Contents

Annotation

The automatic speaker verification systems have shown excellent results working with target and non-target real speech, but they still perform poorly with synthesized and replayed speech. In the last couple of years some joint automatic speaker verification and countermeasure systems have been developed. However, they aimed to work only with imposter and text-to-speech or voice conversion attacks. In this work, we implement a combined system, which performs well with non-target real speech and replay attacks. For that reason some fusion and cascade methods are reviewed, as well as different countermeasure systems.

Аннотация

Автоматические системы верификации говорящего показывают отличные результаты в работе с целевой и нецелевой настоящей речью, но до сих пор остаются уязвимыми к синтезированной и проигранной речи. В последние пару лет были созданы объединенные автоматические системы верификации говорящего и системы контрмер фальсификациям. Тем не менее, они все еще рассчитаны на работу с нецелевой настоящей и синтезированной речью. В данной работе создана единая система, которая отражает как нецелевую речь, так и атаки, созданные посредством проигрывания записанной речи владельца устройства. С этой целью изучаются каскадные методы и методы слияния отдельных систем, а также сами системы борьбы с фальсификациями.

Keywords

Deep Learning, Automatic Speaker Verification Systems, Countermeasure Systems, Spoofing-Aware Speaker Verification

1 Introduction

During the last few years the use of the automatic speaker verification (ASV) systems has increased significantly. Such models define whether the speaker is the owner of the system or another real person. Despite the fact that modern systems show excellent results with equal-error rate (EER) around 1% [8] on the VoxCeleb dataset [23], even state-of-the-art models still stay vulnerable to some kinds of spoofing attacks. In general, spoofing attacks [9] include non-target genuine speech, audio replay and synthesized speech (text-to-speech or voice conversion). While imposter attacks do not pose a threat to modern biometric systems anymore, replayed and synthesized speech can still trick even best ASV models. Thus, more and more special algorithms, known as countermeasures (CMs), have been designed recently to distinguish whether the speech is bona fide or not. The development of the countermeasure systems is also inspired by the ASVspoof challenge series [25, 35].

The ASVspoof competition typically consists of two scenarios: logical (LA) and physical(PA) access, which stands for synthesized and replayed speech respectively. In the first case, top models have performed with the equal-error rate of 0.22% on the LA partition in 2019 [25] and 1.32% on the LA in 2021 [35]. In case of physical access, the winning solution [4] in 2019 showed the EER of 0.52%. Though the best performance of suggested solutions seems quite successful, the quality of an ASV system, which was the original subject of interest, can remain poor even in tandem with a strong CM system. Hence, the Spoofing-Aware Speaker Verification (SASV) challenge [16] was introduced. Led by this initiative, some state-of-the-art joint systems [1, 33, 5] had been developed, but their work is limited to text-to-speech and voice conversion attacks, not replay ones.

The main goal of this work is to develop a spoofing-aware speaker verification system, which will show comparatively good performance with recorded speech rather than synthesized. This includes reviewing different ASV and CM systems and finding the best way to integrate them into a combined solution. Overall, there are three popular methods of creating a SASV solution. They include the cascade approach [33], which means using one model after another; the fusion method [1], also known as the ensemble; and simply implementing end-to-end model [30]. Some strategies for the cascade and fusion approaches are reviewed further.

In this work, we took the best approaches for synthesized speech detection and applied them in physical access scenario. We implemented some top solutions of the SASV challenge [16] and compared them to two baselines, which had been offered by the organizers (see Section 3 for more details). Overall, despite the fact that our CM subsystem is not state-of-the-art solution, we have achieved quite a reliable quality on the evaluation set, which also significantly outperformed all of the baselines.

The rest of the paper organises as follows: related works, which include the descriptions of modern SASV systems, are reviewed in Section 2; experimental setup with the information about the dataset, metrics and our combined systems can be found in Section 3; results are given in Section 4, and, finally, the main findings are summarized in Section 5.

2 Related Works

2.1 Countermeasure systems

A biometric system can be described by two its components: front-end and back-end. Here front-end characterises the way we work with the input: we can use raw audio features, spectrogram, cepstral coefficients or self-supervised embeddings. Back-end, to put it in another way, is simply the architecture of the model: it can be Gaussian mixture model or deep neural network; we are looking for the second option. Such models were designed, for example, in the physical access scenario of the ASVspoof 2019, 2021 challenges [25, 35].

In the physical access scenario of 2019 competition [25] recorded speech consisted of 9 various sets of parameters [32], including different room sizes, T60 reverberation time and talker-to-ASV distance. Two baseline solutions, suggested by organizers [25], were based on 512-component Gaussian mixture model back-end binary classifier trained using expectation-maximization (EM) algorithm. They used constant Q cepstral coefficients [22] and linear frequency cepstral coefficients [21] as front-end, and ended up with 11.04% and 13.54% equal-error rates respectively. At the same time, five best single systems [4, 3, 20, 19] showed EER between 0.52% and 1.66%. Most of them were based upon ResNet18 or ResNet34 architectures [12] and used diverse front-ends, while primary systems [4, 20, 19, 3] (fusion of ResNet-based models or light convolutional neural networks [34] with various front-ends) performed slightly better, with EER starting from 0.39%.

2.2 Spoofing-Aware Speaker Verification Challenge

Moving on to joint ASV and CM systems, we will take a look at the results of the Spoofing-Aware Speaker Verification Challenge [16]. The VoxCeleb2 [6] database was used for the training of ASV sub-systems, the ASVspoof 2019 logical access scenario database [32] was used for the training of CM sub-systems. Though the spoofing attacks consisted of genuine non-target and synthesized speech only, there are still some interesting ideas about integrating two systems into one solution.

The winning solution [1] was based on the subnetwork - a novel technique for training neural networks. Thus, the backbone for the speaker verification task was firstly trained. Then a small model on the top of a frozen backbone was trained to solve the anti-spoofing problem. The SASV output included fusion of cosine similarity scoring of backbone and anti-spoofing subnetwork embeddings and an anti-spoofing subnetwork spoofing probability score. Speaking of other top solutions, some teams built a modified cascade framework [33], while others developed a latent space in which both speaker identity and spoofing artefacts can be captured using ASV and CM sub-system embeddings in order to map these embeddings into a single SASV one using a DNN with condition layers [30]. And, of course, there were differences in the training strategies too.

2.3 Fusion Strategies

To start with, fusion techniques are important even in one-task scenario. As it was mentioned above, best systems often use a couple of countermeasure subsystems. In [28] it is showed why constant Q cepstral coefficients [22] are effective in detecting some attacks but less effective in detecting others. While working with synthesized attacks, authors suggest that the artefacts that distinguish spoofed speech from real speech might lay in specific sub-bands. They later come up with an idea that there is no single CQCC front-end that can perform well for all types of spoofing attacks; different attacks algorithms produce artefacts at different parts of the spectrum and these can only be detected reliably when the front-end emphasises information in the relevant frequency bands. This result explains why fusion is vitally important to generalisation, i.e. system's ability to perform well in the face of various spoofing attacks, even if some of them are absolutely unknown to our model.

Furthermore, in [29] it is suggested that usual fusion methods (for instance, average or weighted sum) can be inappropriate in some cases. In particular, linear combinations of mostly non-informative systems may dilute the success from other systems. Because of that, the authors consider four different fusion approaches: a support vector machine (SVM), multinomial logistic regression, traditional linear fusion and a Gaussian mixture model (GMM)-based approach to fusion. Experiments are performed with the logical access partition of the ASVspoof 2019 database [25]. The best result with a minimum normalised tandem detection cost function (t-DCF [18]) of 0.0740 is showed by non-linear GMM method, followed by non-linear SVM approach with a t-DCF of 0.0748. At the same time, linear fusion and multinomial logistic regression end up with a t-DCF of 0.0911 and 0.1182 respectively, which means that a non-linear approach is definitely better for the fusion of sub-band countermeasure systems scores.

Moving on to the fusion between independent ASV and CM systems, generally, there are two strategies for their fusion. Thus, some methods offer performing fusion in the embedding space [5, 27]. In this case, we build a model that operates upon embeddings from different latent spaces. Prior works, for example, [27, 11], propose designing a deep neural network aimed to optimize both ASV and CM embeddings in order to produce a single SASV score. Another option to consider is performing fusion in the score level, almost as it have been suggested in previous works for one system. In [36] two methods of score-level fusion are defined. They are based on the probabilistic framework, proposed by the authors:

$$P(y^t = 1|x^e_{ASV}, x^t_{ASV}, x^t_{CM}) = P(y^t_{ASV} = 1|x^e_{ASV}, x^t_{ASV})P(y^t_{CM} = 1|x^t_{CM}). \tag{1}$$

In this formulation $x^e_{ASV}, x^t_{ASV}, x^t_{CM}$ are the embeddings for the enrollment and test utterances computed by ASV and CM correspondingly. By definition, $y^t = 1$ – test utterance is target – if and only if $y^t_{ASV} = 1$ and $y^t_{CM} = 1$. Thus, assuming conditional independence between ASV and CM systems, we get the expression above.

The first fusion strategy, direct inference, involves using pre-trained ASV and CM subsystems. As their scores do not initially fit to the probabilistic framework, it is suggested to use a sigmoid function on the output score of CM system. The final decision is represented as:

$$S_{SASV} = \sigma(S_{CM}) \times f(S_{ASV}) \tag{2}$$

Here $S_{SASV}, S_{CM}, S_{ASV}$ are the SASV, CM and ASV scores respectively. Three options for the function f are proposed: a linear mapping $f(s) = \frac{(s+1)}{2}$, a sigmoid function and a trained with logistic regression calibration function.

The second strategy, fine-tuning, can not work in the conditional independence assumption. An alternative derivation of the posterior probability in this case:

$$P(y^t = 1|x^e_{ASV}, x^t_{ASV}, x^t_{CM}) = P(y^t_{ASV} = 1|x^e_{ASV}, x^t_{ASV})P(y^t_{CM} = 1|y^t_{ASV}, x^t_{CM}) \tag{3}$$

The decision score also looks like eq. (2), but in this case f can be either linear mapping or sigmoid function. In this method we fine-tune the FC layer of the CM with fixed ASV score. Moreover, the model directly optimizes the joint score, and spoof and non-target utterances share the same negative labels.

All proposed strategies performed significantly better on LA partition of the ASVspoof19 database than baselines, provided by the SASV organizers. In direct inference method configuration with the linear mapping was the best, followed by the sigmoid function. Fine-tuning approach showed evaluation EER of 1.54% and 1.53% with linear and sigmoid functions respectively, while the result of direct inference with linear function is 1.68%. Therefore, even such simple methods of building a fusion can be quite successful.

2.4 Cascade Approach

The next strategy we consider is the sequential cascade approach. Therefore, we implement the DKU-OPPO system [33], designed for the SASV challenge [16], as it is the best solution with such technique. This system basically consists of two modules: ASV and CM. The first module produces a hard decision (0 or 1) based on a threshold, which is tuned according to the EER on the development set. If this decision is positive, the second module generates a raw score, which makes up the output score. If the first module's decision is negative, then the minimal score of the second module on the development set is used as the final score for the audio (see Figure 2.1 for an example). Therefore, there are two ways of building a cascade: ASV and CM subsystems can be used in any order.

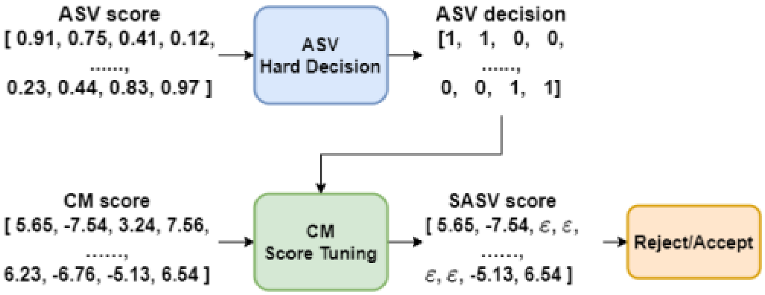

Figure 2.1: [33] Cascade system, where an ASV subsystem is followed by a CM subsystem. ε stands for the minimal CM score on the development set.

The authors consider different ASV and CM subsystems, as well as different training strategies; however, the best EER of 0.209% on the evaluation set of the LA partition of the ASVspoof2021 challenge [32] is shown by the ASV-CM cascade.

2.5 FiLM

Applying feature-wise linear modulation (FiLM [26]) to an input speaker embedding is an another successful method used in building a joint system. It was a part of the HYU team submission [5] for the SASV challenge [16], and it finished on the 3rd place. This system produces the spoofing-aware speaker embeddings (SASE), and the SASV score can be obtained simply by calculating cosine similarity with the enrolment utterance.

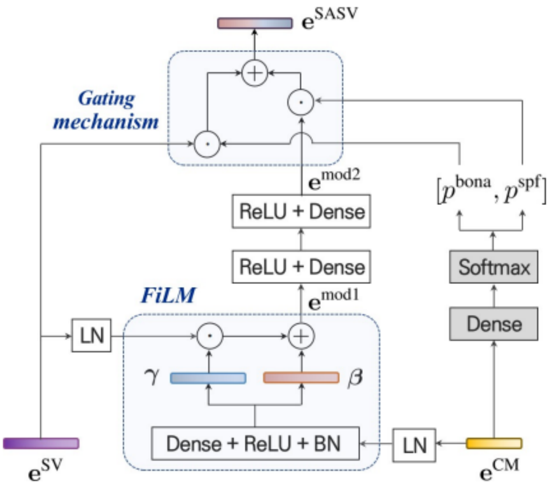

Figure 2.2: [5] Block diagram of HYU solution.

The main idea of the method is to build a neural network-based backend that operates with the speaker embeddings using the FiLM [26] technique (see Figure 2.2 for visualization). Firstly, a CM embedding of the utterance is used to calculate the FiLM parameters:

$$[\gamma^T, \beta^T]^T = \text{BN}\left(\text{ReLU}\left(W_1^T \text{LN}\left(e^{CM}\right) + b_1\right)\right) \in \mathbb{R}^{2d_{sv}}. \tag{4}$$

Here $\gamma, \beta \in \mathbb{R}^{d_{sv}}$ are the scale and shift parameters of the FiLM method; $W_1 \in \mathbb{R}^{d_{cm} \times 2d_{sv}}, b_1 \in \mathbb{R}^{2d_{sv}}$ are the trainable weight and bias; e^{CM} is a CM embedding, d_{sv} and d_{cm} denote the dimensions of the ASV and CM embeddings respectively. BN, LN and ReLU are the classic batch normalization [14], layer normalization [2] and ReLU [24].

After obtaining the FiLM parameters with eq.(4), we use these variables to reform ASV embeddings as follows:

$$e^{mod1} = \gamma \odot \text{LN}\left(e^{SV}\right) + \beta \in \mathbb{R}^{d_{sv}}, \tag{5}$$

where e^{SV} represents speaker embedding, $\odot$ denotes element-wise multiplication.

Thirdly, the modified embedding from eq.(5) is processed through two fully connected layers:

$$e^{mod2} = W_3^T \text{ReLU}\left(W_2^T \text{ReLU}\left(e^{mod1}\right) + b_2\right) + b_3 \in \mathbb{R}^{d_{sv}}, \tag{6}$$

where $W_2, W_3 \in \mathbb{R}^{d_{sv} \times d_{sv}}, b_2, b_3 \in \mathbb{R}^{d_{sv}}$ are the trainable weights and biases.

Finally, the resulting speaker embedding is obtained as follows:

$$e^{SASV} = p^{spf} e^{mod2} + p^{bona} e^{SV}, \tag{7}$$

where e^{SV} is the SV embedding, e^{mod2} is a modified embedding from eq.(6); p^{spf}, p^{bona} are the probabilities of the utterance being spoofed or bona fide respectively. The probabilities can be obtained from the pretrained CM model.

As a result, from eq.(7) we get a new embedding e^{SASV} with the following characteristics: if the speech is predicted by CM to be bona fide, our embedding will be similar to the input speaker embedding; on the contrary, if the utterance is more likely to be spoofed, we get a CM-conditioned embedding. So, in order to get the SASV score, the cosine similarity between the pair of reformed speaker embeddings is calculated.

3 Experimental Setup

3.1 ASV subsystem

An unofficial re-implementation [7] of the ECAPA-TDNN [8], which is the efficient state-of-the-art speaker embedding extractor, was used as the ASV subsystem. It operates on the Mel-Spectrogram; includes SE-Res2Blocks, which is Res2Net [10] block followed by Squeeze-Excitation block [13]. This implementation is available as open source[1].

3.2 CM subsystem

A classic light convolution network [34] with Max-Feature-Map activation operating on LFCC features [21] was used as a CM subsystem. This model is described in [31], and it had been offered as a baseline for the ASVspoof2021 challenge, scoring the EER of 44.77% on the PA partition of the ASVspoof2019 dataset. Implementation of the model can be found in the offical repository of the ASVspoof2021[2].

3.3 Data Usage and Evaluation Metrics

The ASVspoof19 [32] dataset for the physical access scenario was used. As it was mentioned before, the audio can be described by three different acoustic parameters: the room size, T60

[1] ECAPA-TDNN at https://github.com/TaoRuijie/ECAPATDNN (visited on May 9, 2024)

[2] LFCC-LCNN at https://github.com/asvspoof-challenge/2021/tree/main/PA/Baseline-LFCC-LCNN (visited on May 9, 2024)

reverberation time and talker-to-ASV distance - each parameter has 3 options. Moreover, there are also 9 various replay configurations, which are characterized by the attacker-to-speaker recording distance and the loudspeaker quality. Both training and development data include short audios from 8 males and 12 females; the number of bona fide utterances is similar for these subsets (5400 utterances), while the number of spoof utterances comprises 48600 and 24300 for train and dev respectively. What is worth mentioning, the evaluation set is disjoint in terms of speakers, and though the data is generated with the same categories as training and development subsets, the sets of characteristics is different. It means that despite the fact that the categories are known from the train, the particular impulse responses used to simulate genuine and spoofed speech are unknown. Under these circumstances reliable performance can be achieved only by systems which generalise well to unfamiliar data.

Equal-error rate (EER) was used as a metric like it was in the ASVspoof and SASV challenges. EER is equal to the false acceptance rate with the threshold s such that false acceptance rate at s is equal to the false rejection rate at s. In this formulation, our metric works as «the lower the better».

Furthermore, as we are evaluating a joint system and our «negative» audio consists of non-target real speech and spoofed speech, we compute three EERs as well as it was offered in the SASV challenge. Hence, the EER_{SV} denotes the EER between target speech as positive and nontarget real speech as negative; EER_{SPF} takes target speech as positive and spoofed speech as negative; finally, EER_{SASV} counts both nontarget real and spoofed speech.

3.4 Joint Systems Setup

3.4.1 Baseline

A simple score-level fusion was used as a baseline: it produced the output score for the test utterance by summing ASV and CM scores. Such system was not expected to perform well as the scores were not normalized, however, another baseline uses the sigmoid function to preliminarily map the scores into a $[0, 1]$ interval.

3.4.2 Score Fusion

The next system in our work was also based on the score-level fusion: we considered the direct inference approach, which had been defined previously. The official implementation of the original paper [36] for the SASV challenge is available as open source[3]. As it have been designed for

[3] Probabilistic Fusion Framework at https://github.com/yzyouzhang/SASV_PR (visited on May 9, 2024)

dealing with the logical access scenario, it uses another countermeasure subsystem – AASIST [15] – which is state-of-the-art CM for the synthesized speech. However, we adopted this method for the physical access scenario by using our CM subsystem to extract the embeddings, which were later passed to the framework as well as the embeddings from the ASV subsystem.

3.4.3 Cascade

The sequential cascaded system, defined previously, was also implemented. Our implementation allows to choose whether you want the ASV or the CM subsystem to be the first. The system includes one hyperparameter, the threshold s, based on which the first module of the cascade makes a hard decision. The hyperparameter was tuned according to the EER on the development set, and our best hyperparameters are:

$$\begin{cases} s = 0.596, & \text{for the ASV-CM cascade} \\ s = 0.411, & \text{for the CM-ASV cascade} \end{cases}$$

3.4.4 FiLM

Another joint system we had implemeted was a HYU team solution [5] based on the FiLM [26] technique, which was described before. We trained this system using the binary cross entropy loss, Adam optimizer [17] with weight_decay= 0.001 and LambdaLR scheduler with the following lambda function:

$$\mathrm{lr}_{k+1} = \frac{1}{1 + \mathrm{decay} * \mathrm{lr}_k}, \tag{8}$$

where lr_k, lr_{k+1} denote the learning rates on the k-th and k+1-th steps respectively, starting lr was 0.0003 and the parameter decay was initially set to 0.0001.

4 Results

Table 4.1 represents our results. Here *ecapa-tdnn* and *lfcc-lcnn* are our ASV and CM subsystems respectively; *baseline* denotes the simple summing of ASV and CM scores. Suffix *linear* denotes the following mapping, proposed in [36]:

$$f(s) = \frac{s+1}{2}. \tag{9}$$

Suffix *sigmoid* represents the sigmoid function used to map the SASV scores into a $[0, 1]$ interval. *pr-linear, pr-sigmoid* systems are the score fusion based solutions with the corresponding mapping functions, *casc-asv-cm* and *casc-cm-asv* denotes the sequential cascaded systems with ASV and CM as the first module respectively. Finally, *film* is the FiLM-based joint system.

Table 4.1: SASV performance of different joint systems

	Dev			Eval		
	EER$_{SV}$(%)	EER$_{SPF}$(%)	**EER$_{\mathbf{SASV}}$(%)**	EER$_{SV}$(%)	EER$_{SPF}$(%)	**EER$_{\mathbf{SASV}}$(%)**
ecapa-tdnn	**4.148**	39.091	29.926	**5.224**	38.636	25.309
lfcc-lcnn	55.085	2.778	29.468	51.211	2.596	35.123
baseline	51.741	2.724	27.866	47.450	2.577	32.863
baseline-linear	10.111	**2.593**	8.593	10.394	**2.561**	9.064
baseline-sigmoid	11.815	2.695	10.296	12.407	2.577	11.235
pr-linear	9.751	2.708	8.148	9.915	2.577	8.704
pr-sigmoid	10.963	2.741	9.444	11.242	2.585	9.977
casc-asv-cm	4.466	4.778	**4.710**	6.523	5.646	**5.795**
casc-cm-asv	6.889	4.050	5.704	7.299	3.008	6.651
film	4.644	8.058	6.698	5.509	9.415	7.207

As it can be seen from the table 4.1, the lowest EER_{SV} both on the development and the evaluation sets is achieved by single ECAPA-TDNN, which is quite predictable as this model is state-of-the-art speaker verification system by itself. On the contrary, it is interesting that the lowest EER_{SPF} is shown not by LFCC-LCNN, which is our CM subsystem, but by the baseline with the linear mapping function. However, it can be seen that all three baselines and the probabilistic approach based solutions showed EER_{SPF} which is comparable to the LFCC-LCNN result.

Speaking in terms of common result, which means looking at the EER_{SASV}, we can see quite anticipated figures. For example, even the simpliest baseline outperforms a single CM subsystem; moreover, two baselines with normalization perform significantly better, reducing the EER by more than three times. Moving on to more complicated solutions, two probabilistic-based systems show the EER of 8.704% and 9.977% on the evaluation set, which is quite good, especially taking into the account the brevity and simplicity of their idea. More interestingly, it can be seen that the linear mapping function shows better results than the sigmoid one both for the baseline and the probabilistic systems. FiLM-based solution ended up with the EER_{SASV} of 6.698% on the development data and 7.207% on the evaluation data, which is even better than all the previous methods. Anyway, the sequential cascade approach shows the lowest EER_{SASV}, in case of cascade-asv-cm it equals to 4.710% and 5.795% on the development and evaluation subsets respectively. Furthermore, the smallest difference between the EER_{SV} and EER_{SPF} is shown by the cascaded systems, which means that such joint solutions are more balanced while dealing both with the non-target real speech and replayed speech.

5 Conclusion

To conclude, the aims of this work have been achieved. We considered the best approaches for building a unified biometric system working with synthesized speech and adopted them for the replayed speech. Overall, the behaviour of these solutions in the physical access scenario is quite similar to their behaviour in case of logical access, which allows us to build a joint system with a reliable performance. What is also worth mentioning, these results can be improved by working with a stronger CM subsystem: obviously, our LFCC-LCNN is not one the best countermeasure systems. Nevertheless, our combined systems were able to significantly outperform the baselines.

References

[1] Alexander Alenin, Nikita Torgashov, Anton Okhotnikov, Rostislav Makarov, and Ivan Yakovlev. "A Subnetwork Approach for Spoofing Aware Speaker Verification". In: *Proc. Interspeech 2022*. 2022, pp. 2888–2892.

[2] Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. "Layer Normalization". In: *arXiv preprint arXiv:1607.06450* (2016).

[3] Weicheng Cai, Haiwei Wu, Danwei Cai, and Ming Li. "The DKU Replay Detection System for the ASVspoof 2019 Challenge: On Data Augmentation, Feature Representation, Classification, and Fusion". In: *arXiv preprint, arXiv:1907.02663* (2019).

[4] Xingliang Cheng, Mingxing Xu, and Thomas Fang Zheng. "Replay detection using CQT-based modified group delay feature and ResNeWt network in ASVspoof 2019". In: *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. 2019, pp. 540–545.

[5] Jeong-Hwan Choi, Joon-Young Yang, Ye-Rin Jeoung, and Joon-Hyuk Chang. "HYU Submission for the SASV Challenge 2022: Reforming Speaker Embeddings with Spoofing-Aware Conditioning". In: *Proc. Interspeech 2022*. 2022, pp. 2873–2877.

[6] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. "VoxCeleb2: Deep Speaker Recognition". In: *Interspeech 2018*. ISCA, 2018.

[7] Rohan Kumar Das, Ruijie Tao, and Haizhou Li. "HLT-NUS SUBMISSION FOR 2020 NIST Conversational Telephone Speech SRE". In: *arXiv preprint arXiv:2111.06671* (2021).

[8] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification". In: *Interspeech 2020*. ISCA, 2020.

[9] Nicholas Evans, Tomi Kinnunen, and Junichi Yamagishi. "Spoofing and countermeasures for automatic speaker verification". In: *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH* (Aug. 2013).

[10] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. "Res2Net: A New Multi-Scale Backbone Architecture". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.2 (Feb. 2021), pp. 652–662. ISSN: 1939-3539.

[11] Alejandro Gomez-Alanis, Jose A. Gonzalez-Lopez, S. Pavankumar Dubagunta, Antonio M. Peinado, and Mathew Magimai.-Doss. “On Joint Optimization of Automatic Speaker Verification and Anti-Spoofing in the Embedding Space”. In: *IEEE Transactions on Information Forensics and Security* 16 (2021), pp. 1579–1593.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *arXiv preprint, arXiv:1512.03385* (2015).

[13] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. “Squeeze-and-Excitation Networks”. In: *arXiv preprint arXiv:1709.01507* (2019).

[14] Sergey Ioffe and Christian Szegedy. “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In: *arXiv preprint arXiv:1502.03167* (2015).

[15] Jee-weon Jung, Hee-Soo Heo, Hemlata Tak, Hye-jin Shim, Joon Son Chung, Bong-Jin Lee, Ha-Jin Yu, and Nicholas Evans. “AASIST: Audio Anti-Spoofing using Integrated Spectro-Temporal Graph Attention Networks”. In: *arXiv preprint, arXiv:2110.01200* (2021).

[16] Jee-weon Jung, Hemlata Tak, Hye-jin Shim, Hee-Soo Heo, Bong-Jin Lee, Soo-Whan Chung, Ha-Jin Yu, Nicholas Evans, and Tomi Kinnunen. “SASV 2022: The First Spoofing-Aware Speaker Verification Challenge”. In: *Proc. Interspeech (submitted)*. 2022.

[17] Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: *arXiv preprint arXiv:1412.6980* (2017).

[18] Tomi H. Kinnunen, Kong-Aik Lee, Héctor Delgado, Nicholas W. D. Evans, Massimiliano Todisco, Md. Sahidullah, Junichi Yamagishi, and Douglas A. Reynolds. “t-DCF: a Detection Cost Function for the Tandem Assessment of Spoofing Countermeasures and Automatic Speaker Verification”. In: *arXiv preprint, arXiv:1804.09618* (2018).

[19] Cheng-I Lai, Nanxin Chen, Jesús Villalba, and Najim Dehak. “ASSERT: Anti-Spoofing with Squeeze-Excitation and Residual neTworks”. In: *Interspeech*. 2019.

[20] Galina Lavrentyeva, Sergey Novoselov, Andzhukaev Tseren, Marina Volkova, Artem Gorlanov, and Alexander Kozlov. “STC Antispoofing Systems for the ASVspoof2019 Challenge”. In: *Interspeech*. 2019.

[21] T. Kinnunen M. Sahidullah and C. Hanilci. “A comparison of features for synthetic speech detection”. In: *Proc. Interspeech* (2015), pp. 2087–2091.

[22] H. Delgado M. Todisco and N. Evans. “Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification”. In: *Computer Speech Language* 45 (2017), pp. 516–535.

[23] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. "Voxceleb: Large-scale speaker verification in the wild". In: *Computer Science and Language* (2019).

[24] Vinod Nair and Geoffrey E. Hinton. "Rectified Linear Units Improve Restricted Boltzmann Machines". In: *International Conference on Machine Learning.* 2010.

[25] Andreas Nautsch, Xin Wang, Nicholas Evans, Tomi Kinnunen, Ville Vestman, Massimiliano Todisco, Héctor Delgado, Md Sahidullah, Junichi Yamagishi, and Kong Aik Lee. "ASVspoof 2019: Spoofing Countermeasures for the Detection of Synthesized, Converted and Replayed Speech". In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* PP (Feb. 2021), pp. 1–1.

[26] Ethan Perez, Florian Strub, Harm Vries, Vincent Dumoulin, and Aaron Courville. "FiLM: Visual Reasoning with a General Conditioning Layer". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32 (Sept. 2017).

[27] Hye-jin Shim, Jee-weon Jung, Ju-ho Kim, and Ha-jin Yu. "Integrated Replay Spoofing-Aware Text-Independent Speaker Verification". In: *Applied Sciences* 10.18 (Sept. 2020), p. 6292. ISSN: 2076-3417.

[28] Hemlata Tak, Jose Patino, Andreas Nautsch, Nicholas W. D. Evans, and Massimiliano Todisco. "An explainability study of the constant Q cepstral coefficient spoofing countermeasure for automatic speaker verification". In: *The Speaker and Language Recognition Workshop.* 2020.

[29] Hemlata Tak, Jose Patino, Andreas Nautsch, Nicholas W. D. Evans, and Massimiliano Todisco. "Spoofing Attack Detection using the Non-linear Fusion of Sub-band Classifiers". In: *Interspeech.* 2020.

[30] Zhongwei Teng, Quchen Fu, Jules White, Maria E Powell, and Douglas C Schmidt. "SA-SASV: An End-to-End Spoof-Aggregated Spoofing-Aware Speaker Verification System". In: *arXiv preprint, arXiv:2203.06517* (2022).

[31] Xin Wang and Junichi Yamagishi. "A Comparative Study on Recent Neural Spoofing Countermeasures for Synthetic Speech Detection". In: *Interspeech.* 2021.

[32] Xin Wang, Junichi Yamagishi, Massimiliano Todisco, Hector Delgado, Andreas Nautsch, Nicholas Evans, Md Sahidullah, Ville Vestman, Tomi Kinnunen, Kong Aik Lee, Lauri Juvela, Paavo Alku, Yu-Huai Peng, Hsin-Te Hwang, Yu Tsao, Hsin-Min Wang, Sebastien Le Maguer, Markus Becker, Fergus Henderson, Rob Clark, Yu Zhang, Quan Wang, Ye Jia, Kai Onuma, Koji Mushika, Takashi Kaneda, Yuan Jiang, Li-Juan Liu, Yi-Chiao Wu, Wen-Chin

Huang, Tomoki Toda, Kou Tanaka, Hirokazu Kameoka, Ingmar Steiner, Driss Matrouf, Jean-Francois Bonastre, Avashna Govender, Srikanth Ronanki, Jing-Xuan Zhang, and Zhen-Hua Ling. "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech". In: *arXiv preprint, arXiv:1911.01601* (2020).

[33] Xingming Wang, Xiaoyi Qin, Yikang Wang, Yunfei Xu, and Ming Li. "The DKU-OPPO System for the 2022 Spoofing-Aware Speaker Verification Challenge". In: *arXiv preprint, arXiv:2207.07510* (2022).

[34] Zhenzong Wu, Rohan Kumar Das, Jichen Yang, and Haizhou Li. "Light Convolutional Neural Network with Feature Genuinization for Detection of Synthetic Speech Attacks". In: *arXiv preprint, arXiv:2009.09637* (2020).

[35] Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, and Héctor Delgado. "ASVspoof 2021: accelerating progress in spoofed and deepfake speech detection". In: Sept. 2021, pp. 47–54.

[36] You Zhang, Ge Zhu, and Zhiyao Duan. "A Probabilistic Fusion Framework for Spoofing Aware Speaker Verification". In: *arXiv preprint, arXiv:2202.05253* (2022).