INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Speaker Representation Learning using Global Context Guided Channel and Time-Frequency Transformations

Wei Xia, John H.L. Hansen

Center for Robust Speech Systems, University of Texas at Dallas, TX 75080

wei.xia@utdallas.edu, john.hansen@utdallas.edu

Abstract

In this study, we propose the global context guided channel and time-frequency transformations to model the long-range, non-local time-frequency dependencies and channel variances in speaker representations. We use the global context information to enhance important channels and recalibrate salient time-frequency locations by computing the similarity between the global context and local features. The proposed modules, together with a popular ResNet based model, are evaluated on the VoxCeleb1 dataset, which is a large scale speaker verification corpus collected in the wild. This lightweight block can be easily incorporated into a CNN model with little additional computational costs and effectively improves the speaker verification performance compared to the baseline ResNet-LDE model and the Squeeze&Excitation block by a large margin. Detailed ablation studies are also performed to analyze various factors that may impact the performance of the proposed modules. We find that by employing the proposed L2-tf-GTFC transformation block, the Equal Error Rate decreases from 4.56% to 3.07%, a relative 32.68% reduction, and a relative 27.28% improvement in terms of the DCF score. The results indicate that our proposed global context guided transformation modules can efficiently improve the learned speaker representations by achieving time-frequency and channel-wise feature recalibration.

Index Terms: Text-independent speaker verification, global context modeling, attention mechanism, representation learning

1. Introduction

Automatic Speaker Verification (ASV) task involves determining a person's identity from audio streams. It provides a natural and efficient way for biometric identity authentication. Being able to perform text-independent speaker verification that does not utilize any fixed input text content can significantly help us retrieve a target person. We can use speaker recognition for audio surveillance [1], computer access control, and telephone voice authentication for long distance calling [2, 3]. It is also helpful for targeted speech enhancement and separation systems if we have good speaker embeddings [4, 5].

Learning a good speaker representation is crucial to speaker verification tasks. The paradigm has shifted from GMM-UBM and factor analysis based methods like i-vector [6, 7] with a probabilistic linear discriminant (PLDA) back-end [8, 9] to deep neural network based models. Different neural network architectures [10, 11, 12] were explored to improve the speaker embedding extraction. Margin based softmax loss functions like Angular Softmax [13], Additive Margin Softmax [14], and recently proposed Additive Angular Margin loss [15] were useful to learn a more discriminative speaker embedding. Several new temporal pooling methods like attentive pooling [16], Spatial Pyramid Pooling [17] and GhostVLAD [18] were presented to aggregate the variable length input features to a fixed-length utterance level representation. Various noise and language robust speaker recognition models [19, 20], and training paradigms [21, 22] have been proposed and significantly improve speaker verification systems' performance. Cai et al. [23] introduced a Learnable Dictionary Encoding (LDE) layer to combine frame-level speaker features to an utterance-level speaker embedding. This ResNet with LDE encoding model has become very successful in various speaker recognition tasks. We use it as the baseline in this study.

Many speaker verification models are based on convolution neural networks, which learn filters to capture local patterns. However, the filter that only operates on the neighboring local context cannot capture long-range, non-local global information. Also, the time-frequency (TF) and channel information at salient regions may not be well emphasized through a standard convolution layer. Many recent works [24, 25, 26, 27, 28] try to alleviate these issues by improving the encoding of TF and channel information. One promising approach to accomplish this is a component called the "Squeeze & Excitation" (SE) block [24, 29], which explicitly models the inter-dependencies between the channels of feature maps. Deformable network [25] designs deformable convolution to enhance spatial modeling ability. AAConv [26] uses two-dimensional relative self-attention to augment the convolution operator.

In this study, we introduce a generalized global time frequency context modeling framework for text-independent speaker verification. Speech signals contain different information at each time-frequency location. For example, we may pay more attention to high energy parts in the spectrogram. Our proposed approach tries to better capture long-range time frequency dependencies and channel variances. We firstly present the l_p norm based attentive time-frequency context embedding to efficiently model the global speech contextual information. With carefully designed components, the Global Time Frequency Context (GTFC) vector is used for channel and time-frequency wise feature recalibrations. It aims to get a better combination of the Non-local block [30] and SE block [24] to adaptively recalibrate the learned feature map and provides time-frequency attention to specific regions. Further, we combine the channel wise GTFC and time-frequency GTFC on the score level by a linear fusion. It aggregates the unique properties of each method and makes feature maps more informative on both domains. We show that with the linear fusion, the Equal Error Rate (EER) of the ASV system decreases from 4.56% to 2.70%, a relative 40.79% reduction. It also has a 38.10% relative improvement of DCF compared to the baseline ResNet34-LDE model.

In the following sections, we describe the global time frequency modeling framework and corresponding baseline systems in Section 2. We provide detailed explanations of our experiments in Section 3, as well as results and discussions in Section 4. Finally we conclude our work in Section 5.

http://dx.doi.org/10.21437/Interspeech.2020-1845

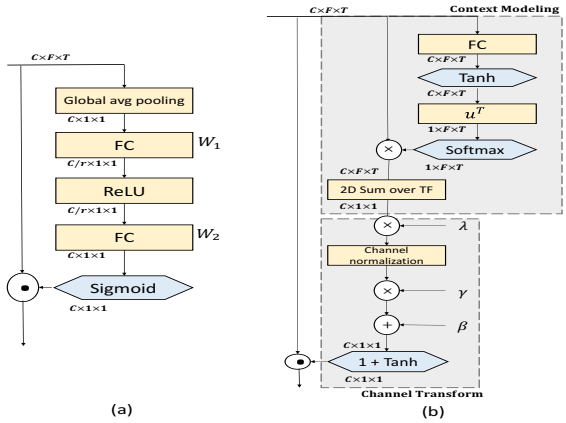

Figure 1: *(a) SE block. (b) Proposed global time-frequency context modeling framework and channel-wise transformation.*

2. Speaker representation learning

2.1. ResNet based speaker recognition backbone

We use the ResNet34 with Learnable Dictionary Encoding (LDE) [23] as our baseline speaker recognition model. It uses a well-known ResNet-34 architecture and a Dictionary Learning method to aggregate the variable-length input sequence into a fixed utterance-level speaker embedding.

2.2. Revisiting SE channel attention

Squeeze and Excitation Network (SE-Net) [24] is a well-known method proposed recently to rescale the input feature map to highlight useful channels. Shown in Fig. 1 (a), a global average pooling layer is used to generate a channel-wise vector. Two fully-connected layers $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ capture the local channel dependencies. The dimensional reduction factor r indicates the bottleneck in the channel excitation block. Finally, with a sigmoid layer, the channel-wise attention vector is obtained to emphasize essential channels.

2.3. Global time frequency context modeling framework

The global context information and channel relationships in the SE-Net are inherently implicit. To better model long-range relationships and local interactions, we propose a new generalized framework of global context modeling for channel and time-frequency wise feature recalibration. We compute an attention map for each time frequency location and attentively pool the corresponding feature values with a l_p norm unit to get the global representation.

We firstly learn a global time frequency embedding $\mathbf{g} \in \mathbb{R}^C$, then apply channel wise transformation by broadcasting the global TF context vector to each channel; or time-frequency wise transformation by scoring the similarity between the global TF context vector and the local feature vector to get an attention map.

In Fig. 1 (b), we show the process to learn the Global Time-Frequency Context (GTFC) embedding and apply it for the channel wise feature map enhancement: (a) the context modeling module groups the features of all positions together via the l_p norm attentive pooling; (b) GTFC is normalized to capture channel-wise dependencies; (c) we use a fusion function to distribute the context vector across channels. In the following, we describe the process in detail, and later in Section 2.3.4 we apply the GTFC embedding for the time-frequency feature enhancement.

2.3.1. L_p-norm attentive time frequency context embedding

A global context embedding module is firstly designed based on l_p-norm to aggregate the non-local, long-range time-frequency relationship in each channel. Since individual T-F locations may have different importance, we also use an attention mechanism to focus on salient regions that may have more considerable impact on the global context. The module can exploit comprehensive contextual information outside small receptive fields of convolutional layers to better encode the global T-F information. Given the embedding weight $\boldsymbol{\lambda} = [\lambda_1, ..., \lambda_C]$ along channel and an input feature vector $\mathbf{x}^{i,j} \in \mathbb{R}^C$, the module is defined as the following,

$$g_c = \lambda_c f(\alpha_{i,j}, \|x_c\|_p) = \lambda_c \left\{ \left[\sum_{i=1}^{F} \sum_{j=1}^{T} \alpha_{i,j} |x_c^{i,j}|^p \right] \right\}^{\frac{1}{p}} \quad (1)$$

$$\alpha_{i,j} = softmax\left(\mathbf{u}_\alpha^{\mathsf{T}} tanh(\boldsymbol{W}_\alpha |\mathbf{x}^{i,j}|^p + \mathbf{b}) \right) \quad (2)$$

where α_{ij} is the learned attention weight at a time-frequency location (i, j) through an MLP $\boldsymbol{W}_\alpha$ and a hidden vector $\mathbf{u}_\alpha$. The l_p-norm unit is efficient at representing nonlinear, complex activations, and is a general form of mean or max pooling with positive values. It defines a spherical shape in a non-Euclidean space and summarizes a high-dimensional collection of neural responses. It can avoid the inferior result that average pooling may lead to in some extreme cases. Additionally, we use an attention mechanism to learn the weight at each time-frequency location for a better global context representation. We compare the performance of various l_p-norms and choose the best one, l_2-norm, to be our default setting. Trainable parameter λ_c is introduced to control the weight of each channel because they may have different significances.

2.3.2. Channel normalization

We use a channel normalization method to scale the GTFC vector to capture the competition (high variance) relationship among neuron outputs. It is a lightweight operator and reduces the computational cost of the two FC layers used in the SE block from $O(C^2)$ to $O(C)$ but still with a steady performance. Let $\mathbf{g} = [g_1, ..., g_c]$, channel normalization is formulated as,

$$\hat{g}_c = \frac{k g_c}{\|\mathbf{g}\|_2} = \frac{k g_c}{\sqrt{\sum_{c=1}^{C} g_c^2 + \epsilon}} \quad (3)$$

where ϵ is a small constant for numerical stability. To prevent a too small value of g_c when C is large, we use a scalar k and set it as $\sqrt{C}$ to normalize the scale of g_c,

2.3.3. Channel gating adaptation

We finally use a gating mechanism on the normalized GTFC vector to perform channel-wise recalibration on the original feature maps. Let gating weights $\boldsymbol{\gamma} = [\gamma_1, ..., \gamma_C]$ and gating biases $\boldsymbol{\beta} = [\beta_1, ..., \beta_C]$. They are trainable parameters to adjust the activations of gates channel-wisely. We use the following gating function with the identity mapping ability.

$$\hat{\boldsymbol{X}}_c = \boldsymbol{X}_c \cdot [1 + \tanh(\gamma_c \hat{g}_c + \beta_c)] \quad (4)$$

The scale of each original feature map $\boldsymbol{X}_c \in \mathbb{R}^{F \times T}$ in the channel c is adapted by its corresponding gate so that important channels are emphasized, and less important ones are diminished. Also, when $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are zeros, we can pass the original features unimpeded to the next layer. This allows any layer to be

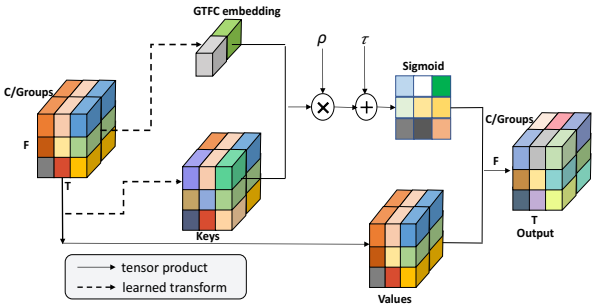

Figure 2: *Time-frequency transformation using a group-wise content interaction between the GTFC embedding and the local T-F feature vector.*

represented as its initial input. Inspired by ResNet, being able to model the identity mapping can make the network easy to be optimized and robust to the degradation problem in deep networks. Therefore, we initialize $\boldsymbol{\gamma}$ to and $\boldsymbol{\beta}$ to 0 in the proposed blocks.

2.3.4. Time frequency content interaction

To capture the time-frequency relationship and analyze which time-frequency location we need to pay attention to, we also propose a way to compute the TF attention map based on the correlation between the GTFC embedding and local feature vectors. The group wise time-frequency enhancement method is illustrated in Fig. 2. Firstly we divide the C channels, $F \times T$ convolutional feature map into G groups along the channel dimension. We assume that each group could gradually learn a specific response during the training process. In each group, we have a set of local feature vectors $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_m\}, \mathbf{x}_i \in \mathbb{R}^{C/G}, m = F \times T$. We ideally hope to get features with strong responses at salient time frequency positions (e.g., high energy region). However, due to noises and reverberations, we may not be able to get desirable neuron activations after convolution. To reduce this problem, the group-wise normalized GTFC embedding $\hat{\mathbf{g}}$ is used as a global group representation, and we compute the correlation between the GTFC vector with the local feature vector $\mathbf{x}_i$ at each time-frequency location. The similarity score is calculated as the following.

$$e_i = \text{score}(\hat{\mathbf{g}}, \mathbf{x}_i) = \hat{\mathbf{g}}^{\mathsf{T}} \boldsymbol{W}_e \mathbf{x}_i \tag{5}$$

With the normalized GTFC vector $\hat{\mathbf{g}}$, we can generate the corresponding importance coefficient e_i for each position, using the general dot product scoring function [31] in Eq. (5). $\boldsymbol{W}_e$ is a weight matrice to be learned. In order to prevent the biased magnitude of coefficients between various samples, we also normalize $\mathbf{e}$ over the time-frequency domain,

$$\hat{e}_i = \frac{e_i - \mu_e}{\sigma_e + \epsilon}, \mu_e = \frac{1}{m}\sum_{j=1}^{m} e_j, \sigma_e^2 = \frac{1}{m}\sum_{j=1}^{m}(e_j - \mu_e)^2 \tag{6}$$

where ϵ (e.g., 1e-5) is a constant added for numerical stability. To make sure that the normalization inserted in the network can represent the identity transform, we introduce a pair of parameters (ρ, τ) for each coefficient e_i, which scale and shift the normalized value. Finally, to obtain the enhanced feature vector $\hat{\mathbf{x}}_i$, the original $\mathbf{x}_i$ is scaled by the generated importance coefficients via a sigmoid function over the space,

$$s_i = \rho \hat{e}_i + \tau \tag{7}$$

$$\hat{\mathbf{x}}_i = \mathbf{x}_i \cdot \sigma(s_i) \tag{8}$$

All the enhanced features form the recalibrated new feature group. Note that the total number of ρ and τ is the number of groups, which is negligible compared with millions of model parameters.

3. Experimental Setup

3.1. Dataset and feature extraction

To study the effectiveness of the global time-frequency context guided transformation for speaker representation learning, we used VoxCeleb1 [32] dataset for experiments. It is a large scale text-independent dataset extracted from YouTube videos that contain 153,516 utterances for 1251 celebrities. The proposed speaker model is trained only on the VoxCeleb1 development set which contains 1211 speakers. We do not use any data augmentation strategies. There are 40 speakers in the test set with 4874 utterances. The performance is reported in terms of Equal Error Rate (EER) and minimum Detection Cost Function (DCF) with $P_{target} = 0.01$.

We compute 64 dimensional log-mel filter-bank energies (fbank) on the frame level as input features. A Hamming window of length 25 ms with a 10 ms frame shift is used to extract the fbanks from input audio signals. We use a random chunk of 300-800 frame features of each audio file as the input to the network, like the strategy used in [23]. The input feature is mean and variance normalized on the frame level. Kaldi energy-based VAD is used to remove silent frames.

3.2. Model training

The baseline model is a ResNet-34 model with LDE pooling [23] and angular softmax [13] (margin $m = 4$). We split 90% of the development set for training, and the remaining 10% for validation and parameter tuning. It is found that the best input feature setting is a frame length of 25 ms, 64 dimensional fbanks. We use the same ResNet34-LDE model parameter settings in [23], where residual layers' channel sizes are 16, 32, 64, and 128 respectively.

For our proposed GTFC based models, *Swish* [33] activation function is used at all positions in the ResNet34-LDE model, and we find it is helpful to improve the performance. The model is trained on the VoxCeleb1 training split for 50 epochs with a batch size of 120 on 4 GPUs. We use a SGD optimizer with 0.9 momentum and initialize the learning rate as 10^{-3} as well as 1e-4 weight decay. The learning rate is reduced by 0.1 when the validation loss does not reduce for 10 epochs. The extracted utterance-level embedding size is 128. l_2 norm is used as the default setting in the l_p norm unit. The general dot-product scoring function is applied to compute the time-frequency attention matrix. We insert the proposed GTFC module after the last Batch Norm layer in each residual basic block.

For the backend, we use LDA to reduce the dimension of the embeddings to 120, and they are also centered and length normalized. PLDA scoring is applied to evaluate the verification performance.

4. Results and Discussions

4.1. Experimental results

In order to thoroughly evaluate our proposed methods, we conduct a detailed ablation analysis in this section. We first perform experiments on the GTFC guided channel wise transformation (c-GTFC), followed by the group wise time frequency transfor-

mation (tf-GTFC), and then the linear fusion results for speaker verification. Finally, we analyze various factors that may affect the performance of GTFC blocks.

From Table 1, we observe that our proposed channel wise GTFC block (L2-c-GTFC+ResNet34-LDE) improves the SV performance by a large margin compared with the ResNet34-LDE model. With the L2-c-GTFC block, overall EER of the ResNet34-LDE model decreases from 4.56% to 3.13%, relatively 31.36%; also from 4.01% to 3.13% compared with the SE block, a relative reduction of 21.95%. It may suggest that our proposed l_2 norm based global time-frequency context block can greatly recalibrate the significant feature regions and improve the speaker verification performance.

We also find a consistent performance improvement from L2-c-GTFC model to the L2-tf-GTFC results. The EER reduces from 3.13% to 3.07%, with a relative 1.92% improvement. It indicates that the L2-tf-GTFC might be more efficient than L2-c-GTFC, which aligns with our assumption that the time-frequency space may have more meaningful information than channels for the SV task.

Table 1: *SV results on the VoxCeleb1 test set using various models and ResNet34-LDE +our proposed GTFC guided blocks.*

Model	EER (%)	DCF	Train Set
Ivector [32]	8.80	0.7300	VoxCeleb1
Xvector [16]	3.85	0.4060	VoxCeleb1
UtterIdNet [12]	4.26	N/A	VoxCeleb2
SPE [17]	4.20	0.4220	VoxCeleb1
ResNet34-LDE [23]	4.56	0.4410	VoxCeleb1
ResNet34-LDE +SE	4.01	0.3940	VoxCeleb1
+L1-c-GTFC (ours)	4.14	0.4141	VoxCeleb1
+L1-tf-GTFC (ours)	3.38	0.3435	VoxCeleb1
① +L2-c-GTFC (ours)	**3.13**	**0.3169**	VoxCeleb1
② +L2-tf-GTFC (ours)	**3.07**	**0.3207**	VoxCeleb1
① & ② linear fusion	**2.70**	**0.2730**	VoxCeleb1

We further compare the global time frequency context embedding with different l_p norm units. We investigate the l_1 norm and l_2 norm based GTFC blocks and observe that l_2 norm based GTFC blocks perform better than the l_1 norm based blocks in all cases. We also tried to set the order p as a learning parameter, but the result is usually worse than the l_2 norm, so we use l_2 norm based GTFC blocks in all our experiments and subsequent analysis. Finally, a score level linear fusion with equal weights is employed to combine the L2-c-GTFC and the L2-tf-GTFC results. It achieves the best results with 2.70% EER and 0.2730 DCF.

4.2. Empirical analysis

Channel adaptation operator. We examine the activation function of the channel gating adaptation in the L2-c-GTFC block with a few different non-linear activation functions and show the results in Table 2 (a). All the non-linear gating adaptation operators achieve promising performance, and $1 + tanh$ gets the best result. It shows that the identity mapping in the gating function is helpful for the channel adaptation.

Normalization components ρ and τ. Shown in Table 2 (b), we find that the initialization of normalization parameters ρ and τ in the L2-tf-GTFC block has a considerable effect on the results. Initializing ρ to 0 tends to give better results. With a grid search, we find that the best setting is to assign ρ to 0 and τ to 1. It suggests that in the very early stage of the network training, it might be appropriate to discard the context guided time-frequency attention mechanism. The important thing is to learn a meaningful representation with the convolution stem firstly.

Table 2: *Empirical analysis for different components of our proposed blocks.*

(a) L2-c-GTFC channel adaptation operator

Operator	EER(%)	DCF
sigmoid	4.69	0.4434
1+ELU	3.85	0.3735
1+tanh	**3.13**	**0.3169**

(b) L2-tf-GTFC normalization parameters

(ρ, τ)	EER(%)	DCF
(0, 0)	3.52	0.3631
(0, 1)	**3.07**	**0.3207**
(1, 0)	4.47	0.4729
(1, 1)	4.48	0.4895

(c) L2-tf-GTFC group number

Group number	EER(%)	DCF
4	4.69	0.4201
8	3.07	**0.3207**
16	**3.01**	0.3451

(d) L2-tf-GTFC block position

Block position	EER(%)	DCF
after BN	**3.07**	**0.3207**
before BN	3.24	0.3718
before Conv	3.29	0.3885

Group number. We further investigate the number of groups in the L2-tf-GTFC transformation module in Table 2 (c). Too few groups may cause the diversity of feature representations limited. Using the group number 16, we obtain the best EER and the group number 8 for the best DCF values. However, too many groups may also result in a dimension reduction in the feature space, causing a weaker representation for each group response. We set the group number to 8 in all our experiments.

Block position. Inserting the proposed module after/before the Batch Norm layer, or before the convolution layer in the Residual basic block all improves the results, compared with the baseline ResNet-LDE and SE model. We only insert one proposed block after the Batch Norm layer in our experiments. The L2-tf-GTFC block only requires about 0.082M additional parameters, and therefore is very computationally efficient.

5. Conclusions

In this study, we proposed a global time-frequency context modeling framework and successfully applied it to the channel and time-frequency wise feature map recalibration. This model can capture long-range time-frequency dependency and channel variances. With this lightweight block, we can enhance the latent speaker representation and suppress possible distortions. The block was inserted after the last Batch Norm layer of each Residual basic block. The proposed method was evaluated on the VoxCeleb1 dataset, and it was shown to improve the ResNet-LDE and SE models in terms of both EER and DCF by a large margin. Additional analysis and ablation studies indicate that our proposed method can effectively improve the speaker representation learning by strengthening significant time-frequency and channel locations.

6. References

[1] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, "Audio surveillance of roads: A system for detecting anomalous sounds," *IEEE transactions on intelligent transportation systems*, vol. 17, no. 1, pp. 279–288, 2016.

[2] K. A. Lee, A. Larcher, H. Thai, B. Ma, and H. Li, "Joint application of speech and speaker recognition for automation and security in smart home," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[3] M. Crocco, M. Cristani, A. Trucco, and V. Murino, "Audio surveillance: A systematic review," *ACM Computing Surveys (CSUR)*, vol. 48, no. 4, p. 52, 2016.

[4] Q. Wang, H. Muckenhirn, K. Wilson, P. Sridhar, Z. Wu, J. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. L. Moreno, "Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking," *arXiv preprint arXiv:1810.04826*, 2018.

[5] K. Žmolíková, M. Delcroix, K. Kinoshita, T. Ochiai, T. Nakatani, L. Burget, and J. Černockỳ, "Speakerbeam: Speaker aware neural network for target speaker extraction in speech mixtures," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, pp. 800–814, 2019.

[6] P. Matějka, O. Glembek, F. Castaldo, M. J. Alam, O. Plchot, P. Kenny, L. Burget, and J. Černocky, "Full-covariance ubm and heavy-tailed plda in i-vector speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2011, pp. 4828–4831.

[7] J. H. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal processing magazine*, vol. 32, no. 6, pp. 74–99, 2015.

[8] P. Kenny, "Bayesian speaker verification with heavy-tailed priors." in *Odyssey*, 2010, p. 14.

[9] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision (ICCV), IEEE International Conference on*, 2007, pp. 1–8.

[10] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," *Acoustics, Speech and Sig- nal Processing (ICASSP), IEEE International Conference on*, 2018.

[11] D. Michelsanti and Z.-H. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," in *INTERSPEECH*, 2017.

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

[13] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 212–220.

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

[15] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.

[16] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," *Proc. Interspeech, 2018*, pp. 2252–2256.

[17] Y. Jung, Y. Kim, H. Lim, Y. Choi, and H. Kim, "Spatial pyramid encoding with convex length normalization for text-independent speaker verification," *Proc. Interspeech, 2019*, pp. 4030–4034.

[18] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5791–5795.

[19] H. Yu, Z.-H. Tan, Z. Ma, and J. Guo, "Adversarial network bottleneck features for noise robust speaker verification," in *INTERSPEECH*, 2017, pp. 1492–1496.

[20] W. Xia, J. Huang, and J. H. Hansen, "Cross-lingual text-independent speaker verification using unsupervised adversarial discriminative domain adaptation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5816–5820.

[21] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2016, pp. 5115–5119.

[22] H.-s. Heo, J.-w. Jung, I.-h. Yang, S.-h. Yoon, and H.-j. Yu, "Joint Training of Expanded End-to-End DNN for Text-Dependent Speaker Verification," in *INTERSPEECH*, 2017, pp. 1532–1536.

[23] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Proc. Odyssey The Speaker and Language Recognition Workshop*, 2018, pp. 74–81.

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[25] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 764–773.

[26] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le, "Attention augmented convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3286–3295.

[27] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "Gcnet: Non-local networks meet squeeze-excitation networks and beyond," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[28] Z. Yang, L. Zhu, Y. Wu, and Y. Yang, "Gated channel transformation for visual recognition," *arXiv preprint arXiv:1909.11519*, 2019.

[29] W. Xia and K. Koishida, "Sound event detection in multichannel audio using convolutional time-frequency-channel squeeze and excitation," in *INTERSPEECH*, 2019, pp. 3629–3633.

[30] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7794–7803.

[31] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015.

[32] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. Interspeech, 2017*, pp. 2616–2620.

[33] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," *arXiv preprint arXiv:1710.05941*, 2017.