ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10096040

# HYBRID NEURAL NETWORK WITH CROSS- AND SELF-MODULE ATTENTION POOLING FOR TEXT-INDEPENDENT SPEAKER VERIFICATION

*Jahangir Alam, Woo Hyun Kang, Abderrahim Fathan*

Computer Research Institute of Montreal (CRIM)

jahangir.alam,woohyun.kang,abderrahim.fathan@crim.ca

## ABSTRACT

Extraction of a speaker embedding vector plays an important role in deep learning-based speaker verification. In this contribution, to extract speaker discriminant utterance level embeddings, we propose a hybrid neural network that employs both cross- and self-module attention pooling mechanisms. More specifically, the proposed system incorporates a 2D-Convolution Neural Network (CNN)-based feature extraction module in cascade with a frame-level network, which is composed of a fully Time Delay Neural Network (TDNN) network and a TDNN-Long Short Term Memory (TDNN-LSTM) hybrid network in a parallel manner. The proposed system also employs a multi-level cross- and self-module attention pooling for aggregating the speaker information within an utterance-level context by capturing the complementarity between two parallelly connected modules. In order to evaluate the proposed system, we conduct a set of experiments on the Voxceleb corpus, and the proposed hybrid network is able to outperform the conventional approaches trained on the same dataset.

***Index Terms***— Speaker verification, speaker embeddings, hybrid neural network, cross-module attention

## 1. INTRODUCTION

Speaker verification is the task of verifying whether the person is who he/she claims to be. It has become a key technology for personnel authentication in various applications. Usually, an utterance-level fixed-dimensional vector (i.e., embedding vector) is extracted from the enrolment and test speech recordings and then fed into a backend classifier (e.g., cosine similarity) to measure their similarity.

Since the introduction of d-vector [1], deep learning architectures have been actively employed as embedding extractors in order to efficiently capture the speaker-dependent information from the given speech signal. Subsequently, the x-vector framework [2], which uses a time-delay neural network (TDNN) architecture and statistics pooling for extracting a speaker embedding, became the most popular approach due to its superior performance in text-independent speaker verification [3] task. Several extensions [4, 5] of the x-vector framework were also proposed for improving speaker verification performance.

For the past several years, there have been lots of attempts on employing the residual network (ResNet) architecture for speaker embedding extraction [6, 7], which have proven to be the dominant approach in the image classification field [8]. Moreover, to exploit the speaker-dependent information within the temporal variability of the speech sequence, many researches also focused on employing a recurrent neural network such as the Long Short-Term Memory (LSTM) network for speaker embedding extraction [9, 10].

Although the embedding extractors mentioned above have shown reasonable performance in terms of speaker verification, most of them rely on a single deep learning architecture (e.g., TDNN, ResNet, LSTM). However, different network architectures are known to learn complementary information about the input representation such as TDNN and convolutional neural network (CNN) are adept at reducing changes in frequency, while the LSTMs are good for sequential & temporal modeling and DNN is suitable for mapping input features to a more separable regions.

In an attempt to make use of the complementary speaker information encoded by different neural architectures, various hybrid approaches have been proposed, which employs non-TDNN modules such as LSTM or CNN to the x-vector framework, and have demonstrated noticable gain in the speaker verification performance. It was observed in [11, 12, 13] that the robustness of x-vectors can be boosted by appending residual connections between the frame-level layers. Moreover, a multi-level pooling scheme was introduced in [14] to take into account the statistics from different modules (e.g., TDNN, LSTM), and promising performance in speaker verification was reported. In view of this, a hybrid neural network (HNN) was recently proposed for robust speaker embedding extraction [15, 16], which not only employs different types of network architectures (i.e., 2D-CNN, TDNN, LSTM) but also exploits the short-durational statistics of the hidden representations to capture the instantaneous speaker-dependent information.

In this work, we further expand the hybrid approach to fully exploit the complementary speaker information learned by the different network modules. More specifically, we propose a new HNN architecture with Cross- and Self-module Attention pooling mechanisms and we denote the proposed system as CSA-HNN. Like the HNN approach [15, 16], the CSA-HNN also considers both global and local statistics to take into account speaker information within the long and short temporal context. In order to evaluate the speaker verification performance of our proposed system, we have conducted a set of experiments on the Voxceleb [17, 18] dataset.

Our contribution in this work can be summarized as:

- We proposed a new HNN architecture that incorporates a 2D-CNN - based feature extraction module in cascade with a frame-level network, which is formed by connecting a fully TDNN and a TDNN-LSTM hybrid modules parallely. The proposed system also employs a multi-level cross-module attention pooling for aggregating the speaker information within an utterance-level context by capturing the complementarity between two parallelly connected modules. This system is denoted as CA-HNN (Cross-module Attention pooling - based HNN).

- By introducing a small modification to the CA-HNN system, we also propose another HNN architecture that employs both cross- and self-module attention pooling mechanisms. We denote this approach as Cross- & Self-module Attention - based HNN (CSA-HNN).

- We perform an investigative study on the influence of cross-module attention for speaker verification.

## 2. HYBRID NEURAL NETWORK WITH CROSS- AND SELF-MODULE ATTENTION POOLING

In this section, we provide a description of the proposed embedding extraction approaches for robust speaker verification. To further exploit the complementarity between different network modules, we introduce a new Hybrid Neural Network (HNN) architecture based on cross-module attention pooling. We denote this system as CA-HNN and is presented in Figure 1. There are two main differences between the standard HNN [15, 16], and the proposed CA-HNN systems and they are: (i) unlike the standard HNN, where the different modules of the frame-level network were connected in cascade, in the proposed CA-HNN they are connected in a parallel manner, (ii) in contrast to HNN, which employs a multi-level global-local statistics pooling, the proposed approach utilizes a multi-level cross-module attention statistics pooling.

By introducing a small modification to the CA-HNN system, we also propose another HNN architecture that employs not only cross-module attention but also self-module attention in order to make best use of both pooling mechanisms. We denote this approach as Cross- & Self-module Attention HNN (CSA-HNN) and is depicted in Figure 2.

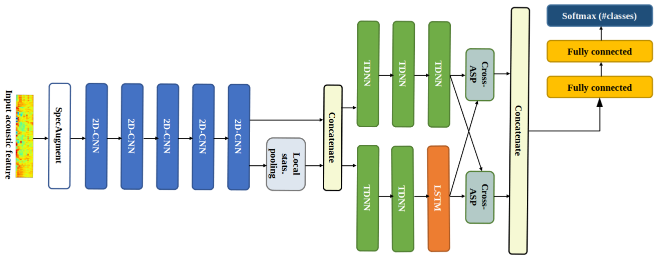

**Fig. 1**. The general architecture of the proposed Cross-module Attention pooling - based Hybrid Neural Network (CA-HNN) for speaker verification.

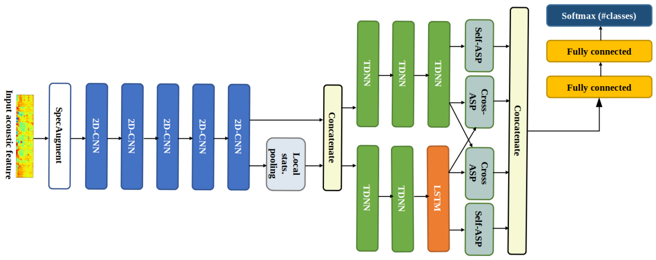

**Fig. 2**. The general architecture of the proposed Cross- & Self-module Attention pooling - based Hybrid Neural Network (CSA-HNN) for speaker verification.

### 2.1. 2D-CNN-based feature extraction module

Similar to the standard HNN, in order to make sure that the hybrid network can capture the temporal-spectral correlations within the speech, the CA-HNN (and CSA-HNN) uses 2D-CNNs to process the input Mel-FilterBank (MFB) features over which SpecAugment [19] is applied on the fly, where both time and frequency masking are performed. By passing the input augmented MFB features (after applying SpecAugment) through a stack of 5 2D-CNN layers, frame-level representations with information on not only the relation between the local frames, but also the local frequency bins could be obtained.

### 2.2. Frame-level network

In the standard HNN framework [15, 16], the frame-level network was made up of TDNN and LSTM layers connected in cascade for the extraction of local descriptors with sufficient temporal information for speaker discrimination. The frame-level network used in the HNN is similar to the TDNN-LSTM approach presented in [20], where the second TDNN layer of the standard x-vector [21] configuration is replaced with a LSTM layer. On the contrary, in the proposed CA-HNN or CSA-HNN, the frame-level network is comprised of parallelly connected a fully TDNN network (with 3 TDNN layers) and a TDNN-LSTM hybrid network (with 2 TDNN layers and 1 LSTM layer) for the extraction of speaker discriminant local descriptors with sufficient temporal information. As shown in Figures 1 & 2, the output representation of the 2D-CNN module is fed into both a fully TDNN network and a TDNN-LSTM hybrid network. The frame-level output from each network is then aggregated using cross-module attention statistics pooling, where the attention weights are computed from the other module output. The aggregated representations from both networks are then concatenated and then fed into a fully connected layer.

### 2.3. Cross- & self-module attention pooling

Given frame-level representations $\{\mathbf{h}_1, ..., \mathbf{h}_T\}$, the formulation of the standard Attention Statistics Pooling (ASP), also known as self-module attention, is defined as follows:

$$\omega = \sum_{t=1}^{T} \alpha_t \mathbf{h}_t \tag{1}$$

where $\alpha_t \in [0, 1]$ is a normalized weight, which is computed by

$$\alpha_t = \frac{\exp(e_t)}{\sum_{t=1}^{T} \exp(e_t)}, \tag{2}$$

$$e_t = \mathbf{v}_t \tanh(\mathbf{W}_t \mathbf{h}_t + \mathbf{b}_t) \tag{3}$$

where $\mathbf{v}_t$, $\mathbf{W}_t$, and $\mathbf{b}_t$ are trainable parameters and superscript $t$ indicates transpose operation. By using different weight for each frame, speech frames with relatively higher speaker-relevancy can contribute more to the embedding vector. As the name implies, in self-module attention pooling the frame-level output from each network is aggregated via attention statistics pooling (ASP) by computing the attention weight from the same module output.

In the proposed CA-HNN, we use a variant of the ASP called cross-module ASP (CASP) for incorporating the information from different modules. Given the outputs from the TDNN network $\{\mathbf{h}_1^{TDNN}, ..., \mathbf{h}_T^{TDNN}\}$ and TDNN-LSTM $\{\mathbf{h}_1^{TDNN-LSTM}, ..., \mathbf{h}_T^{TDNN-LSTM}\}$, the proposed CASP, as presented in Figure 1, is computed as follows:

$$\omega_{TDNN} = \sum_{t=1}^{T} \alpha_t^{TDNN-LSTM} \mathbf{h}_t^{TDNN}, \tag{4}$$

$$\alpha_t^{TDNN-LSTM} = \frac{\exp(e_t^{TDNN-LSTM})}{\sum_{t=1}^{T} \exp(e_t^{TDNN-LSTM})}, \tag{5}$$

$$\begin{aligned} &e_t^{TDNN-LSTM} \\ &= \mathbf{a}_t \tanh(\mathbf{W}_t^{TDNN-LSTM} \mathbf{h}_t^{TDNN-LSTM} + \mathbf{b}_t), \end{aligned} \tag{6}$$

$$\omega_{TDNN-LSTM} = \sum_{t=1}^{T} \alpha_t^{TDNN} \mathbf{h}_t^{TDNN-LSTM}, \tag{7}$$

$$\alpha_t^{TDNN} = \frac{\exp(e_t^{TDNN})}{\sum_{t=1}^{T} \exp(e_t^{TDNN})}, \tag{8}$$

$$e_t^{TDNN} = \mathbf{c}_t \tanh(\mathbf{W}_t^{TDNN} \mathbf{h}_t^{TDNN} + \mathbf{d}_t), \tag{9}$$

where $\mathbf{a}_t$, $\mathbf{b}_t$, $\mathbf{c}_t$, $\mathbf{d}_t$, $\mathbf{W}_t^{TDNN}$, and $\mathbf{W}_t^{TDNN-LSTM}$ are trainable parameters and superscript $t$ indicates transpose operation, and $\omega_{TDNN}, \omega_{TDNN-LSTM}$ are the embeddings extracted from the TDNN and TDNN-LSTM networks, respectively.

Since the TDNN and TDNN-LSTM modules process the frame-level data differently, they may yield representations with complementary information relevant to the speaker identity. Therefore providing the attention scores for each module computed from a different network can yield the embedding to have enhanced speaker specific information. For example, the TDNN is known to take the local frame-level correlation into account, while the LSTM is known to encode the sequential context. To obtain the utterance-level embedding $\omega_{TDNN}$ from the TDNN module, the CASP will aggregate the frame-level TDNN outputs using the attention scores derived from the TDNN-LSTM module. Thus the embedding $\omega_{TDNN}$ will not only have information about the local correlation between adjacant frames, but also the context in terms of sequential pattern.

As depicted in Figure 2, the proposed CSA-HNN paradigm utilizes both Cross-module and Self-module Attention Statistics Pooling (CSASP) techniques in order to take advantage from both pooling mechanisms.

## 3. EXPERIMENTS AND RESULTS

### 3.1. Experimental setups

VoxCeleb corpus [17, 18] is used to to evaluate the speaker verification performance of the proposed CA-HNN & CSA-HNN frameworks. As training data to train the embedding extractors, only the Voxceleb2 development set that contains only 5,994 speakers with 1,092,009 utterances is used [17]. Experimental results are reported on all three Voxceleb 1 test sets, namely, VoxCeleb1_O: original Voxceleb 1 test set, VoxCeleb1_E: extended Voxceleb 1 test set, and VoxCeleb1_H: Voxceleb 1 hard test set [18].

80-dimensional Mel-FilterBank (MFB) features are extracted using an analysis window size of 25 ms over a frame shift of 10 ms. The acoustic features are randomly augmented on-the-fly with either MUSAN noise, speed perturbation with rate between 0.95 1.05 or reverberation. In addition, we used SpecAugment for applying frequency and time masking on the MFB features. The embedding networks are trained with segments consisting of 180 frames, using the ADAM optimization technique [22]. The AAMSoftmax objective was used for training the proposed embedding networks. The networks were trained with an initial learning rate 0.0000001 with triangular cyclic scheduling. The batch size for training was set to be 64, which consists of 64 randomly selected speakers. Cosine similarity was used for computing the verification scores in the experiments. Local statistics pooling of

**Table 1**. Statistics of the three test sets of voxceleb 1 dataset in terms of numbers of speakers, recordings, total number of trials and target trials.

| Test sets | # Speakers | # Recordings | # Trials | # Target trials |
|---|---|---|---|---|
| VoxCeleb1_O | 40 | 4874 | 37720 | 18860 |
| VoxCeleb1_E | 1251 | 145375 | 581480 | 290743 |
| VoxCeleb1_H | 1190 | 138137 | 552536 | 276270 |

**Table 2**. Comparison of speaker verification performances of various pooling mechanisms with the new HNN architecture. Results are reported on the Voxceleb1_O test set in terms of EER metric.

| | Scoring | EER (%) |
|---|---|---|
| SP | Cosine | 1.75 |
| ASP | Cosine | 1.63 |
| CASP | Cosine | **1.38** |
| CSASP | Cosine | **1.32** |

**Table 3**. Speaker verification performance on the three test partitions of Voxceleb 1 corpus. Here, AMS stands for additive margin softmax.

| | Scoring | EER (%) |
|---|---|---|
| | | Voxceleb1_O |
| *Chung et al.* [17] | Cosine | 3.95 |
| *Xie et al.* [23] | Cosine | 3.22 |
| *Hajavi et al.* [24] | Cosine | 4.26 |
| *Xiang et al.* [25] | Cosine | 2.69 |
| *Monteiro et al.* [26] | Learned sim. | 2.51 |
| SpecAugment+TDNN [19] | PLDA | 2.59 |
| SpecAugment+TDNN+AMS [19] | Cosine | 1.96 |
| SpecAugment+ResNet34 [19] | PLDA | 1.68 |
| HNN [15, 16] | PLDA | 1.55 |
| CA-HNN (*Ours*) | Cosine | **1.38** |
| CSA-HNN (*Ours*) | Cosine | **1.32** |
| | | Voxceleb1_E |
| *Chung et al.* [17] | Cosine | 4.42 |
| *Xie et al.* [23] | Cosine | 3.13 |
| *Xiang et al.* [25] | Cosine | 2.76 |
| *Monteiro et al.* [26] | Learned sim. | 2.57 |
| SpecAugment+TDNN [19] | PLDA | 2.77 |
| SpecAugment+TDNN+AMS [19] | Cosine | 2.31 |
| SpecAugment+ResNet34 [19] | PLDA | 1.80 |
| HNN [15, 16] | PLDA | 1.75 |
| CA-HNN (*Ours*) | Cosine | **1.62** |
| CSA-HNN (*Ours*) | Cosine | **1.53** |
| | | Voxceleb1_H |
| *Chung et al.* [17] | Cosine | 7.33 |
| *Xie et al.* [23] | Cosine | 5.06 |
| *Xiang et al.* [25] | Cosine | 4.73 |
| *Monteiro et al.* [26] | Learned sim. | 4.73 |
| SpecAugment+TDNN [19] | PLDA | 4.83 |
| SpecAugment+TDNN+AMS [19] | Cosine | 4.02 |
| SpecAugment+ResNet34 [19] | PLDA | **3.08** |
| HNN [15, 16] | PLDA | 3.00 |
| CA-HNN (*Ours*) | Cosine | **2.86** |
| CSA-HNN (*Ours*) | Cosine | **2.79** |

the 2D-CNN module are computed in a 199 frames window over a shift of 9 frames, and the input frame is extracted on multiples of 3 frames. Results are reported in terms of equal error rate (EER) metric only on all three Voxceleb1 test sets presented in Table 1.

### 3.2. Experimental results and discussion

In our first set of experiment, we analyze effectiveness of various pooling mechanisms on the new HNN architecture described in Section 2. More specifically, we compare the performances of statistics pooling (SP), attention statistics pooling (ASP), and the proposed cross-module attention statistics pooling (CASP) and cross- & self-module attention statistics pooling (CSASP). As shown in Table 2, the new HNN system can benefit even from a simple attention mechanism, where ASP outperformed the standard SP with a relative improvement of 6.86% in terms of EER.

On the other hand, the proposed cross-module attention statistics pooling (CASP) scheme further improved the performance, which achieved a relative improvement of 15.34% compared to ASP in terms of EER. From this result, we can assume that the complementary speaker-relevant information within the attention weights driven from the TDNN and TDNN-LSTM modules may be beneficial to each other in terms of extracting a reliable speaker embedding vector. Furthermore, combining the standard ASP with the proposed CASP (CSASP) enhanced the performance to a greater extent, which outperformed the CASP with a relative improvement of 4.55% in terms of EER. The competitiveness of the proposed framework is demonstrated in Table 3, which shows the EER results of the proposed CASP and CSASP with the new HNN architecture (i.e., CA-HNN, CSA-HNN), along with the conventional systems on the VoxCeleb1_O, VoxCeleb1_E, and VoxCeleb1_H trial sets. As shown in the results, we can see that the proposed CA-HNN and CSA-HNN can outperform the baseline systems in all trial conditions. Impressively, the performance of the proposed frameworks were able to even surpass the conventional HNN system, where on the VoxCeleb1_O trial, the CSA-HNN achieved a relative improvement of 14.84% compared to the HNN in terms of EER. This further substantiates our idea that the complementary attributes latent in the attention weights generated from different network modules can allow the embeddings to have more speaker-relevant information.

## 4. CONCLUSION

In this paper, we proposed a new HNN for speaker embedding extraction, that employs both cross- and self-module attention pooling mechanisms for aggregating the speaker information within an utterance-level context by exploiting the complementarity between two parallelly connected networks (i.e., TDNN, TDNN-LSTM). The proposed framework was evaluated using the VoxCeleb dataset, and our results showed that the proposed cross-module attention pooling mechanism can be beneficial for the embedding extraction system.

In our future study, we will further expand the proposed cross-module attention pooling scheme to make use of the complementarity of more various network modules (e.g., ETDNN, ETDNN-LSTM). Moreover, we will explore the possibility of applying the cross-module attention pooling to different variants of HNN, such as multi-stream HNN or ensembled HNN.

## 5. REFERENCES

[1] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. of ICASSP*, 2014, pp. 4052–4056.

[2] Yunqi Cai, Lantian Li, Dong Wang, and Andrew Abel, "Deep speaker vector normalization with maximum gaussianality training," 2020.

[3] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. of ICASSP*. IEEE, 2018, pp. 5329–5333.

[4] David Snyder et al., "The JHU Speaker Recognition System for the VOiCES 2019 Challenge," in *Proc. Interspeech 2019*, 2019, pp. 2468–2472.

[5] Jesús Villalba et al., "State-of-the-Art Speaker Recognition for Telephone and Video Speech: The JHU-MIT Submission for NIST SRE18," in *Proc. Interspeech 2019*, 2019, pp. 1488–1492.

[6] Tianyan Zhou, Yong Zhao, and Jian Wu, "Resnext and res2net structures for speaker verification," in *Proc. of SLT 2021*, 2021, pp. 301–307.

[7] Joon Son Chung et al., "In defence of metric learning for speaker recognition," in *Interspeech*, 2020, pp. 2977–2981.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, 2016, pp. 770–778.

[9] Georg Heigold, Ignacio Moreno, Samy Bengio, and Noam Shazeer, "End-to-end text-dependent speaker verification," in *Proc. of ICASSP*. 2016, p. 5115–5119, IEEE Press.

[10] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. of ICASSP*, 2018, pp. 4879–4883.

[11] Sergey Novoselov, Andrey Shulipa, Ivan Kremnev, Alexander Kozlov, and Vadim Shchemelinin, "On deep speaker embeddings for text-independent speaker recognition," *CoRR*, vol. abs/1804.10080, 2018.

[12] Aleksei Gusev et al., "Deep speaker embeddings for far-field speaker recognition on short utterances," 2020.

[13] W. Lu L. Wang M. Liu L. Zhang J. Jin J. Xu R. Zhang, J. Wei, "Aret: Aggregated residual extended time-delay neural networks for speaker verification," in *Proc. Interspeech 2020*, 2020, pp. 946–950.

[14] Yun Tang et al., "Deep speaker embedding learning with multi-level pooling for text-independent speaker verification," in *Proc. of ICASSP*. IEEE, 2019, pp. 6116–6120.

[15] Jahangir Alam, Abderrahim Fathan, and Woo Hyun Kang, "Text-Independent Speaker Verification Employing CNN-LSTM-TDNN Hybrid Networks," in *Proc. of SPECOM*, 2021, vol. 12997, pp. 1–13.

[16] Woo Hyun Kang, Jahangir Alam, and Abderrahim Fathan, "Hybrid network with multi-level global-local statistics pooling for robust text-independent speaker recognition," in *Proc. of ASRU*, 2021, pp. 1116–1123.

[17] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[18] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[19] S. Wang, J. Rohdin, O. Plchot, L. Burget, K. Yu, and J. Černocký, "Investigation of specaugment for deep speaker embedding learning," in *Proc. of ICASSP*, 2020, pp. 7139–7143.

[20] Chien-Lin Huang, "Speaker Characterization Using TDNN, TDNN-LSTM, TDNN-LSTM-Attention based Speaker Embeddings for NIST SRE 2019," in *Proc. of Odyssey*, 2020, pp. 423–427.

[21] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *Proc. of ICASSP*, 2018, pp. 5329–5333.

[22] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proc. of ICLR*, Yoshua Bengio and Yann LeCun, Eds., 2015.

[23] Weidi Xie, Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *Proc. of ICASSP*. IEEE, 2019, pp. 5791–5795.

[24] Amirhossein Hajavi and Ali Etemad, "A deep neural network for short-segment speaker recognition," *Proc. Interspeech 2019*, pp. 2878–2882, 2019.

[25] Xu Xiang et al., "Margin matters: Towards more discriminative deep neural network embeddings for speaker recognition," *arXiv preprint arXiv:1906.07317*, 2019.

[26] Joao Monteiro, Isabela Albuquerque, Jahangir Alam, R Devon Hjelm, and Tiago Falk, "An end-to-end approach for the verification problem: learning the right distance," in *ICML*, 2020.