Attention enhanced dynamic kernel convolution for TDNN-based speaker verification

Xiaofan Lang[ab], Ya Li[*a]
[a]Ningbo Artificial Intelligence Institute of Shanghai Jiao Tong University, Ningbo, Zhejiang, China;
[b]Department of Automation, School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China

ABSTRACT

Speaker embedding is a state-of-the-art front-end module, which is used to extract discriminative speaker features for speaker-related tasks. The time delay neural network (TDNN) has been a classical network architecture since it was first applied on speaker related tasks known as X-vector. In this paper, we propose new network structures based on current popular ECAPA-TDNN. We propose a dynamic kernel convolution module to extract features from short-term and long-term context adaptively, thus achieving multi-scale receptive fields. We also apply three enhanced attention modules instead of plain squeeze-excitation (SE) layer to realize more efficient information interaction between channels and spaces. The proposed architectures are superior to the most advanced network, with an optimal equal error rate (EER) of 6.40% and a parameters reduction of 6.32%, they also achieve better performances when speaker utterances are shortened.

Keywords: Text-independent speaker verification, multi-scale, channel attention, dynamic kernel convolution

1. INTRODUCTION

The target of an Automatic Speaker Verification (ASV) system is to verify whether the speaker of an unknown utterance is the claimed one by comparing the test utterance with the registered utterances. There are two main types of Speaker Verification (SV) tasks: text-dependent (TD) and text-independent (TI), the former requires speakers to enrol utterances with a predefined fixed text, while the latter makes use of registered utterances without fixed text to verify the identity of the speaker. Therefore, text-independent SV has more application scenarios and is more challenging compared with text-dependent SV.

In recent years, deep learning methods have been widely used to extract the speaker representation with a fixed dimension called speaker embedding from the given utterance[1-4]. Recently, ResNet architectures[5,6] and Time delay neural network (TDNN) architectures[7-9] have been frequently used in SV tasks. Among these, ECAPA-TDNN[9] and its variants provide[10-12] the current optimal results. ECAPA-TDNN applied a 1D-Res2Net structure as the backbone blocks to process multi-scale speaker features and cut down the quantity of model parameters. It also applied a squeeze-excitation (SE)[13] layer after each Res2Block to obtain channel-wise weights to rescale the frame-level features. Benefiting from its multi-scale feature extraction structure and channel attention strategy, it can obtain more distinguishing features for SV tasks.

In this work, we propose a 1D dynamic kernel convolution (DKC) structure inspired by selective kernel attention[14] to adaptively capture features in short-term and long-term contexts. The proposed module can dynamically select the appropriate kernel size of the convolution according to the channel-wise weights. Inspired by References[15-17], we replace the channel attention layer from the original SE module with a spatial pyramid attention (SPA) module, an efficient channel attention (ECA) module, and a convolutional block attention module (CBAM). The results of our experiments under three evaluation protocols using the proposed DKC module and the improved attention mechanisms show superior performance to baselines. Experiments on utterances of short duration verify the above conclusion.

The remainder of this paper is summarized as follows: Section 2 introduces two baseline system architectures of ResNet and ECAPA-TDNN. Section 3 detailed explains the proposed dynamic kernel convolution and robust channel attention architectures. Section 4 presents the detailed settings of our experiments. The complete results and analysis are given in Section 5. Finally, in Section 6, we summarize the full paper.

* xxathena@sjtu.edu.cn

Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)
edited by Yingfa Lu, Changbo Cheng, Proc. of SPIE Vol. 12506, 1250605
 · 0277-786X · doi: 10.1117/12.2662523

2. BASELINE SYSTEM ARCHITECTURES

In this section, we will describe two distinct speaker verification architectures, both of which perform excellently on speaker verification tasks.

2.1 ResNet

The Thin ResNet-34 proposed in Reference[5] is the first baseline system. It reduces the number of channels in the convolution layer of residual blocks, thereby cutting down the computational cost. The convolution layers of the network process 2D features on the frame-level feature extraction stage. Attentive statistics pooling[4] is used as the temporal pooling strategy to concatenate the first-order and second-order statistics of frame-level features of each frame in the time dimension, thereby generating utterance-level features. See Reference[5] for more details about the topology.

2.2 ECAPA-TDNN

The second baseline system is ECAPA-TDNN. It is a strengthened version of vanilla X-vector system[2,3]. It utilizes the hierarchical residual structure of the Res2Net[18] to capture multi-scale feature. It also integrates the SE module into the residual blocks to rescale the frame-level features per channel. Channel attention method is used at the temporal pooling layer to generate different attention coefficients for different frames for each feature map. The method of multi-layer feature aggregation and summation is used to generate the input features of the pooling layer using a dense layer, which concatenates the output feature maps of both deeper and shallower SE-Res2Blocks.

3. PROPOSED SYSTEM ARCHITECTURES

3.1 1D dynamic kernel convolution

The dynamic kernel convolution (DKC) is a dynamic channel selection mechanism. It is a multi-branch convolutional module which can select the kernel size adaptively in order to capture features in short-term and long-term contexts. The complete structure of the dynamic kernel convolution consists of three parts: split, attention, and select. A two-branch case is depicted in Figure 1.

At the split stage, for the input feature $X \in \mathbb{R}^{C\times T}$,we conduct transformations $\mathcal{F}_1 : X \to U_1 \in \mathbb{R}^{C\times T}$ and $\mathcal{F}_2 : X \to U_2 \in \mathbb{R}^{C\times T}$ as two 1D convolution operators with two different kernel sizes k_1 and k_2 , respectively. In the proposed model, the same kernel size is exploited in two branches, one branch uses the standard convolution while the other uses the dilation convolution. Such a dissimilarity can reduce network parameters while achieving almost the same performance.

At the attention stage, we combine the multi-scale information from different convolution branches by an element-wise summation:

$$U = U_1 + U_2 \tag{1}$$

The mean $\mu \in \mathbb{R}^C$ and standard deviation $\sigma \in \mathbb{R}^C$ of each channel of U are collected by a statistics pooling layer. Specifically, the c-th element of μ and σ is calculated as follow:

$$\mu_c = \mathcal{F}_{mean}(U_c) = \frac{1}{T}\sum_{t=1}^{T} U_c(t) \tag{2}$$

$$\sigma_c = \mathcal{F}_{std}(U_c) = \sqrt{\frac{1}{T-1}\sum_{t}^{T}(U_c(t)^2 - {\mu_c}^2)} \tag{3}$$

Taking the concatenation of μ and σ as input, we can obtain a compact feature $\mathbf{z} \in \mathbb{R}^d$ after a simple full connected (fc) layers as follow:

$$z = \delta(\mathcal{B}(\mathcal{F}_{fc}[\mu;\sigma])) \tag{4}$$

where δ is ReLU function, $\mathcal{B}$ is batch normalization, the full connected layer $\mathcal{F}_{fc}([\mu;\sigma])$ equals to $W^T[\mu;\sigma]$, $\mathbf{W} \in \mathbb{R}^{2C \times d}$ donates the weight metric, a compression ratio r is introduced to generate the squeezed dimension: $d = C/r$. Then softmax-weight of the i-th convolution branch can be obtained after another full connected layer:

$$s_i = \tau(\mathcal{F}_{fc}(z)) = \tau(V_i^T z) \tag{5}$$

where τ is the softmax activation function, $\mathbf{V}_i \in \mathbb{R}^{d \times C}$ is the weight metric, $s_i \in \mathbb{R}^{c \times 1}$ donates the channel-wise attention vector for the feature map $\mathbf{U}_i$ $(i = 1, 2)$.

At the select stage, the c-th channel of the final dynamic representation $\mathbf{Y} \in \mathbb{R}^{C \times T}$ can be calculated by the weighted summation over each branch as the equation:

$$Y_c = \sum_i s_{i,c} \cdot U_{i,c}, \quad \sum_i s_{i,c} = 1 \tag{6}$$

where $s_{i,c}$ represents the c-th element of s_i, and $U_{i,c}$ represents the c-th row of U_i.

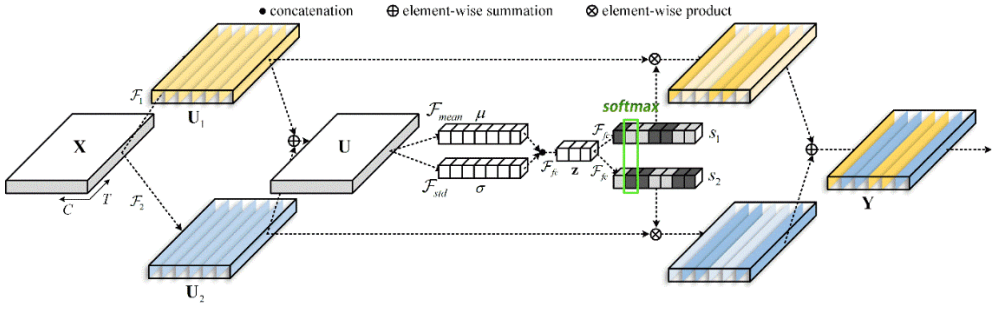

Figure 1. Dynamic Kernel Convolution (DKC) module.

3.2 Enhanced attention mechanisms

Considering that a wider temporal context contains more speaker feature information, SE module is used in two baseline systems to rescale the frame-level features using global properties of the utterance. Specifically, frame-level features are compressed through spatial dimension using a global average pooling, and channel-wise weights are produced using a multi-layer perceptron (MLP). Some enhanced attention mechanisms can replace SE module to further mine the context information of features.

Spatial pyramid attention (SPA) replaces the single-scale global average pooling layer in SE module with a group of adaptive average pooling (AAP) layers of different sizes. Such a spatial pyramid structure can capture more spatial information of the input feature map.

For a given feature $\mathbf{X} \in \mathbb{R}^{C \times T}$, let $w \in \mathbb{R}^{C \times 1}$ be the channel-wise attention weight vector and $\mathbf{X}' \in \mathbb{R}^{C \times T}$ donates the output feature map after channel rescaling. SPA module can be presented as following equations:

$$A_i = \mathcal{R}(\mathcal{F}_{aap}(x, s_i)) \tag{7}$$

$$A = [A_1; A_2; A_3] \tag{8}$$

$$w = \tau(\mathcal{F}_{fc}(\delta(\mathcal{F}_{fc}(A)))) \tag{9}$$

$$X' = X \cdot w \tag{10}$$

where $F_{aap}(X, s_i)$ donates the AAP layer with the output size of s_i, different outputs are resized into three 1-dimension vectors and concatenated to generate a 1-dimension attention map $\mathbf{A} \in \mathbb{R}^{C' \times 1}$, $\mathcal{R}(\cdot)$ donates the resize function. $\delta(\cdot)$, $\tau(\cdot)$ and $\mathcal{F}_{fc}(\cdot)$ represent ReLU, sigmoid activation function and full connected layer, respectively.

Efficient channel attention (ECA) utilizes a 1-dimension convolution instead of MLP to generate channel-wise attention weights. The convolution operator can appropriately capture local-channel interaction while involving fewer parameters, which guarantees both effectiveness and efficiency. ECA module can be presented as following equations:

$$w = \tau(\mathcal{C}_k(\mathcal{F}_{gap}(X))) \tag{11}$$

$$X' = X \cdot w \tag{12}$$

where $\mathcal{F}_{gap}(\cdot)$ donates the global average pooling layer and $\mathcal{C}_k(\cdot)$ represents the 1D convolution with kernel size of *k*.

Convolutional block attention module (CBAM) is composed of channel sub-module and spatial sub-module. The channel sub-module generates channel-wise attention weights utilizing max-pooling outputs and average-pooling outputs along time axis, then the spatial sub-module generates time-wise attention weights utilizing the two pooling outputs along channel axis and forwards them to a convolutional layer. CBAM takes both channel-wise and time-wise information interaction into consideration. It can be presented as following equations:

$$w_c = \tau(\mathcal{F}_{mlp}(\mathcal{P}_{\max}(X,t)) + \mathcal{F}_{mlp}(\mathcal{P}_{avg}(X,t))) \tag{13}$$

$$w_s = \tau(\mathcal{C}_k([\mathcal{P}_{\max}(X,c); \mathcal{P}_{avg}(X,c)])) \tag{14}$$

$$X' = (X \cdot w_c) \cdot w_s \tag{15}$$

where $w_c \in \mathbb{R}^{C\times 1}$ and $w_s \in \mathbb{R}^{1\times T}$ donates channel-wise and time-wise attention weight vector, respectively. $\mathcal{P}_{max}(\cdot,c)$ and $\mathcal{P}_{max}(\cdot,t)$ represent max-pooling layer along channel and time axis, respectively. $\mathcal{P}_{avg}(\cdot,c)$ and $\mathcal{P}_{avg}(\cdot,t)$ represent average-pooling layer similarly. $\mathcal{F}_{mlp}(\cdot)$ is the MLP module that can be represented by $\mathcal{F}_{fc}(\delta(\mathcal{F}_{fc}(\cdot)))$ specifically.

4. EXPERIMENTS

4.1 Datasets

VoxCeleb[19,20] is a commonly used dataset for speaker related tasks. VoxCeleb1[19] contains more than 150000 utterances from 1251 speakers, VoxCeleb2[20] contains more than 1 million utterances from 5994 speakers. We train models with a subset of VoxCeleb2 that we make by selecting 10 utterances randomly for each speaker. We use MUSAN dataset[21] and RIR dataset[22] to realize online augmentation. The augmentation methods are the same as the settings in Reference[23]. SpecAugment[24] is added to the log Fbanks of the training samples with a frequency masking dimension of 8 and a temporal masking dimension of 10. All models are evaluated on VoxCeleb1.

4.2 Training settings

A two-second segment is extracted randomly from each training sample. The input features are 80-dimentional log Fbanks extracted from a hamming window with 25ms length and 10ms shift.

The proposed dynamic kernel convolution is used to replace the conventional 1D convolutions in each 1D SE-Res2block of ECAPA-TDNN baseline. Two branches are set for each DKC module. The first branch uses convolution with kernel size of 3, and the second branch uses the dilation convolution with kernel size of 3 and dilation factor of 2. Channel reduction ratio *r* is set at 16. Scale of each Res2block is set at 8.

We replace the SE layer with SPA, ECA, and ABAM, respectively. In SPA, three AAP layers are set with sizes of 1, 2, and 4, and the bottleneck dimension is set at 128. In ECA, kernel size of the convolution layer is set at 5. In CBAM, the bottleneck dimension in channel sub-module is set at 128, and kernel size of the convolution layer is set at 7.

All models use attentive statistics pooling as the temporal pooling strategy to generate utterance-level features. All convolutions for frame-level feature extraction are set to 512 channels, the last full connected layer is set as 192 dimensions to fix the dimension of the output speaker embeddings for all models. We train all models using AAM-softmax loss[25] with a margin of 0.2 and a scale of 30.

4.3 Evaluation protocol

Performance of models is measured by equal error rate (EER) and minimum detection cost function (MinDCF) with $P_{target} = 10^{-2}$ and $C_{FA} = C_{FR} = 1$. Models are evaluated on VoxCeleb1-O, VoxCeleb1-E and VoxCeleb1-H. For the complete utterances, we split each utterance into 5 segments and then extract speaker embedding for each segment. For each trail, we formulate a score matrix with the size of 5×5 to calculate the cosine similarities of all pairs of segments. All scores are averaged to obtain the final similarity score. For each trail, we take segments with a duration of 4 s and 2 s to test the robustness of the models on short utterances. Cosine similarity is calculated directly as the final score.

4.4 Implementation details

We implement all models with PyTorch framework and conducted them on three NVIDIA Quadro RTX 8000 GPUs, each GPU has 48GB of memory. We train 80 epochs for all models. The initial learning rate is set at 0.001 and decreases 3% in every one epoch. The network parameters are updated by Adam optimizer[26]. The mini-batch size for training is 400.

5. RESULTS

5.1 Results on VoxCeleb1

Table 1 gives an overview of the verification performance of all implemented models for full utterances, Tables 2 and 3 show the performance overview for 4 s and 2 s segments, respectively. DKC-TDNN with ECA and CBAM modules has reduced network parameters compared with baseline systems. Most systems we proposed apparently outperform the baseline systems in the case of full utterances. This reveals that it is beneficial to capture multi-scale features adaptively with dynamic convolution. The improved channel and spatial attention method can indeed bring performance improvement in a TDNN-based architecture. DKC-TDNN with SPA module, in particular, achieves EER of 6.40% and MinDCF of 0.375, gives a relative improvement of 17.84% and 1.99% in EER compared with ResNet and ECAPA-TDNN, respectively. DKC-TDNN with ECA achieves similar performance enhancement. DKC-TDNN with CBAM shows the minimal improvement. Almost all proposed systems have just slightly reduced or even increased MinDCF compared with ECAPA-TDNN baseline system. When the utterances are shortened to 4 s and 2 s, proposed systems have better performances either in EER or in MinDCF compared with two baseline systems. Finally, we assume branch structure reduces the inference speed of the network by comparing the computing speeds of all the proposed models to two baseline systems.

Table 1. Performance of models on VoxCeleb1-O, VoxCeleb1-E and VoxCeleb1-H for full utterances.

Model	**Params. (M)**	**VoxCeleb1-O**		**VoxCeleb1-E**		**VoxCeleb1-H**	
		EER (%)	MinDCF	EER (%)	MinDCF	EER (%)	MinDCF
ResNet	6.72	7.79	0.458	8.07	0.474	11.55	0.587
ECAPA-TDNN	6.65	6.53	0.374	6.79	0.396	9.89	0.502
DKC-TDNN (SPA)	7.73	**6.40**	**0.375**	**6.69**	**0.392**	**9.82**	**0.504**
DKC-TDNN (ECA)	6.23	6.41	0.378	6.71	0.397	9.89	0.505
DKC-TDNN (CBAM)	6.60	6.47	0.379	6.78	0.398	9.98	0.506

Table 2. Performance of models on VoxCeleb1-O, VoxCeleb1-E and VoxCeleb1-H for 4 s segments.

Model	**VoxCeleb1-O**		**VoxCeleb1-E**		**VoxCeleb1-H**	
	EER (%)	MinDCF	EER (%)	MinDCF	EER (%)	MinDCF
ResNet	9.35	0.534	9.35	0.525	13.08	0.638
ECAPA-TDNN	7.99	0.435	7.90	0.445	11.33	0.558
DKC-TDNN (SPA)	7.69	0.439	7.81	0.441	11.32	0.559

Model	VoxCeleb1-O		VoxCeleb1-E		VoxCeleb1-H	
	EER (%)	MinDCF	EER (%)	MinDCF	EER (%)	MinDCF
DKC-TDNN (ECA)	**7.56**	**0.433**	**7.80**	**0.445**	**11.33**	**0.563**
DKC-TDNN (CBAM)	7.81	0.444	7.93	0.447	11.45	0.563

Table 3. Performance of models on VoxCeleb1-O, VoxCeleb1-E and VoxCeleb1-H for 2 s segments.

Model	VoxCeleb1-O		VoxCeleb1-E		VoxCeleb1-H	
	EER (%)	MinDCF	EER (%)	MinDCF	EER (%)	MinDCF
ResNet	13.43	0.658	13.67	0.668	17.68	0.767
ECAPA-TDNN	12.36	0.628	12.72	0.624	16.57	0.731
DKC-TDNN (SPA)	12.16	0.610	12.54	0.620	16.50	0.722
DKC-TDNN (ECA)	**12.00**	**0.612**	**12.51**	**0.621**	**16.59**	**0.734**
DKC-TDNN (CBAM)	12.29	0.615	12.60	0.625	16.61	0.731

5.2 Ablation studies

We conduct a series of ablation studies to reveal the impact of each component in our proposed architectures on the final performance. Table 4 gives the results on VoxCeleb1-O. Experiment A.0 is the ECAPA-TDNN baseline system. Experiment B.0 adds the DKC module to replace conventional convolution in each branch of res2blocks of the baseline system. Experiments A.1, A.2 and A.3 replace the SE module of the baseline system as SPA, ECA and CBAM, respectively. Experiments B.1, B.2 and B.3 replace the SE module of experiment B.0 similarly.

The results of experiments A.0 and B.0 demonstrate the effectiveness of the DKC module. Benefits of all three enhanced attention mechanisms can also be evaluated by comparing experiments A.1, A.2 and A.3 with A.0. In particular, CBAM gives the lowest EER of 6.38% in three of the attention enhanced methods. we suppose that the temporal dimension information aggerated by the spatial attention sub-model in CBAM is useful for speaker feature extraction. Experiments B.1 and B.2 show that combine DKC module with SPA or ECA can achieve better results. However, comparing experiment B.3 with A.3, combine DKC module with CBAM achieves a worse result. The conflict of the two mechanisms needs further study. Ablation study of attention enhanced DKC on VoxCeleb1-O.

Table 4. Ablation study of attention enhanced DKC on VoxCeleb1-O.

	Systems	EER (%)	MinDCF
A.0	Baseline (ECAPA-TDNN)	6.53	0.374
A.1	Baseline with SPA	6.44	0.370
A.2	Baseline with ECA	6.50	0.387
A.3	Baseline with CBAM	6.38	0.383
B.0	DKC without Channel Att.	6.49	0.386
B.1	DKC with SPA	6.40	0.375
B.2	DKC with ECA	6.41	0.378
B.3	DKC with CBAM	6.47	0.379

6. CONCLUSION

In this paper, we introduce a dynamic kernel convolution and three enhanced channel attention methods for automatic speaker verification to achieve multi-scale receptive fields and more efficient information interaction in speaker feature extraction. Vast experiments on three evaluation protocols of VoxCeleb1 demonstrate that the novel architectures outperform two baseline systems both in full utterances and short-duration utterances. Our ablation study shows the effectiveness of proposed dynamic kernel convolution and three attention mechanisms, respectively.

REFERENCES

[1] Variani, E., Lei, X., McDermott, E., Moreno, I. L. and Gonzalez-Dominguez, J., "Deep neural networks for small footprint text-dependent speaker verification," 2014 IEEE Inter. Conf. on Acoustics, Speech and Signal Processing, 4052-4056 (2014).

[2] Snyder, D., Garcia-Romero, D., Povey, D. and Khudanpur, S., "Deep neural network embeddings for text-independent speaker verification," Interspeech, 999-1003 (2017).

[3] Snyder, D., Garcia-Romero, D., Sell, G., Povey, D. and Khudanpur, S., "X-vectors: Robust DNN embeddings for speaker recognition," 2018 IEEE Inter. Conf. on Acoustics, Speech and Signal Processing, 5329-5333 (2018).

[4] Okabe, K., Koshinaka, T. and Shinoda, K., "Attentive statistics pooling for deep speaker embedding," Interspeech, 2252-2256 (2018).

[5] Chung, J. S., Huh, J., Mun, S., Lee, M., Heo, H. S., Choe, S., Ham, C., Jung, S., Lee, B. J. and Han, I., "Indefence of metric learning for speaker recognition," Interspeech, (2020).

[6] Zhou, T., Zhao, Y. and Wu, J., "Resnext and res2net structures for speaker verification," 2021 IEEE Spoken Language Technology Workshop, 301-307 (2021).

[7] Snyder, D., Garcia-Romero, D., Sell, G., McCree, A., Povey, D. and Khudanpur, S., "Speaker recognition for multi-speaker conversations using x-vectors," 2019 IEEE Inter. Conf. on Acoustics, Speech and Signal Processing, 5796-5800 (2019).

[8] Povey, D., Cheng, G., Wang, Y., Li, K., Xu, H., Yarmohammadi, M. and Khudanpur, S., "Semi-orthogonal low-rank matrix factorization for deep neural networks," Interspeech, 3743-3747 (2018).

[9] Desplanques, B., Thienpondt, J. and Demuynck, K., "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," Interspeech, 3830-3834 (2020).

[10] Thienpondt, J., Desplanques, B. and Demuynck, K., "Integrating frequency translational invariance in TDNNS and frequency positional information in 2d Res2Net to enhance speaker verification," Interspeech, 2302-2306 (2021).

[11] Liu, T., Das, R. K., Lee, K. A. and Li, H., "MFA: TDNN with multi-scale frequency-channel attention for text-independent speaker verification with short utterances," 2022 IEEE Inter. Conf. on Acoustics, Speech and Signal Processing, (2022).

[12] Mun, S. H., Jung, J. W., and Kim, N. S., "Selective kernel attention for robust speaker verification," Interspeech, (2022).

[13] Hu, J., Shen, L. and Sun, G., "Squeeze-and-Excitation networks," 2018 IEEE Conf. on Computer Vision and Pattern Recognition, 7132-7141 (2018).

[14] Li, X., Wang, W., Hu, X. and Yang, J., "Selective kernel networks," 2019 IEEE Conf. on Computer Vision and Pattern Recognition, 510-519 (2019).

[15] Guo, J., Ma, X., Sansom, A., McGuire, M., Kalaani, A., Chen, Q., Tang, S., Yang, Q. and Fu, S., "Spanet: Spatial pyramid attention network for enhanced image recognition," 2020 IEEE Inter. Con. on Multimedia and Expo, 1-6 (2020).

[16] Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W. and Hu, Q., "ECA-Net: Efficient channel attention for deep convolutional neural networks," 2020 IEEE Conf. on Computer Vision and Pattern Recognition, 11531-11539 (2020).

[17] Woo, S., Park, J., Lee, J. Y. and Kweon, I. S., "Cbam: Convolutional block attention module," 2018 European Conf. on Computer Vision, 3-19 (2018).

[18] Gao, S., Cheng, M. M., Zhao, K., Zhang, X., Yang, M. H. and Torr, P. H. S., "Res2net: A new multi-scale backbone architecture," 2019 IEEE Transactions on Pattern Analysis and Machine Intelligence, 652-662 (2019).

[19] Nagrani, A., Chung, J. S. and Zisserman, A., "Voxceleb: A large-scale speaker identification dataset," Interspeech, 2616-2610 (2017).

[20] Chung, J. S., Nagrani, A. and Zisserman, A., "Voxceleb2: Deep speaker recognition," Interspeech, 1086-1090 (2018).

[21] Snyder, D., Chen, G. and Povey, D., "Musan: A music, speech, and noise corpus," arXiv preprint arXiv:1510.08484, (2015).

[22] Ko, T., Peddinti, V., Povey, D., Seltzer, M. L. and Khu-Danpur, S., "A study on data augmentation of reverberant speech for robust speech recognition," 2015 IEEE Inter. Conf. on Acoustics, Speech and Signal Processing, 5220-5224 (2015).

[23] Das, R. K., Tao, R. and Li, H., "HLT-NUS submission for 2020 NIST conversational telephone speech SRE," arXiv preprint arXiv:2111.06671, (2021).

[24] Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D. and Le, Q. V., "Specaugment: A simple data augmentation method for automatic speech recognition," Interspeech, 2613-2617 (2019).

[25] Deng, J., Guo, J., Xue, N. and Zafeiriou, S., "Arcface: Additive angular margin loss for deep face recognition," 2019 IEEE Conf. on Computer Vision and Pattern Recognition, 4690-4699 (2019).

[26] Kingma, D. and Ba, J., "Adam: A method for stochastic optimization," 2015 Inter. Conf. on Learning Representations, (2015).