# Pindrop Labs' Submission to the ASVspoof 2021 Challenge

*Tianxiang Chen, Elie Khoury, Kedar Phatak, Ganesh Sivaraman*

Pindrop, Atlanta, GA, USA

{tchen,ekhoury,kphatak,gsivaraman}@pindrop.com

## Abstract

Voice spoofing has become a great threat to automatic speaker verification (ASV) systems due to the rapid development of the speech synthesis and voice conversion techniques. How to effectively detect these attacks has become a crucial need to those systems. The ASVspoof 2021 challenge provides a unique opportunity to foster the development and evaluation of new techniques to detect logical access (LA), physical access (PA) and Deepfake (DF) attacks covering a wide range of techniques and audio conditions. The Pindrop Lab participated to both the LA and DF detection tracks. Our submissions to the challenge consist of a cascade of an embedding extractor and a backend classifier. Instead of focusing on an extensive feature engineering and complex score fusion methods, we focus on improving the generalization of the embedding extractor model and the backend classifier model. We use log filter banks as the acoustic features in all our systems. Different pooling methods and loss functions are studied in this work. Additionally, we investigated the effectiveness of stochastic weight averaging, further improved the robustness of the spoofing detection system. Overall, three different variants of the same system have been submitted to the challenge. They all achieved a very competitive performance on both LA and DF tracks, and their combination achieved a min-tDCF of 0.2608 on the LA track and an EER of 16.05% on the DF track.

## 1. Introduction

Automatic Speaker Verification (ASV) has been widely adopted in many human-machine interfaces. The accuracy of the ASV system has improved greatly in the past decades due to the help of deep learning algorithms. Meanwhile, the deep learning based text-to-speech synthesis (TTS) and voice conversion (VC) techniques are also able to generate extremely realistic speech utterances. TTS and VC techniques like WaveNet [1], Deep Voice [2] and Tacotron [3] greatly enhanced the quality of the voice-spoofed utterances. These spoofed utterances are often indistinguishable to human ears and are able to deceive state-of-the-art ASV systems. Thus, the detection of these voice spoofing attacks has drawn great attention in the research community and the technology industry.

To benchmark the progress of research in voice spoofing detection and foster the research efforts, ASVspoof[1] challenge releases a series of spoofing datasets. In 2019, the ASVspoof [6] releases two datasets, physical access (PA) and logical access (LA). PA dataset focuses on replay attacks and LA dataset refers to synthesized speech. LA dataset was largely based on detecting deep learning based spoofing techniques, and it was primarily focusing on evaluating of the generalization of the spoofing detection model. In total, it includes seventeen different TTS and VC techniques, but only seven of them are in the training and development set. During the ASVspoof 2019 challenge, many submissions were focused on investigating different low level spectro-temporal features [7, 8, 9, 10, 11, 12] and ensemble-based approaches.

[1]http://www.asvspoof.org

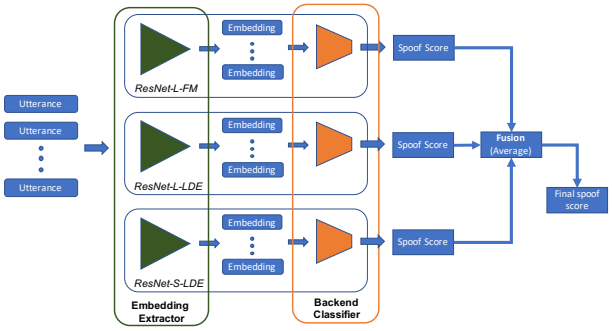


Figure 1: *High level framework of the spoofing detection system. Each system contains two major components, embedding extractor and backend classifier. Our final submission is an average score fusion of three systems.*

In ASVspoof 2021 [13], the challenge has further included more data to simulate more practical and realistic scenarios of different spoofing attacks. There are three sub-challenges: physical access (PA), logical access (LA) and deepfake detection (DF). The PA dataset contains *real* replayed speech and a small portion of *simulated* replayed speech. For the Logical Access (LA) dataset, while the training and development data remain the same as ASVspoof 2019, various codec and channel transmission effects are added to the evaluation data. This is aimed to simulate the telephony scenarios and evaluate the robustness of the spoofing detection model against different channel effects. The challenge has also further extended the LA track to general speech Deepfake detection (DF). Deepfake detection deals with detecting synthesized voice in any audio recording. The speech Deepfake detection task involves different audio compression techniques such as mp3 and m4a along with additional spoofing techniques. This Deepfake detection task aims to evaluate the spoofing detection system against different unknown conditions. Therefore, the detection systems for both LA and DF tracks need to be robust to unseen attacks and audio compression techniques.

This paper presents the Pindrop Labs' submissions to the LA and DF track, and introduces a novel spoofing detection system. Our submissions were among the top performing systems in the full evaluation sets on both LA and DF tracks. In total, we have trained three systems. The first system is proposed in [14], which is a ResNet based spoofing detection system trained using large margin cosine loss. The second system is an extension



10.21437/ASVSPOOF.2021-14

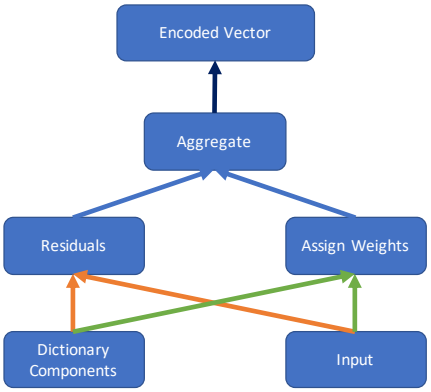


Figure 2: *Learnable dictionary pooling layer forward diagram.*

of the first system, we use a novel learnable dictionary encoding (LDE) [15] layer to replace the mean and standard deviation pooling layer. The third system also uses the LDE pooling layer but is trained using Softmax activation in the output layer and the cross-entropy loss function. All systems contain two main components, embedding extractor and backend classifier. Figure 1 shows the frame work of our spoof detection system. The final submissions to both LA and DF tracks are the fusion of the three spoofing detection systems.

This paper is organized as follows: Section 2 describes the datasets used to train the proposed spoofing detection systems. Section 3 details the three spoofing detection systems and the fine-tuning strategy. Section 4 presents the results on LA and DF evaluation datasets. Section 5 concludes this paper.

## 2. Datasets

We use the ASVspoof 2019 official LA train and development datasets to train and evaluate our systems. Various data augmentation methods are performed on the training dataset to increase the amount of data and robustness of the models. the ASVspoof 2019 and 2021 datasets are presented in Sec 2.1 and Sec 2.2. The data augmentation technique is introduced in Sec 2.3.

### 2.1. ASVspoof 2019 Challenge Dataset

ASVspoof 2019 [6] logical access (LA) dataset comprises seventeen different text-to-speech (TTS) and voice conversion (VC) techniques, from traditional vocoders to the recent state-of-the-art neural vocoders. The spoofing techniques are divided into two groups, six as *known* techniques, and eleven *unknown* techniques. The train and development sets have six *known* spoofing techniques, while the evaluation set contains eleven *unknown* spoofing techniques.

In this work, only the training and development sets are used for developing the spoofing detection systems.

### 2.2. ASVspoof 2021 LA & DF Dataset

The ASVspoof 2021 LA track is aiming to evaluate the robustness of the spoofing detection model across different channels. Although, the spoofing techniques used in this dataset are the same as in 2019, multiple codec and transmission effects are added to the audio samples. Both bonafide and spoofed samples are transmitted through either a public switched telephone networks or a voice over internet protocol. After passing through different networks, all audio samples are resampled to 16 kHz.

Deepfake detection track is an extension of the LA track. In contrast to the LA track, DF track is focusing on evaluating the spoofing detection systems across different audio compressions. It represents detecting spoofed audios on social media or other internet platforms, where the audio compression techniques and audio qualities are largely different. The compression algorithms include mp3, m4a and other unknown techniques.

### 2.3. Data Augmentation

Three different types of data augmentation are applied to the train dataset in this work. The first type augmentation is similar to the augmentation used in [14], we added two types of distortion to the clean samples: reverberation, and background noise. For the reverberation effect, random room impulse responses (RIR) were chosen from publicly available RIR datasets[2]. For the background noises, we used three types of noises - music, babble and *freesound*[3]. The *freesound* noises were the general noise files from the MUSAN corpus which consisted of files collected from *freesound* and *soundbible*. For babble and *freesound* noises, we added the background noise files to the clean audio and then reverberated the mixture using a randomly selected RIR. The noises were added with a random SNR between 5 dB to 20 dB.

Second type of augmentation is to simulate audio compression effects. All clean audio samples were also passed through audio compression. The compression algorithms include mp3 and m4a.

Finally, the third type of augmentation was applied to add codec transmission effects. The training dataset were logically played through Twilio's Voice service[4] and recorded at the receiver's end. The resulting dataset has VoIP channel characteristics and has reduced bandwidth from 16 kHz to 8 kHz sampling rate. Twilio's default OPUS codec[5] was used for encoding and decoding audio.

## 3. Methodology

In this section, we first describe the low-level features and the preprocessing techniques used during training (Sec 3.1). Then, we present the architectures of the embedding extraction models in the spoofing detection systems. (Sec 3.2). Finally, we describe the back-end classifiers used in all systems (Sec 3.5).

### 3.1. Features and Preprocessing

Features used in this work are linear filter banks (LFBs) in this work. LFBs are a direct compressed version of the short-time Fourier transforms (SFT) with a linearly spaced filter bank, and thus more adequate for lower computational cost, and has lower risk of overfitting at training time. We use 60-dimensional LFBs extracted on 30 ms windows with an overlap of 10 ms. Mean and variance normalization was performed per utterance during training and testing.

Online frequency masking is applied during training to randomly drop out a contiguous band of frequency channels $v = [f_0 + f]$. The value $f$ is chosen from a uniform distribution from 0 to the parameter $F$, where defines the maximum

[2]http://www.openslr.org/28/

[3]https://freesound.org/

[4]https://support.twilio.com/hc/en-us/articles/360010317333-Recording-Incoming-Twilio-Voice-Calls

[5]https://www.opus-codec.org/

| Layer | Filter size | # filters | Stride | Output size |
|---|---|---|---|---|
| Freq Masking | - | - | - | $200 \times 60$ |
| Conv1 | $3 \times 3$ | 64 | $1 \times 2$ | $200 \times 30$ |
| MaxPooling | $1 \times 3$ | - | $1 \times 4$ | $200 \times 7$ |
| Res block 1 | $3 \times 3$ | 64 | $1 \times 1$ | $200 \times 7$ |
| Res block 2 | $3 \times 3$ | 128 | $1 \times 1$ | $200 \times 4$ |
| Res block 3 | $3 \times 3$ | 256 | $1 \times 1$ | $200 \times 2$ |
| Res block 4 | $3 \times 3$ | 512 | $1 \times 1$ | $200 \times 1$ |
| Mean and std | - | - | - | 1024 |
| FC1 | - | - | - | 512 |
| FC2 | - | - | - | 256 |
| LMCL output | - | - | - | 2 |

Table 1: *This table details the architecture of the embedding extractor in ResNet-L-FM system. All convolutional and fully connect layers are followed by batch normalization and Selu activation layer. The outputs from FC2 layer are used as feature embeddings.*

number of frequency channels to be masked. Then, $f_0$ is chosen between $[0, v - f_0]$. After creating the frequency mask, an element wise multiplication operation is done between original LFBs and the frequency mask, so that the feature of the selected frequency channel can be set to zero.

### 3.2. Embedding Extractors

For this challenge, three different embedding extractors are used for logical access and Deepfake detection. All three embedding extractors are a modified version of the Residual neural network [16] . Residual neural network architecture has shown a great generalization ability in many classification tasks. It allows us to train an extensively deeper network to achieve more compelling results. In the following sections, all three networks will be explained in detail.

#### *3.2.1. ResNet-L-FM system*

The spoof embedding extractor used in the first system is the ResNet18-L-FM model described in [14]. As shown in Table 1, this residual network is a variant of the ResNet-18 [16] where the global average pooling layer is replaced by mean and standard deviation pooling layers [17]. Before the input layer, a random frequency masking augmentation is applied to randomly mask a range of frequency bins. Large margin cosine loss (LMCL) was used during training to increase the generalization ability of the model. The model is trained to classify the audio recordings into two classes: *bonafide* and *spoofed*. The spoof embedding is the output of the second fully connect layer, and its dimension is 256.

#### *3.2.2. ResNet-L-LDE system*

The ResNet-L-LDE is an evolution of the first ResNet18-L-FM system described above. Similar to the ResNet-L-FM system, the ResNet-L-LDE system also uses the ResNet-18 architecture as an encoder. However, it replaces the mean and standard deviation pooling layer with a learnable dictionary encoding (LDE) layer [15]. The LDE pooling layer assumes the frame level representations after the ResNet encoder are distributed in $C$ clusters. It is motivated by GMM super-vector encoding procedure, it learns the encoding parameters and the inherent dictionary in a full supervised manner. Figure 2 illustrates the forward diagram of LDE layer. In this work, we set the number of components equals to 16 and the hidden feature dimension for each components to 256. Thus, the output size of the LDE pooling layer is 4,096. The ResNet-L-LDE system is also trained using both frequency masking augmentation and LMCL.

The ResNet-S-LDE system is also trained using LDE pooling layer. The architecture is the same as the ResNet-L-LDE model, but is trained using Softmax output and cross-entropy loss. After training the model, we extract the outputs from the LDE pooling layer on the full utterances. The pooling output embeddings are then used to fine-tune the last two fully connected layers. LMCL is used to train the network in the fine-tuning stage. Because the model is trained on fixed-length two-second audio chunks, this fine-tuning stage consists of adapting the embedding on the full utterance with variable length.

### 3.3. Backend Classifier

After extracting the feature embedding, the embeddings are then fed into the backend classifier to classify whether it is spoofed or bonafide audio. The backend classifier is the same architecture proposed in [14]. It is a shallow neural network that consists of one fully connected layer (FC) with 256 neurons, followed by batch normalization layer, Relu activation, dropout layer with dropout rate of 50%, and one Softmax output layer. In order to further increase the generalization ability, we use the stochastic weight averaging [18] (SWA) procedure to train the backend classifier. SWA is a novel Deepfake in the weights space. It averages the weights of the models at different training epochs. By averaging the weights, it can create similar properties compare to traditional ensemble method. Thus, it can provide better generalization ability. The SWA algorithm can be defined as:

$$W_{SWA} \leftarrow \frac{W_{SWA} \times n_{models} + W}{n_{models} + 1} \tag{1}$$

Where $W_{SWA}$ stores the running average of the weights, $W$ is the model weights at the end of each training epoch. The model $W_{SWA}$ is the final model used in the inference stage.

## 4. Experiments

In this section, we report the results of our systems on the official ASVspoof 2021 evaluation set. Two key performance metrics are used to evaluate the systems. The first is EER that represents the point where false rejection rate (FRR) equals the false acceptance rate (FAR). In this case, the negative class is spoofing. The second metric is the minimum normalized tandem detection cost function (t-DCF) [19]. The t-DCF is defined as follows:

$$t\text{-}DCF^{min}_{norm} = min\left\{\beta P^{cm}_{miss}(s) + P^{cm}_{fa}(s)\right\} \tag{2}$$

where $\beta$ depends on application parameters (priors, costs) and ASV performance, $P^{cm}_{miss}(s)$ and $P^{cm}_{fa}(s)$ are the countermeasure system miss and false alarm rate at threshold $s$. In contrast to the EER computation, the negative class in t-DCF computation is either spoofing or zero-effort impostor. Therefore, the ASV scores should be provided.

Table 4 shows the results on the LA track for all our different systems. The *ResNet-L-LDE* system has the best min t-DCF and EER compared to other systems. Table 2 shows the detailed results based on min-tDCF for all different conditions and spoofing algorithms. It clearly shows that our system is robust to most of the channel effects. Because there is only one type

| Algorithm \ Codec | LA-C1 | LA-C2 | LA-C3 | LA-C4 | LA-C5 | LA-C6 | LA-C7 | Pooled |
|---|---|---|---|---|---|---|---|---|
| A07 | 0.0643 | 0.1513 | 0.2278 | 0.0774 | 0.1585 | 0.2191 | 0.0629 | 0.1588 |
| A08 | 0.0674 | 0.1837 | 0.2860 | 0.0948 | 0.1947 | 0.2735 | 0.0672 | 0.1918 |
| A09 | 0.2433 | 0.3359 | 0.3752 | 0.2717 | 0.3311 | 0.3933 | 0.2694 | 0.3732 |
| A10 | 0.0728 | 0.1613 | 0.2386 | 0.0920 | 0.1691 | 0.2305 | 0.0743 | 0.1772 |
| A11 | 0.0632 | 0.1566 | 0.2363 | 0.0816 | 0.1637 | 0.2279 | 0.0648 | 0.1616 |
| A12 | 0.0662 | 0.1682 | 0.2762 | 0.0903 | 0.1729 | 0.2645 | 0.0761 | 0.1898 |
| A13 | 0.0639 | 0.1736 | 0.2701 | 0.0780 | 0.1809 | 0.2436 | 0.0623 | 0.1714 |
| A14 | 0.0657 | 0.1580 | 0.2305 | 0.0851 | 0.1662 | 0.2392 | 0.0672 | 0.1658 |
| A15 | 0.0632 | 0.1559 | 0.2302 | 0.0792 | 0.1627 | 0.2245 | 0.0622 | 0.1606 |
| A16 | 0.0671 | 0.1590 | 0.2845 | 0.0890 | 0.1641 | 0.2495 | 0.0669 | 0.1795 |
| A17 | 0.8073 | 0.8090 | 0.8263 | 0.8998 | 0.7910 | 0.7813 | 0.7979 | 0.8016 |
| A18 | 0.3959 | 0.6189 | 0.7278 | 0.4387 | 0.6165 | 0.7141 | 0.4505 | 0.7104 |
| A19 | 0.1046 | 0.3454 | 0.3815 | 0.1208 | 0.3637 | 0.4304 | 0.1079 | 0.3273 |
| Pooled | 0.1143 | 0.2354 | 0.3411 | 0.1391 | 0.2419 | 0.3315 | 0.1204 | 0.2608 |

Table 2: *Logical access detailed results based on min-tDCF for different conditions. LA-C2 to LA-C7 indicate different codec and transmission effects. LA-C1 has no codec effect. A07 to A19 indicate different spoofing algorithms. Those algorithm are thoroughly described in [6].*

| Algorithm \ Codec | DF-C1 | DF-C2 | DF-C3 | DF-C4 | DF-C5 | DF-C6 | DF-C7 | DF-C8 | DF-C9 | Pooled |
|---|---|---|---|---|---|---|---|---|---|---|
| Traditional vocoder | 14.18% | 15.35% | 15.15% | 15.61% | 14.73% | 11.81% | 10.95% | 11.91% | 12.03% | 13.17% |
| Waveform concatenation | 7.80% | 19.84% | 8.46% | 8.80% | 9.69% | 8.86% | 5.90% | 12.35% | 8.77% | 10.38% |
| Neural vocoder (autoregre) | 22.14% | 25.62% | 22.91% | 23.13% | 22.59% | 14.70% | 13.61% | 15.84% | 14.48% | 18.42% |
| Neural vocoder | 20.24% | 23.95% | 20.53% | 22.42% | 21.69% | 14.22% | 12.95% | 15.13% | 14.34% | 17.47% |
| Unknown | 17.91% | 23.08% | 18.50% | 18.05% | 17.60% | 14.22% | 12.95% | 14.87% | 13.57% | 16.24% |
| Pooled | 18.43% | 21.49% | 18.85% | 19.44% | 19.37% | 13.21% | 12.39% | 14.09% | 13.43% | 16.05% |

Table 3: *Deepfake detection detailed results based on EER for different conditions. DF-C2 to DF-C9 indicate different audio compression techniques. DF-C1 has no codec effect. Row IDs refer to spoofing attack vocoder categories.*

| System | Progress phase | | Evaluation phase | |
|---|---|---|---|---|
| | t-DCF | EER (%) | t-DCF | EER (%) |
| ResNet-L-FM* | 0.2346 | 3.10 | 0.2845 | 4.18 |
| ResNet-L-FM | 0.2303 | 3.10 | 0.2827 | 4.10 |
| ResNet-L-LDE | 0.2377 | 3.10 | 0.2720 | 3.68 |
| ResNet-S-LDE | 0.2462 | 3.99 | 0.2844 | 4.39 |
| Fusion | 0.2137 | 2.39 | 0.2608 | 3.21 |

Table 4: *System performance on logical access track. The backend classifier in ResNet-L-FM* is trained without using SWA strategy, the embedding extractor is the same as ResNet-L-FM. The fusion method used in this work is simply averaging the score of all three systems.*

of codec augmentation in our training dataset, thus the system doesn't perform well on some of the codec and transmission effects such as LA-C3 and LA-C6.

In order to investigate the effectiveness of SWA strategy, we added the system *ResNet-L-FM**. The embedding extractor used in *ResNet-L-FM** is the same as the *ResNet-L-FM* system. However, the backend classifier used in *ResNet-L-FM** is trained using conventional strategy, while the *ResNet-L-FM** is trained using SWA. The result shows that SWA training can provide better generalization to the model.

Detailed results on Deepfake detection evaluation set report in Table 5. The final submission ensembles the three systems at the score level, achieves an EER of 16.05%. It is worth noting that our best single model system has achieved an EER of 16.36%, which is very competitive results compares to the ensemble results. Table 3 reports the EER on different conditions. Although, we only added mp3 and m4a compression effects in our training dataset, our system shows good generalization ability against most of the audio compression artifacts.

| System | Progress EER (%) | Evaluation EER(%) |
|---|---|---|
| ResNet-L-FM | 1.79 | 16.36 |
| ResNet-L-LDE | 2.31 | 16.42 |
| ResNet-S-LDE | - | 17.25 |
| Fusion | - | 16.05 |

Table 5: *System performance on the Deepfake detection full evaluation set. We did not submit the ResNet-S-LDE and Fusion systems during the progress phase due to shortage of time.*

## 5. Conclusions

In this paper, we present the Pindrop Labs's submission to the ASVspoof 2021 competition. For LA and DF tracks, we combined the Residual Neural Network architecture and different pooling techniques, and achieved very competitive results on the final evaluation set. Our final submission is a fusion of only three systems. We also investigated the effectiveness of the stochastic weight averaging. We used the SWA technique to

train the backend classifier and showed good improvement.

Although our system obtained good results on LA evaluation set, it is still not generalizing well on the DF evaluation set. We believe by adding more audio compression augmentations to the training data will further narrow the gap. More research work is needed to further improve the generalization across different audio conditions.

## 6. References


[1] Aäron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio.," *SSW*, vol. 125, 2016.

[2] Sercan Ö Arik, Mike Chrzanowski, Adam Coates, Gregory Diamos, Andrew Gibiansky, Yongguo Kang, Xian Li, John Miller, Andrew Ng, Jonathan Raiman, et al., "Deep voice: Real-time neural text-to-speech," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 195–204.

[3] Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Zongheng Yang Jaitly, Ying Xiao, Zhifeng Chen, Samy Bengio, Quoc Le, et al., "Tacotron: Towards end-to-end speech synthesis," in *Interspeech*, 2017.

[4] Jaime Lorenzo-Trueba, Junichi Yamagishi, Tomoki Toda, Daisuke Saito, Fernando Villavicencio, Tomi Kinnunen, and Zhenhua Ling, "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 195–202.

[5] Tomi Kinnunen, Jaime Lorenzo-Trueba, Junichi Yamagishi, Tomoki Toda, Daisuke Saito, Fernando Villavicencio, and Zhenhua Ling, "A spoofing benchmark for the 2018 voice conversion challenge: Leveraging from spoofing countermeasures for speech artifact assessment," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 187–194.

[6] Massimiliano Todisco, Xin Wang, Ville Vestman, Md Sahidullah, Hector Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi Kinnunen, and Kong Aik Lee, "Asvspoof 2019: Future horizons in spoofed and fake audio detection," *arXiv preprint arXiv:1904.05441*, 2019.

[7] Massimiliano Todisco, Héctor Delgado, and Nicholas Evans, "A new feature for automatic speaker verification anti-spoofing: Constant q cepstral coefficients," in *Odyssey 2016, The Speaker and Language Recognition Workshop*, 2016.

[8] Rohan Kumar Das, Jichen Yang, and Haizhou Li, "Long range acoustic and deep features perspective on asvspoof 2019," in *IEEE Autom. Speech Recognit. Understanding Workshop*, 2019.

[9] Yanmin Qian, Nanxin Chen, Heinrich Dinkel, and Zhizheng Wu, "Deep feature engineering for noise robust spoofing detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1942–1955, 2017.

[10] Hossein Zeinali, Themos Stafylakis, Georgia Athanasopoulou, Johan Rohdin, Ioannis Gkinis, Lukáš Burget, Jan Černockỳ, et al., "Detecting spoofing attacks using vgg and sincnet: But-omilia submission to asvspoof 2019 challenge," *arXiv preprint arXiv:1907.12908*, 2019.

[11] Yexin Yang, Hongji Wang, Heinrich Dinkel, Zhengyang Chen, Shuai Wang, Yanmin Qian, and Kai Yu, "The sjtu robust anti-spoofing system for the asvspoof 2019 challenge," *Proc. Interspeech 2019*, pp. 1038–1042, 2019.

[12] Balamurali BT, Kin Wah Edward Lin, Simon Lui, Jer-Ming Chen, and Dorien Herremans, "Towards robust audio spoofing detection: a detailed comparison of traditional and learned features," *arXiv preprint arXiv:1905.12439*, 2019.

[13] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, and H. Delgado, "Asvspoof2021: accelerating progress in spoofed and deep fake speech detection," *in Proc. ASVspoof 2021 Workshop*, 2021.

[14] Tianxiang Chen, Avrosh Kumar, Parav Nagarsheth, Ganesh Sivaraman, and Elie Khoury, "Generalization of audio deepfake detection," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 132–137.

[15] Weicheng Cai, Zexin Cai, Xiang Zhang, Xiaoqi Wang, and Ming Li, "A novel learnable dictionary encoding layer for end-to-end language identification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5189–5193.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," *ICASSP*, 2018.

[18] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson, "Averaging weights leads to wider optima and better generalization," *arXiv preprint arXiv:1803.05407*, 2018.

[19] Tomi Kinnunen, Héctor Delgado, Nicholas Evans, Kong Aik Lee, Ville Vestman, Andreas Nautsch, Massimiliano Todisco, Xin Wang, Md Sahidullah, Junichi Yamagishi, et al., "Tandem assessment of spoofing countermeasures and automatic speaker verification: Fundamentals," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2195–2210, 2020.