environments. In this paper, we have proposed a parent embedding learning (PEL) method for training the speaker embedding extractor, so that the extracted speaker embeddings have better generalization ability when applied to down-stream SV tasks. The main idea is to connect each embedding layer to an individual speaker classifier when training the speaker embedding extractor to learn discriminative representations. Furthermore, we have proposed a partial adaptive score normalization (PAS-Norm) method to reduce the influence of interference on speaker embedding similarity comparisons. The results of experiments on the VoxCeleb1 dataset show that the PEL-based embeddings can achieve better SV performance than the SOTA ECAPA-TDNN x-vector embeddings. The results of experiments on the SITW and CN-Celeb datasets show that PAS-Norm outperformed AS-Norm and other classical score normalization methods in the presence of interference from other speakers and different recording situations. From the experimental results, the proposed PEL and PAS-Norm have the potential to improve the applicability of SV in complex real-world environments.

In our future work, several issues will be considered to generate reliable speaker embeddings, including emotional speech, mixed speech, far-field scenarios, and various noisy environments.

REFERENCES

[1] I. Bisio, C. Garibotto, A. Grattarola, F. Lavagetto, and A. Sciarrone, "Smart and robust speaker recognition for context-aware in-vehicle applications," *IEEE Trans. Vehicular Technol.*, vol. 67, no. 9, pp. 8808–8821, Sep. 2018.

[2] S. R. Avutu, D. Bhatia, and B. V. Reddy, "Voice control module for low cost local-map navigation based intelligent wheelchair," in *Proc. IEEE Int. Adv. Comput. Conf.*, Hyderabad, Telangana, India, 2017, pp. 609–613.

[3] Z. Kozhirbayev, B. A. Erol, A. Sharipbay, and M. Jamshidi, "Speaker recognition for robotic control via an IoT device," in *Proc. World Autom. Cong.*, Stevenson, WA, USA, 2018, pp. 1–5.

[4] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 4, pp. 788–798, May 2011.

[5] A. Kanagasundaram, R. Vogt, D. Dean, S. Sridharan, and M. Mason, "I-vector based speaker recognition on short utterances," in *Proc. Interspeech*, Florence, FI, Italy, 2011, pp. 2341–2344.

[6] S. Cumani, and P. Laface, "Speaker recognition using e–vectors," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 4, pp. 736–748, Apr. 2018.

[7] J. Villalba, N. Brümmer, and N. Dehak, "Tied variational autoencoder backends for i-vector speaker recognition," in *Proc. Interspeech*, Stockholm, AB, Sweden, 2017, pp. 1004–1008.

[8] T. Kinnunen, L. Juvela, P. Alku, and J. Yamagishi, "Non-parallel voice conversion using i-vector PLDA: Towards unifying speaker verification and transformation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, New Orleans, LA, USA, 2017, pp. 5535–5539.

[9] J. Chang, and D. Wang, "Robust speaker recognition based on DNN/i-vectors and speech separation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, New Orleans, LA, USA, 2017, pp. 5415–5419.

[10] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Florence, FI, Italy, 2014, pp. 4052–4056.

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

[12] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with LSTM," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 5239–5243.

[13] G. Bhattacharya, J. Alam, and P. Kenny, "Deep speaker embeddings for short-duration speaker verification," in *Proc. Interspeech*, Stockholm, AB, Sweden, 2017, pp. 1517–1521.

[14] S. Wang, Z. Huang, Y. Qian, and K. Yu, "Discriminative neural embedding learning for short-duration text-independent speaker verification," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 11, pp. 1686–1696, Nov. 2019.

[15] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 5329–5333.

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

[17] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K.J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 37, no. 3, pp. 328–339, Mar. 1989.

[18] D. Snyder, D. Garcia-Romero, and D. Povey, "Time delay deep neural network-based universal background models for speaker recognition," in *IEEE Workshop Autom. Speech Recognit. Understand.*, Scottsdale, AZ, USA, 2015, pp. 92–97.

[19] Q. B. Hong, C. H. Wu, H. M. Wang, and C. L. Huang, "Statistics pooling time delay neural network based on x-vector for speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Barcelona, Spain, 2020, pp. 6849–6853.

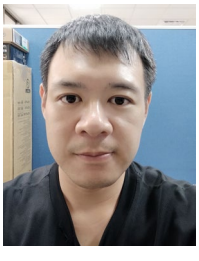
[20] Y. Tang, G. Ding, J. Huang, X. He, and B. Zhou, "Deep speaker embedding learning with multi-level pooling for text-independent speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Brighton, UK, 2019, pp. 6116–6120.

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

[22] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Honolulu, Hi, USA, 2017, pp. 212–220.

[23] Y. Li, F. Gao, Z. Ou, and J. Sun, "Angular softmax loss for end-to-end speaker verification," in *Proc. Int. Symp. Chin. Spoken Lang. Process.*, Taipei, Taiwan, 2018, pp. 190–194.

[24] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Process. Lett.*, vol. 25, no. 7, pp. 926–930, Jul. 2018.
[25] S. M. Kye, Y. Jung, H. B. Lee, S. J. Hwang, and H. Kim, "Meta-learning for short utterance speaker recognition with imbalance length pairs," in *Proc. Interspeech*, Shanghai, China, 2020, pp. 2982–2986.
[26] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 4879–4883.
[27] J. H. Kim, H. J. Shim, J. W. Jung, and H. J. Yu, "Learning metrics from mean teacher: A supervised learning method for improving the generalization of speaker verification system," *arXiv preprint arXiv:2104.06604*, 2021.
[28] R. Caruana, "Multitask learning" *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.
[29] Y. Zhang, and Q. Yang, "An overview of multi-task learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 30–43, 2018.
[30] Y. Zhang, and Q. Yang, "A survey on multi-task learning," *IEEE Trans. Knowl. Data Eng.*, Mar. 2021.
[31] E. Meyerson, and R. Miikkulainen, "Pseudo-task augmentation: From deep multitask learning to intratask sharing—and back," in *Int. Conf. Mach. Learn.*, Stockholm, AB, Sweden, 2018, pp. 3511–3520.
[32] H. Zeinali, S. Wang, A. Silnova, P. Matejka, and O. Plchot, "BUT system description to VoxCeleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.
[33] D. Garcia-Romero, A. McCree, D. Snyder, and G. Sell, "JHU-HLTCOE system for the VoxSRC speaker recognition challenge," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Barcelona, Spain, 2020, pp. 7559–7563.
[34] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Brighton, UK, 2019, pp. 5796–5800.
[35] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, Shanghai, China, 2020, pp. 3830–3834.
[36] S. H. Gao, M. M. Cheng,K. Zhao, X. Y. Zhang,M. H. Yang, and P. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Aug 2019.
[37] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 7132–7141.
[38] J. Thienpondt, B. Desplanques, and K. Demuynck, "The IDLAB VoxCeleb speaker recognition challenge 2021 system description," *arXiv preprint arXiv:2109.04070*, 2021.
[39] S. Cumani, P. D. Batzu, D. Colibro, C. Vair, P. Laface, and V. Vasilakakis, "Comparison of speaker recognition approaches for real applications," in *Proc. Interspeech*, Florence, FI, Italy, 2011, pp. 2365–2368.
[40] S. J. Prince, and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. IEEE Int. Conf. Comput. Vision*, Rio de Janeiro, Brazil, 2007, pp. 1–8.
[41] D. Garcia-Romero, and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. Interspeech*, Florence, FI, Italy, 2011, pp. 249–252.
[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Paradise, NV, USA, 2016, pp. 770–778.
[43] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. Interspeech*, Hyderabad, India, 2018.
[44] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," in *Proc. Interspeech*, Stockholm, AB, Sweden, 2017.
[45] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, South Brisbane, QLD, Australia, 2015, pp. 5206–5210.
[46] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The speakers in the wild (SITW) speaker recognition database," in *Proc. Interspeech*, San Francisco, CA, USA, 2016, pp. 818–822.
[47] Y. Fan, J. W. Kang, L. T. Li, K. C. Li, H. L. Chen, S. T. Cheng, P. Y. Zhang, Z. Y. Zhou, Y. Q. Cai, and D. Wang, "CN-Celeb: A challenging chinese speaker recognition dataset," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Barcelona, Spain, 2020, pp. 7604–7608.
[48] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *IEEE Workshop Autom. Speech Recognit. Understand.*, Big Island, HI, USA, 2011.
[49] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.
[50] 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. Int. Conf. Acoust., Speech, Signal Process.*, New Orleans, LA, USA, 2017, pp. 5220–5224.
[51] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Comput. Speech Lang.*, vol. 60, 2020, Art. no. 101027.
[52] L. N. Smith, "Cyclical learning rates for training neural networks," in *IEEE Winter Conf. App. Comput. Vision*, Santa Rosa, CA, USA, 2017, pp. 464–472.
[53] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Int. Conf. Mach. Learn.*, Atlanta, GA, USA, 2013.
[54] P. Matejka, O. Novotny, O. Plchot, L. Burget, M. D. Sanchez, and J. "H." Cernocky, "Analysis of score normalization in multilingual speaker recognition," in *Proc. Interspeech*, Stockholm, AB, Sweden, 2017, pp. 1567–1571.
[55] A. kr. Sarkar and Z.-H. Tan, "On training targets and activation functions for deep representation learning in text-dependent speaker verification," *arXiv preprint arXiv: 2201.06426*, 2022.
[56] A. K. Sarkar, Z.-H. Tan, H. Tang, S. Shon, and J. Glass, "Time-contrastive learning based deep bottleneck features for text-dependent speaker verification," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1267–1279, Aug. 2019.
[57] T. Zhou, Y. Zhao, and J. Wu, "Resnext and res2net structures for speaker verification," in *Proc. IEEE Spoken Lang. Technol. Workshop*, Shenzhen, China, 2021, pp. 301–307.
[58] Y.-J. Zhang, Y.-W. Wang, C.-P. Chen, C.-L. Lu, and B.-C. Chan, "Improving time delay neural network based speaker recognition with convolutional block and feature aggregation methods," in *Proc. Interspeech*, Brno, Czechia, 2021, pp. 76–80.
[59] M. MohammadAmini, D. Matrouf, J.-F. Bonastre, S. Dowerah, R. Serizel, and D. Jouvet, "Learning noise-robust ResNet-based speaker embedding for speaker recognition," in *Odyssey Speak. Lang. Recognit. Workshop*, Beijing, China, 2022.
[60] Y. Kwon, H.-S. Heo, B.-J. Lee, and J. S. Chung, "The ins and outs of speaker recognition: lessons from VoxSRC 2020," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Toronto, ON, Canada, 2021, pp. 5809–5813.
[61] T. Liu, R. K. Das, K. A. Lee, and H. Li, "MFA: TDNN with multi-scale frequency-channel attention for text-independent speaker verification with short utterances," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Singapore, 2022, pp. 7517–7521.
[62] D. E. Sturim and D. A. Reynolds, "Speaker adaptive cohort selection for tnorm in text-independent speaker verification," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Philadelphia, PA, USA, 2005, pp. I/741–I/744.
[63] R. Zhang, J. Wei, W. Lu, L. Wang, M. Liu, L. Zhang, J. Jin, and J. Xu, "ARET: Aggregated residual extended time-delay neural networks for speaker verification," in *Proc. Interspeech*, Shanghai, China, 2020, pp. 946–950.



Qian-Bei Hong received the B.S. and M.S. degrees in electrical engineering from Southern Taiwan University of Science and Technology, Tainan, Taiwan, in 2009 and 2011, respectively. He is currently pursuing the Ph.D. degree in the Graduate Program of Multimedia Systems and Intelligent Computing, National Cheng Kung University and Academia Sinica, Tainan, Taiwan. His research interests include multimedia signal processing, deep learning, and speaker recognition.



Chung-Hsien Wu received the B.S. degree in electronics engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1981, and the M.S. and Ph.D. degrees in electrical engineering from National Cheng Kung University (NCKU), Tainan, Taiwan, in 1987 and 1991, respectively. Since 1991, he has been with the Department of Computer Science and Information Engineering, NCKU. He became the Chair Professor in 2017. He served as the Deputy Dean of the College of Electrical Engineering and Computer Science, NCKU, from 2009 to 2015. He also worked at Computer Science and Artificial Intelligence Laboratory of Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in summer 2003, as a Visiting Scientist. He was the Associate Editor of *IEEE Transactions on Audio, Speech and Language Processing* (2010–2014), *IEEE Transactions on Affective Computing* (2010–2014), *ACM Transactions on Asian and Low-Resource Language Information Processing*, and *APSIPA Transactions on Signal and Information Processing* (2014~2020). He was the APSIPA BoG Member in 2019~2021. He received 2018 APSIPA Sadaoki Furui Prize Paper Award in 2018, and the Outstanding Research Award of Ministry of Science and Technology, Taiwan, in 2010 and 2016. His research interests include deep learning, affective computing, speech recognition/synthesis, and spoken language processing.



Hsin-Min Wang (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1989 and 1995, respectively. In October 1995, he joined the Institute of Information Science, Academia Sinica, Taipei, Taiwan, where he is currently a Research Fellow. He also holds a joint appointment as a Professor in the Department of Computer Science and Information Engineering at National Cheng Kung University. He was an Associate Editor of *IEEE/ACM Transactions on Audio, Speech and Language Processing* from 2016 to 2020. He currently serves a Senior Editor of *APSIPA Transactions on Signal and Information Processing*. His major research interests include spoken language processing, natural language processing, multimedia information retrieval, machine learning and pattern recognition. He was a General Co-Chair of ISCSLP2016 and ISCSLP2018 and a Technical Co-Chair of ISCSLP2010, O-COCOSDA2011, APSIPAASC2013, ISMIR2014, and ASRU2019. He received the Chinese Institute of Engineers Technical Paper Award in 1995 and the ACM Multimedia Grand Challenge First Prize in 2012. He was an APSIPA distinguished lecturer for 2014–2015. He is a member of the International Speech Communication Association and ACM.