ID R&D team submission description for SASV Challenge 2022

Alexander Alenin[1], *Nikita Torgashov*[1], *Anton Okhotnikov*[1],
Rostislav Makarov[1], *Ivan Yakovlev*[1]

[1]ID R&D Inc., New York, USA

{alenin,torgashov,ohotnikov,makarov,yakovlev}@idrnd.net

Abstract

In SASV challenge ID R&D team pursued two goals. The first included building the most precise fusion of systems minimizing the SASV EER. The second goal attempted to build a small single-model system providing a comparable to a big fusion system metrics. This report describes the details of our challenge submission. In particular, we cover up the training strategies for independently trained ASV and CM systems and present the results of a linear fusion of all scores together with QMFs. Our best fusion achieves **0.136**% EER on SASV-2022 evaluation set, while the smallest single-model system with 11.6M parameters achieves **0.223**% EER.

Index Terms: Speaker recognition, Voice anti-spoofing, ASVSpoof2019, SASV Challenge 2022

1. Introduction

A Spoofing Aware Speaker Verification (SASV) challenge's aim is to promote the development of Automatic Speaker Verification (ASV) systems that are able to reject impostor access attempts and be robust against spoofing attacks. Participants are required to build a framework optimising the ASV systems operating in tandem with countermeasures (CM) systems.

In order to develop an integrated SASV solution researchers are encouraged to investigate the possibilities to train a single-model system operating with spoofing and verification embeddings and scores, or trained in a multi-task end-to-end fashion combining ASV and CM losses to minimize the SASV EER.

This report is structured in the following way. In Section 2 we present the models architectures, used loss function properties and input features setup. Section 3 gives an overview of available datasets, training augmentations and models' training sequence. We also present our fusion scheme and used Quality Measurement Functions (QMFs) in this section. Sections 4 and 5 contain Results and Conclusions respectively.

2. System Setup

2.1. Input features

For training, fixed-length 2-second audio segments were used. We randomly cropped segments from each utterance in the training dataset. Then, 80-dimensional Mel filter bank log-energies with a 25 ms frame length and 10 ms step were extracted with an FFT size of 512 over the 20-7600 Hz frequency limits. After feature extraction, we subtract the mean along the time axis. To test the models, we used 8-second input segments.

2.2. Architectures

All the systems in our submission are based on the residual neural networks [1], which made a breakthrough in the task of image classification by using very deep models, and recently have been efficiently applied to the speaker recognition task [2], [3]. A ResNet-34 architecture described in [3] was selected as our baseline system. Since the deeper models usually show a performance improvement in various tasks, we decided to apply some modifications to the baseline architecture. In particular, to increase the capacity of models we have run a series of experiments and optimised such hyperparameters as a number of residual blocks and a number of filters in each residual block. In the end, 2 modifications of the ResNet-34 model with 48, 100 hidden layers were selected.

Detailed architectures are shown in the Table 1 and results of verification testing on VoxCeleb1-test dataset presented in the Table 2, where C_{FA} and C_{Miss} equals to 1, and P_{target} equals to 0.01 for MinDCF metric.

2.3. Subnetwork Approach

For detection of spoofing attacks we have trained a small subnetwork on top of verification backbone, in the same way as it was described in [4].

2.4. Loss function

All our models were trained using the Additive Margin Softmax (AM-Softmax) loss function [5]. The main aim of this loss is to reduce the interclass variance by introducing the margin penalty to the target class logit. AM-Softmax showed itself as an effective loss function in face recognition and has been successfully applied to speaker recognition task as well. According to [3], the margin value was set to 0.3 and a scale value was set to 40.

3. Experiments

3.1. Datasets

For speaker recognition (ASV) systems training VoxCeleb2-dev (5994 speakers) dataset [6] was used, and a training subset of ASVSpoof2019 Logical Access (LA) [7] dataset was used for implementing the voice anti-spoofing (CM) systems. Eval subset of ASVSpoof2019 LA dataset was used to evaluate the systems performance, and dev subset of ASVSpoof2019 LA was used for development purposes, such as best training epoch model weights selection for anti-spoofing model and optimization of linear fusion weights for combined SASV system.

Table 1: *Models architectures*

Layer name	**Output (C $\times$ F $\times$ T)**	**ResNet-48**	**ResNet-100**
Conv2D	C $\times$ 80 $\times$ T	96, 3$\times$3, stride=1	128, 3$\times$3, stride=1
ResBlock-1	C $\times$ 80 $\times$ T	$\begin{bmatrix}3\times 3, 96\\ 3\times 3, 96\end{bmatrix}\times 6$	$\begin{bmatrix}3\times 3, 128\\ 3\times 3, 128\end{bmatrix}\times 6$
ResBlock-2	128 $\times$ 40 $\times$ T/2	$\begin{bmatrix}3\times 3, 128\\ 3\times 3, 128\end{bmatrix}\times 8$	$\begin{bmatrix}3\times 3, 128\\ 3\times 3, 128\end{bmatrix}\times 16$
ResBlock-3	C $\times$ 20 $\times$ T/4	$\begin{bmatrix}3\times 3, 160\\ 3\times 3, 160\end{bmatrix}\times 6$	$\begin{bmatrix}3\times 3, 256\\ 3\times 3, 256\end{bmatrix}\times 24$
ResBlock-4	256 $\times$ 10 $\times$ T/8	$\begin{bmatrix}3\times 3, 256\\ 3\times 3, 256\end{bmatrix}\times 3$	$\begin{bmatrix}3\times 3, 256\\ 3\times 3, 256\end{bmatrix}\times 3$
Flatten (C, F)	2560 $\times$ T/8	—	
Pooling	5120	StatsPooling	
Dense	256	—	
AM-Softmax	Num. of speakers	—	

Table 2: *Results on VoxCeleb1-O Standard protocol*

Model	**EER** [%]	**MinDCF**
ResNet-48	1.09	0.102
ResNet-100	0.90	0.069

3.2. Data augmentation

To augment verification training datasets we used MUSAN corpus [8] and real room impulse responses (RIRs) database [9]. We applied various on-the-fly augmentations during the training process. For each training utterance we applied 6 different augmentation strategies:

- **Music**: A single music file was randomly selected from MUSAN and added to the original audio (5-15dB SNR). The duration of additive noise was matched to the duration of the original signal.
- **Noise**: Randomly selected noise from MUSAN was added to the original recording (0-15dB SNR).
- **Speech**: Three to seven speakers were randomly picked, summed together, and then added to the original signal (13-20dB SNR).
- **Reverb**: Artificially reverberated a signal via convolution with real RIRs.
- **Speed**: Artificially change speed of file (via FFT resampling). Speed is chosen randomly from [0.9, 1.0, 1.1]. Each speed used its own set of classes. The number of target speaker classes has been tripled.
- **Spectral augmentation**: We also applied SpecAugment[10] to the input log Mel-spectrograms and randomly masked 0 to 5 frames in the time domain and 0 to 10 frequency bins.

3.3. Implementation details

All the described models were trained using TensorFlow 2 framework [11] and SGD optimizer with momentum (set to 0.9). To train very deep ResNet architectures faster, Google Cloud TPUs were used. To train the models we used the following two-stage scheme:

3.3.1. Verification backbone

We have trained a speaker recognition models for 50 epoch, where each epoch consists of 5000 steps, with a batch size of 256. To form a batch, we sampled 256 unique speakers and took a single utterance for each of them. During training we updated the learning rate and margin of AM-Softmax loss function. Learning rate linearly increased from minimum (1e-4) to maximum value (0.1), while margin was equal to zero for the first 3 epochs. Then we fixed learning rate at the maximum value and linearly increased the value of margin from zero to it's maximum value (0.3) for the following 10 epochs. For the rest of training, we fixed the margin of AM-Softmax loss and applied an exponential decay to the learning rate every 4 epochs with a rate of 0.5. We have also applied L2-norm regularization of 1e-5 for all model's weights except the AM-Softmax head, for which we increased the regularization value to 1e-4.

3.3.2. Anti-spoofing subnetwork

On the second stage, we freezed the verification backbone and trained an anti-spoofing subnetwork on top of it using the AM-Softmax loss function. The size of input was extended to 4 seconds while training and no augmentations were applied to the training data.

Table 3: *SV, SPF and SASV protocols EER (%) for the SASV 2022 development and evaluation partitions*
E - enrollment utterances, V - verification utterance

Group	**Name**	**Description**	**SV-EER** [%]		**SPF-EER** [%]		**SASV-EER** [%]	
			Dev	*Eval*	*Dev*	*Eval*	*Dev*	*Eval*
Challenge Baseline	ECAPA-asv	ECAPA-TDNN ASV score [12]	1.88	1.63	20.30	30.75	17.38	23.83
	AASIST-cm	AASIST CM score [13]	46.02	49.24	0.07	0.67	15.85	24.37
	Baseline2	Ensemble of ECAPA-TDNN and ASIST	12.87	11.48	0.13	0.78	4.85	**6.37**
ASV	r48-asv	ResNet48 ASV cosine score (E vs V)	0.000	0.151	15.230	24.417	12.263	18.123
	r100-asv	ResNet100 ASV cosine score (E vs V)	0.051	0.111	14.676	22.942	11.921	17.277
CM	r48-cm	ResNet48 CM cosine score (E vs V)	49.265	48.394	0.067	1.400	15.556	24.224
	r48-cm-cls	ResNet48 CM classification score (V)	36.124	50.045	0.135	0.520	13.274	25.381
Single Model System	SF1	r48-asv + r48-cm	0.205	0.522	0.146	0.916	0.199	0.743
	SF2	+ r48-cm-cls	0.068	0.377	0.067	0.431	0.068	0.406
	SF3	++ ResNet48 ASV ASNorm score (E vs V)	0.077	0.278	0.076	0.433	0.076	0.339
	SF4	+++ QMF: ResNet48 CM class. score (E)	0.068	0.238	0.067	0.279	0.068	0.260
	SF5	++++ QMFs: speech lengths (E and V)	0.068	0.186	0.067	0.245	0.068	**0.223**
Models Ensemble	F1	SF2 + r100-asv	0.068	0.279	0.072	0.448	0.068	0.354
	F2	+ AASIST-cm	0.128	0.261	0.007	0.226	0.052	0.242
	F3	++ ASNorm for ResNet48 and ResNet100	0.137	0.258	0.016	0.224	0.062	0.242
	F4	+++ QMF: ResNet48 CM class. score (E)	0.009	0.150	0.007	0.172	0.007	0.153
	Submission	++++ QMFs: speech lengths (E and V)	0.000	0.105	0.004	0.168	0.004	**0.136**

3.4. Fusion description

The output of our integrated SASV system includes fusion of cosine similarity scoring of backbone and anti-spoofing subnetwork embeddings and an anti-spoofing subnetwork spoofing probability output score as follows:

1. ASV cosine similarity score between mean enrollment model backbone embedding and a verification file backbone embedding
2. CM cosine similarity score between mean enrollment model anti-spoofing subnetwork embedding and a verification file anti-spoofing subnetwork embedding
3. CM spoof probability of a verification file from the 2-class head of anti-spoofing subnetwork
4. Same as 1 with additionaly applied ASNorm backend

ASNorm cohort size is 1200 random files from ASVSpoof2019 LA train set with a *top N = 300* trials used to estimate mean and std of scores distribution for normalization.

3.4.1. Quality Measurement Functions

To further improve the target metrics, QMF [4] correcting terms were used in addition to ASV and CM scores to shift each trial. QMF values were extracted from enrollment and verification files, and the following factors were used in a final submission:

- **Enrollment model speech length** - sum of speech lengths across all enrollment files in a model
- **Verification file speech length**
- **Enrollment model *inverted CM score*** - mean value of inverted sign CM system output for all enrollment files in a model. This factor is considered as a feature describing the mean quality of enrollment model.

File speech length was extracted using a simple energy-based VAD from Kaldi toolkit [14].

3.4.2. Fusion scheme

The SASV system output is a linear fusion of ASV and CM scores and QMF values for enrollment and verification files. The optimal weights are estimated using COBYLA toolkit [15] minimizing the EER metric on SASV development set.

3.5. Evaluation

Evaluation of systems' performance is done using Equal Error Rate (EER) metric, corresponding to the operating point of equal False Acceptance and False Rejection error rates.

4. Results

The testing results on SASV dev and eval data are presented for all challenge protocols in the table 3. This table reflects our ASV backbones quality as well CM anti-spoofing subnetworks quality for ResNet48 and ResNet100 models. From this table we can see how our single model system SASV-EER was improved by using a fusion of ASV and CM embedding-based scores (SF1). Moreover, extending such fusion with CM spoof output probability scores and various QMFs (SF5) lead us to a significant x3 EER reduction on SASV protocol for both dev and eval subsets. In the result, the SF5 system consists of ResNet-48 model only (verification backbone with a small anti-spoofing subnetwork) with QMFs in fusion and reaches **0.223**% SASV-EER on eval set, while being relatively compact (11.6M parameters) and fast.

In a similar to the single model system fusion fashion we have built an ensemble system containing both ResNet48 and ResNet100 models together with an open-sourced AASIST [13] CM scores. By exploiting the previously showed fusion improvement strategy with various QMFs we were able to achieve the final metric of **0.136**% SASV-EER on a challenge evaluation set with our ensemble submission.

5. Conclusions

In our paper we showed how to train the voice anti-spoofing subnetwork on top of a precise verification backbone. Additionally, we proposed a novel scoring strategy for SASV protocols, which includes the usage of embedding-based similarity scores and anti-spoofing classification head output spoof probability. Furthermore, we have found out that the usage of QMF factors could be very profitable, especially if applied to both enrollment and verification utterances.

6. References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[2] J. Thienpondt, B. Desplanques, and K. Demuynck, "Tackling the score shift in cross-lingual speaker verification by exploiting language information," *arXiv preprint arXiv:2110.09150*, 2021.

[3] D. Garcia-Romero, G. Sell, and A. Mccree, "MagNetO: X-vector Magnitude Estimation Network plus Offset for Improved Speaker Recognition," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 1–8. [Online]. Available: http://dx.doi.org/10.21437/Odyssey.2020-1

[4] A. Alenin, R. Makarov, N. Torgashov, I. Shigabeev, and K. Simonchik, "The id r&d system description for short-duration speaker verification challenge 2021," in *Interspeech 2021*, 2021, pp. 2297–2301.

[5] 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.

[6] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[7] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. Kinnunen, and K. A. Lee, "Asvspoof 2019: Future horizons in spoofed and fake audio detection," *arXiv preprint arXiv:1904.05441*, 2019.

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

[9] I. Szoke, M. Skacel, L. Mosner, J. Paliesek, and J. Cernocky, "Building and evaluation of a real room impulse response dataset," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, p. 863–876, Aug 2019. [Online]. Available: http://dx.doi.org/10.1109/JSTSP.2019.2917582

[10] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Interspeech 2019*, Sep 2019. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

[11] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 2016, pp. 265–283.

[12] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[13] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, and N. Evans, "Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks," *arXiv preprint arXiv:2110.01200*, 2021.

[14] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.

[15] M. J. Powell, "A view of algorithms for optimization without derivatives," *Mathematics Today-Bulletin of the Institute of Mathematics and its Applications*, vol. 43, no. 5, pp. 170–174, 2007.