CHANNEL INVARIANT SPEAKER EMBEDDING LEARNING WITH JOINT MULTI-TASK AND ADVERSARIAL TRAINING

Zhengyang Chen, Shuai Wang, Yanmin Qian†, Kai Yu

MoE Key Lab of Artificial Intelligence
SpeechLab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
{zhengyang.chen, feixiang121976, yanminqian, kai.yu}@sjtu.edu.cn

ABSTRACT

Using deep neural network to extract speaker embedding has significantly improved the speaker verification task. However, such embeddings are still vulnerable to channel variability. Previous works have used adversarial training to suppress channel information to extract channel-invariant embedding and achieved a significant improvement. Inspired by the successful joint multi-task and adversarial training with phonetic information for phonetic-invariant speaker embedding learning, in this paper, a similar methodology is developed to suppress the channel variability. By treating the recording devices or environments as the channel variability, two individual experiments are carried out, and consistent performance improvement is observed in both cases. The best performance is obtained by sequentially applying multi-task training at the statistics pooling layer and adversarial training at the embedding layer, achieving 10.77% and 9.37% relative improvements in terms of EER compared to the baselines, for the recording environments or devices level, respectively.

Index Terms— channel information, adversarial training, multi-task learning, text-dependent speaker verification

1. INTRODUCTION

Speaker verification (SV) aims to verify a user's claimed identity given his speech segment. Recently, deep neural network (DNN) based speaker embedding learning has become the dominant approach in this field. Researchers investigated different architectures [1, 2, 3, 4], different loss functions [5, 6, 7, 8, 9], and different model compensation methods [10, 11], which greatly boosted the performance of SV systems.

Despite the great success of deep learning technologies in the SV research field, it is still very challenging to build SV systems for real-world applications. It is well-known that speaker verification is more fragile than speech recognition with respect to the system robustness. To improve the robustness of an SV system, two sources of variability need to be addressed: speech content and channel variability. For the text-independent speaker verification, which requires two utterances from the same speaker with different speech content to be grouped into one category, it's important to deal with the phoneme variability in the speaker modeling process. For both real-world text-dependent and text-independent speaker verification tasks, where different devices and recording environments are used, the system performance will degrade dramatically due to such channel mismatch.

Recently, many works have been done to mitigate nuisance attributes in specific tasks. In speech area, adversarial training has been used in [12, 13, 14] to suppress speaker information when doing speech recognition, Hongji et al. uses domain adversarial training in [15] for the speaker anti-spoofing, Jiangfeng et al. and Zhong et al. have used GAN [16] and GRL strategy [17] respectively and done adversarial training to suppress the channel variability in SV task.

All works above are aimed to eliminate nuisance information in the primary task, while adversarial training is the most adopted method. However, it should be aware that there are two ways to take advantages of the available nuisance information, multi-task learning and adversarial learning. Our previous work [18] shows a possibility for combining both methods to better utilize phonetic information in the text-independent SV task: encouraging the phonetic information in the early frame-level layers and suppressing such information in the latter speaker embeddings layer.

In this paper, we follow a similar idea as in [18]. We assume that, even though we want to obtain a channel invariant embedding, the available channel information could be used for better generic acoustic feature learning in the shallow model layers and then be suppressed in the latter speaker embeddings. As verified by our experiments, applying multi-task learning before the embedding extraction layer and adversarial training on the speaker embeddings are both beneficial. When we combine the multi-task learning with adversarial training, two training strategies are designed, including the joint mode and progressive mode. Experiments are carried out on a wake-up word based TD-SV dataset. The best system achieves 10.77% and 9.37% relative improvements in terms of EER on multiple recording environments and devices respectively.

2. RELATED WORK

2.1. x-vector

Deep neural network (DNN) is well-known for its powerful modeling ability, and DNN based speaker embeddings have become the dominant speaker identity modeling methods. *x*-vector [2, 3] is a typical one and used by many researchers. In *x*-vector framework, a time-delay neural network (TDNN) is trained to discriminate different speakers in the training data. The frame-level spectral features

†Yanmin Qian is the corresponding author

This work has been supported by the National Key Research and Development Program of China (Grant No.2017YFB1302402). Experiments have been carried out on the PI supercomputer at Shanghai Jiao Tong University

Authors would like to thank AISPEECH. LTD for providing the wake-up word dataset

will first go through several frame-level layers, followed by a statistics pooling layer which aggregates the frame-level representation into a single segment-level representation. One or more embedding layers can be incorporated in the utterance-level layers to extract speaker embeddings. In our experiments, the *x*-vector extractor is used as the baseline and the backbone for our proposed framework.

2.2. Multi-task and adversarial training

Our previous work [18] has successfully combined multi-task and adversarial training to better use phonetic information in text-independent speaker verification task. The whole system is depicted in Figure 1.

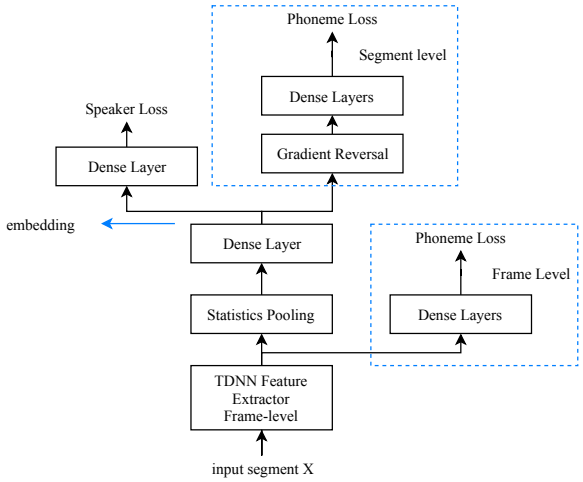

Fig. 1. Structure of combining multi-task and adversarial training to better utilize the phonetic knowledge in text-independent speaker verification task.

The main idea is to integrate the phonetic information which is beneficial to the generic feature learning at the shallow layer of the model and suppress the phoneme variability in the final speaker embedding layer. As shown in Figure 1, besides the primary speaker branch in the original x-vector framework [3], a frame-level multi-task phoneme branch and a segment-level adversarial phoneme branch are also included in our proposed architecture. Via the combination on supervision signals of these three branches, we observed an impressive performance improvement on the text-independent speaker verification task.

3. MODEL DESCRIPTION

Inspired by the success of our previous work [18], we would like to adopt a similar strategy to learn a channel-invariant speaker embedding, and the channel means the recording device and environment in our experiments. Using this new architecture, we want to enhance the channel variability for different training utterances at the shallow layer of the neural network and then suppress it in the latter embeddings layer, which finally can obtain better channel-invariant speaker embeddings.

3.1. Model Architecture

Different from the phonetic information, channel information resides in the segment level, thus, different from Fig.1, we will explore two

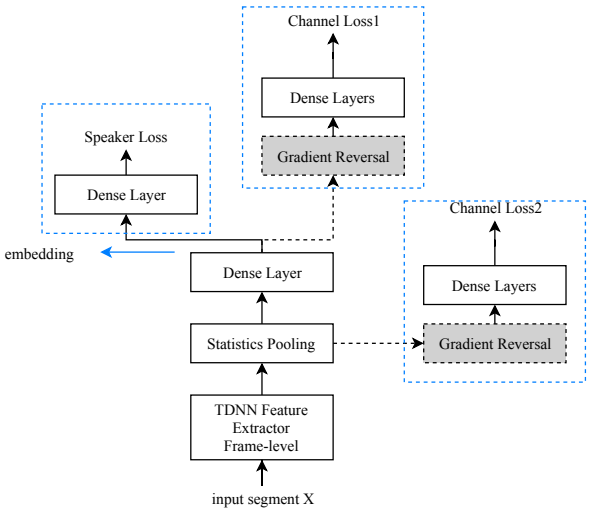

Fig. 2. The proposed structure of applying channel-level multi-task and adversarial training at the different positions of the model.

positions for multi-task and adversarial learning both at the segment level. The proposed architecture is shown in Fig.2. For the first type, instead of performing the multi-task/adversarial training on the output of the last frame-level layer, now we split the branches after pooling layer[1]. The second type is the same as in the Fig. 1, which is performed directly on the embedding layer. When the adversarial training is adopted, a gradient reversal layer (GRL) [19] will be inserted to the normal multitask branch to reverse the sign of computed gradients.

3.2. Loss Function

For an input segment $\mathbf{x}$ with speaker label y_s and channel label y_c, the total loss for the model optimization is composed of speaker loss $\mathcal{L}_s$ and channel loss $\mathcal{L}_{c1}$, $\mathcal{L}_{c2}$ as

$$\mathcal{L}_{total} = \mathcal{L}_s + \mathcal{L}_{c1} + \mathcal{L}_{c2} \tag{1}$$

$\mathcal{L}_{c1}$ and $\mathcal{L}_{c2}$ denote the channel losses whose branches are inserted at x-vector embedding and statistic layer respectively. Cross entropy will be used on the channel classification branch output $\mathbf{o}^i \in \mathbb{R}^l$, $i \in (1, 2)$, and l denotes the number of the channel classes.

$$\mathcal{L}_{ci} = -\log \frac{e^{\mathbf{o}^i_{y_c}}}{\sum_{j=1}^{l} e^{\mathbf{o}^i_j}} \tag{2}$$

For the speaker classification block, the key component of the model, we use the recently proposed additive angular margin loss [6, 5] as our primary speaker loss. The additive angular margin loss posts a more strict constraint which forces the similarity of the correct class to be greater than that of incorrect classes by a margin m.

$$L_s = -\log \frac{e^{s\cdot\cos(\theta_{y_s}+m)}}{e^{s\cdot\cos(\theta_{y_s}+m)} + \sum_{j=1,j\neq y_s}^{n} e^{s\cdot\left(\cos\theta_j\right)}} \tag{3}$$

[1]It's noticeable that the statistics pooling layer has no trainable parameters, so we expect similar effects to be learned by this architecture

where $\cos\theta_j = \mathbf{W}_j^T \mathbf{f}_{y_s}$, $\mathbf{f}_{y_s} \in \mathbb{R}^d$ is the normalized second dense layer output of the x-vector architecture. $\mathbf{W}_j \in \mathbb{R}^d$ denotes the normalized j-th column of the weight $\mathbf{W} \in \mathbb{R}^{d \times n}$. n denotes the number of the speaker classes. The additive angular margin loss also adds a scale parameter s, which helps the model converge faster. We select $m = 0.2$ and $s = 30$ for all our experiments.

3.3. Training strategy

Following the assumption that the channel information helps the generic feature learning at the shallow layers of the model, but it is not needed in the final speaker embeddings, we investigated two different training strategies to combine the multi-task learning and adversarial training, and obtain the final channel-invariant speaker embeddings.

- Joint Multitask-Adversarial training

In this strategy, the whole architecture and all the parameters are optimized simultaneously using both multi-task learning and adversarial training, and three loss functions are utilized for the model training. The multi-task learning will be applied at the statistics pooling layer and adversarial training will be applied at the embedding layer.

- Progressive Multitask-Adversarial training

In this case, we divide the model optimization into two stages, and the multi-task learning is firstly applied in the first stage with several training epochs. Then the multi-task learning branch is discarded and the adversarial training branch is added in the second stage with several training epochs.

4. EXPERIMENTAL SETUPS

4.1. Dataset

A wake-up word based dataset is used in this paper. The average duration of all the segments is around 1.0 second. Each person is demanded to repeat the wake-up word using a specific device in different environments. The dataset is well-annotated with device. But environment labels are not well-annotated across all the devices. We select 1.6M utterances from 2k different speakers as our training set. To generate more data for training, the utterances recorded in the quiet environment are augmented with noises from the MUSAN [20] dataset, resulting in a final training set with 5.2M utterances.

For the experiments with channel information, we consider the different recording devices and environments as available channel information here. The environment represents under which scenarios the recordings are collected, for example, quiet, office and car, etc. Besides, we also consider the augmented noises as different environment types. Experiments will be carried out regarding device or environment as channel information separately.

4.1.1. Data preparation using devices as channel information

For experiment considering device types as channel variability, all the training data described above are used for training, and there are 5 device types in total. Other 20543 utterances from 94 speakers not included in training set are used for enrollment and test. For each speaker, we select 4 clean utterances as the enrollment data, and the remaining utterances of this speaker are used to generated target trials. Besides, for each enrolled speaker, all utterances from other speakers are used to generate non-target test trails. Finally, we get 20167 target and 636798 non-target trials.

Table 1. The basic x-vector extractor configuration

Layer	Layer context	Input x output
frame1	[t - 2, t + 2]	200x100
frame2	{t - 2,t,t + 2}	300x100
frame3	{t - 2,t,t + 2}	300x100
frame4	{t}	100x100
frame5	{t}	100x375
stats pooling	[0, T)	375Tx750
segment6	{0}	750x100
segment7	{0}	750x100

4.1.2. Data preparation using environment as channel information

Because the environment labels are not well-annotated across all the devices, we only do this experiment on the dataset in some specific devices. We select data recorded by two devices from all the training data in section 4.1 to do experiments (we denote them as Device1 and Device2 dataset later). The number of environment types in Device1 and Device2 datasets are both 6. Device1 dataset consists of 352 speakers and 594583 utterances, and Device2 dataset consists of 512 speakers and 841450 utterances.

We use the same strategy as in section 4.1.1 to generate test trials. Finally, Device1 test set contains 35 speakers, 8732 target trials and 324888 non-target trials. Device2 test set contains 29 speakers, 5555 target trials and 158788 non-target trials. Results on two devices will be reported separately.

4.2. System configuration

The basic speaker embedding extractor is a x-vector [3] system with less parameters than the original one, and more detailed configuration could be found in Table 1. All architectures are implemented using Pytorch [21]. 40-dimensional Fbank features are extracted using Kaldi toolkit [22], with silent frames removed using an energy-based voice activity detector. The extracted embeddings are first length normalized and then a two-covariance PLDA [10, 22, 23] is used to calculated the scores.

4.2.1. Baseline System

Our baseline system is a normal x-vector with the architecture shown in Table 1, only the speaker classification loss will used as the optimization target. The margin of additive angular margin loss m will be linearly increased from 0.0 to 0.2 along the training iterations. We use SGD optimizer to optimize our network and set the momentum and learning rate to 0.9 and 0.0001 respectively.

4.2.2. Proposed System

The channel classification block consists of three linear layers with batchnorm [24] layer inserted. The dimensions of three linear layers are (inputdim) $\times$ (inputdim) $\times$ (channel class number). When multitask or adversarial head is added to our baseline network, the speaker and channel classification task will be trained jointly from scratch.

5. RESULTS AND DISCUSSION

5.1. Using environment labels as channel information

5.1.1. Exploring environment information at different positions of the model

The reasonable positions of the multi-task and adversarial branches are first explored. As shown in Table 2, adversarial training at the embedding layers improves the SV task performance, which is consistent with the findings in [16, 17]. Besides, multitask training at the statistics pooling layer obtains better results than the adversarial training, verifying our assumption in section 1 that the channel information could be helpful for the generic feature learning in the shallower model layer.

Table 2. Comparison of multitask or adversarial training results at different position of the model using environment information, STA-MT and STA-ADV denote multitask or adversarial training at the statistics pooling layer, while EMB-MT and EMB-ADV denote the related learning is performed at the embedding layer.

System	Dataset (EER(%))		
	Device1	Device2	Average
baseline	6.12	8.26	7.24
EMB-MT	6.61	7.93	7.27
STA-MT	**6.07**	**7.86**	**6.97**
EMB-ADV	**5.91**	**7.78**	**6.85**
STA-ADV	6.08	8.18	7.13

5.1.2. Joint and progressive Multitask-Adversarial training using environment information

Results in section 5.1.1 have shown that encouraging the environment information at the shallow layer of the model, i.e. statistics pooling layer, and suppressing it at the latter embedding layer, both can improve the model performance. Then we combined multi-task learning and adversarial training in the single architecture proposed in this paper. Two training strategies are performed and compared, and the results are shown in Table 3. It is observed that the proposed new architecture can get further improvement, and it is consistently better on all conditions. For the two training strategies, the PROGRESSIVE mode seems better than the JOINT mode. The best system can achieve 10.77% relative improvement compared to the baseline on average. Compared to JOINT mode, the statistic level multitask training branch only exists at the early stage of the whole training period in the PROGRESSIVE mode. Because the final goal of this task is to learn channel-invariant embedding, it is reasonable that PROGRESSIVE mode can perform better in which multitask training help the model learn better generic acoustic feature during the early training period and adversarial training help model focus on learning the channel-invariant embedding during the latter training period.

5.2. Using device labels as channel information

In this section we present the results obtained when the device labels are used as channel information. In this section, similar experiments to section 5.1 are carried out, while the device label of each utterance will be used as channel labels instead of the environment labels.

The results of doing multitask and adversarial training using device information are showed in Table 4. From the middle block of the table, we can observe that it's better that multitask and adversarial training should be inserted at the statistics pooling and embedding layer respectively, which is consistent with the conclusion in section 5.1.1. Furthermore, the same as the results in section 5.1.2, the architecture integrates both multi-task learning and adversarial training can obtain an additional improvement. For the two training strategies, the progressive mode is still slightly better, which achieves 9.37% relative improvement compared to the baseline.

Table 3. Comparison of two training strategies using environment information for the proposed architecture, JOINT denotes joint multitask-adversarial training mode and PROGRESSIVE denotes the progressive multitask-adversarial training mode for the proposed architecture.

System	Dataset (EER(%))		
	Device1	Device2	Average
baseline	6.12	8.26	7.24
EMB-ADV	5.91	7.78	6.85
STA-MT	6.07	7.86	6.97
JOINT	5.83	7.41	6.62
PROGRESSIVE	**5.57**	**7.36**	**6.46**

Table 4. Comparison of different systems using device information.

System	EER(%)
baseline	4.27
EMB-MT	4.12
STA-MT	4.09
EMB-ADV	4.10
STA-ADV	4.23
JOINT	3.93
PROGRESSIVE	**3.87**

From the above results, the consistent observations are obtained in both environment and device based channel invariant training. It demonstrates the effectiveness of our proposed new framework, and better channel-invariant speaker embeddings can be obtained with the new approach.

6. CONCLUSIONS

In this work, we propose a framework to combine the multitask and adversarial training at the different positions of the model to better utilize the channel information. Treating different devices or recording environments as channel labels, two independent experiments are carried out to verify the proposed models. Consistent performance improvement are observed in both experimental conditions. The results show that enhancing the channel information in the shallower layer of the model is helpful for the generic feature learning, while suppressing such information in the latter layer helps to learn the better channel-invariant speaker embeddings. Two training strategies are designed to optimize the whole model, and the new framework can get a better performance. The progressive learning mode is slightly better than the joint learning mode. The best systems achieves $\sim$10.0% relative improvements in terms of EER compared to the baselines, for both environments and devices levels.

7. REFERENCES

[1] Yuan Liu, Yanmin Qian, Nanxin Chen, Tianfan Fu, Ya Zhang, and Kai Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[2] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *Proc. Interspeech 2017*, pp. 999–1003, 2017.

[3] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[4] Yingke Zhu, Tom Ko, David Snyder, Brian Mak, and Daniel Povey, "Self-attentive speaker embeddings for text-independent speaker verification.," in *Interspeech*, 2018, pp. 3573–3577.

[5] Xu Xiang, Shuai Wang, Houjun Huang, Yanmin Qian, and Kai Yu, "Margin matters: Towards more discriminative deep neural network embeddings for speaker recognition," *arXiv preprint arXiv:1906.07317*, 2019.

[6] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos 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.

[7] Jixuan Wang, Kuan-Chieh Wang, Marc T Law, Frank Rudzicz, and Michael Brudno, "Centroid-based deep metric learning for speaker recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3652–3656.

[8] Chunlei Zhang and Kazuhito Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances.," in *Interspeech*, 2017, pp. 1487–1491.

[9] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[10] Simon JD Prince and James H Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

[11] Claudio Vair, Daniele Colibro, Fabio Castaldo, Emanuele Dalmasso, and Pietro Laface, "Channel factors compensation in model and feature domain for speaker recognition," in *2006 IEEE Odyssey-The Speaker and Language Recognition Workshop*. IEEE, 2006, pp. 1–6.

[12] Taira Tsuchiya, Naohiro Tawara, Testuji Ogawa, and Tetsunori Kobayashi, "Speaker invariant feature extraction for zero-resource languages with adversarial learning," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2381–2385.

[13] Zhong Meng, Jinyu Li, Zhuo Chen, Yang Zhao, Vadim Mazalov, Yifan Gang, and Biing-Hwang Juang, "Speaker-invariant training via adversarial learning," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5969–5973.

[14] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, et al., "English conversational telephone speech recognition by humans and machines," *arXiv preprint arXiv:1703.02136*, 2017.

[15] Hongji Wang, Heinrich Dinkel, Shuai Wang, Yanmin Qian, and Kai Yu, "Cross-domain replay spoofing attack detection using domain adversarial training," *Proc. Interspeech 2019*, pp. 2938–2942, 2019.

[16] Jianfeng Zhou, Tao Jiang, Lin Li, Qingyang Hong, Zhe Wang, and Bingyin Xia, "Training multi-task adversarial network for extracting noise-robust speaker embedding," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6196–6200.

[17] Zhong Meng, Yong Zhao, Jinyu Li, and Yifan Gong, "Adversarial speaker verification," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6216–6220.

[18] Shuai Wang, Johan Rohdin, Lukáš Burget, Oldřich Plchot, Yanmin Qian, Kai Yu, and Jan Černockỳ, "On the usage of phonetic information for text-independent speaker embedding extraction," *Proc. Interspeech 2019*, pp. 1148–1152, 2019.

[19] Yaroslav Ganin and Victor Lempitsky, "Unsupervised domain adaptation by backpropagation," *arXiv preprint arXiv:1409.7495*, 2014.

[20] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[21] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, "Automatic differentiation in pytorch," 2017.

[22] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number CONF.

[23] Aleksandr Sizov, Kong Aik Lee, and Tomi Kinnunen, "Unifying probabilistic linear discriminant analysis variants in biometric authentication," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, 2014, pp. 464–475.

[24] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.