



RW-Resnet: A Novel Speech Anti-Spoofing Model Using Raw Waveform

Youxuan Ma, Zongze Ren, Shugong Xu

Shanghai Institute for Advanced Communication and Data Science (SICS), Shanghai University,
Shanghai, 200444, China

{mayouxuan, zongzeren, shugong}@shu.edu.cn

Abstract

In recent years, synthetic speech generated by advanced text-to-speech (TTS) and voice conversion (VC) systems has caused great harms to automatic speaker verification (ASV) systems, urging us to design a synthetic speech detection system to protect ASV systems. In this paper, we propose a new speech anti-spoofing model named ResWavegram-Resnet (RW-Resnet). The model contains two parts, Conv1D Resblocks and backbone Resnet34. The Conv1D Resblock is based on the Conv1D block with a residual connection. For the first part, we use the raw waveform as input and feed it to the stacked Conv1D Resblocks to get the ResWavegram. Compared with traditional methods, ResWavegram keeps all the information from the audio signal and has a stronger ability in extracting features. For the second part, the extracted features are fed to the backbone Resnet34 for the spoofed or bonafide decision. The ASVspoof2019 logical access (LA) corpus is used to evaluate our proposed RW-Resnet. Experimental results show that the RW-Resnet achieves better performance than other state-of-the-art anti-spoofing models, which illustrates its effectiveness in detecting synthetic speech attacks.

Index Terms: synthesis speech attack, anti-spoofing model, ResWavegram, RW-Resnet

1. Introduction

Speaker voice can be used to confirm a person's identity, making automatic speaker verification (ASV) become a research hot-spot. But in real-world applications, ASV systems are at the risk of various spoofing attacks. The attacker can deceive ASV systems by taking advantage of the voice characteristics of a real speaker to disguise the identity of the target speaker [1]. Synthesis speech generated by advanced text-to-speech (TTS) [2, 3, 4, 5] and voice conversion (VC) systems [6, 7] has become indistinguishable from the real speech of human in recent years, urging us to design an anti-spoofing model to protect the ASV system from synthesis speech attacks.

With the launch of ASVspoof Challenges [8, 9, 10], more and more researchers participate in spoofing attack countermeasure studies using different combinations of features and backbones [11, 12, 13, 14, 15]. The given log power spectrum (LPS) input is transformed to a genuinized feature in [15] for detecting the spoofing attacks. [16] delivers superior performance using the optimised linear frequency cepstral coefficients (LFCC). Constant Q cepstral coefficients (CQCC) and LFCC with the Siamese Convolutional Neural Networks (CNNs) are used in [14] to improve the performance of CNN based anti-spoofing systems. However, some information is discarded in the process of extracting these hand-craft features. For example, the phase information is ignored when the model applies Fast Fourier Transform (FFT) to extract the feature of an audio signal.

Recently, directly using raw waveform as system input attracts more researchers. [17] uses a modified CNN-GRU for the replay attack detection with the raw waveform as input. The adapted Rawnet2 [16] utilizes one-dimensional convolution layers and batch normalization (BN) for synthesis speech anti-spoofing. But these raw audio processing methods can not compare with traditional features while they are only using one-dimensional convolution layers. On AudioSet tagging tasks, [18] proposes Wavegram. It is a neural network based time-frequency representation which is similar to Mel-spectrogram. And the corresponding models, Wavegram-CNN and Wavegram-Logmel-CNN, demonstrate strong competitiveness.

Inspired by all above, we argue that the potential of raw waveform has not been fully stimulated. It is a trend that using raw waveform as input, so the model can be trained end-to-end. Therefore, we explore new possibilities for speech anti-spoofing by using a novel neural network to extract features from raw waveform, avoiding the complicated feature extraction process. To sum up, the main contributions of our paper are as follows:

- We use Conv1D blocks to obtain the two-dimensional Wavegram feature from the raw waveform to replace the complicated process of hand-craft feature extraction. As we know, it is the first try of getting the neural network based time-frequency feature representation by directly using the raw waveform as input in the Speech Anti-Spoofing field.
- On the basis of Conv1D block, we propose the Conv1D Resblock by adding a residual connection. The new path covers long-term features while the original path focuses on short-term features. By stacking the proposed Conv1D Resblocks, we get a better feature extractor for extracting features from the raw waveform. And the feature we get is called ResWavegram.
- We propose a novel Anti-Spoofing model named ResWavegram-Resnet (RW-Resnet) and train it end-to-end. Compared with other speech anti-spoofing models based on traditional hand-craft features, the proposed RW-Resnet makes a great improvement. The result on the ASVspoof2019 logical access (LA) corpus shows that our model achieves state-of-the-art with Equal Error Rate (EER) of 2.98% and tandem detection cost function (t-DCF) of 0.0817, surpassing most advanced anti-spoofing approaches.

The remainder of this paper is organized as follows. Section II introduces our speech anti-spoofing model. In Section III we present our experimental configuration, and our experimental results and discussions will be introduced in the Section IV. Finally, we give our conclusion in Section V.

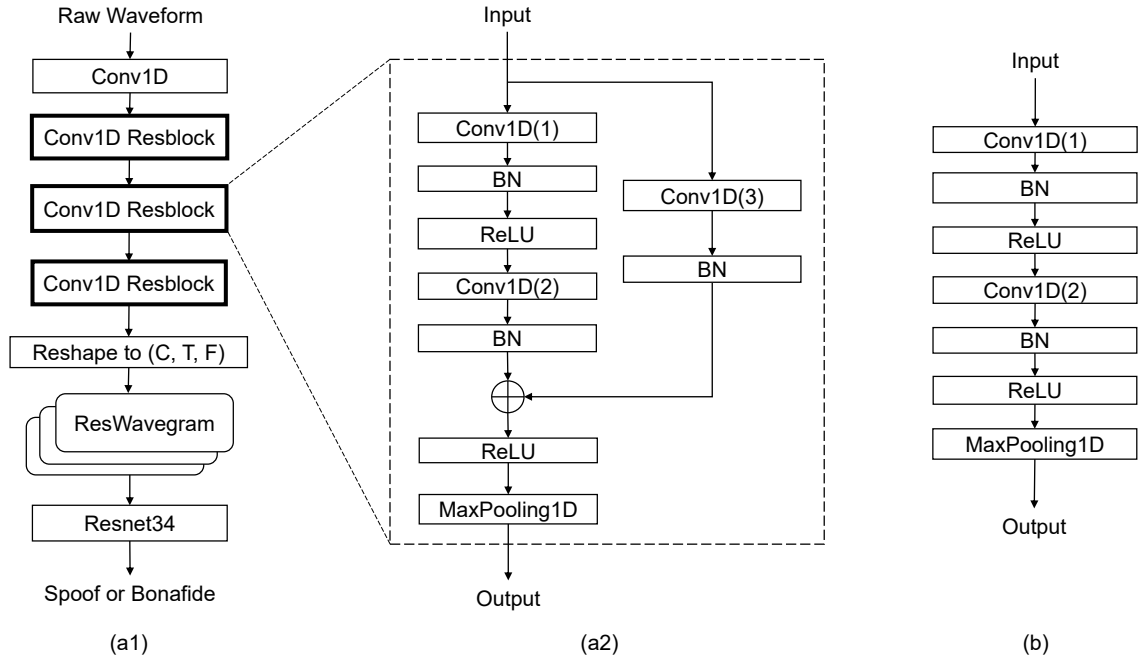


Figure 1: *ResWavegram* based anti-spoofing model. (a1) shows the *ResWavegram* feature used for the anti-spoofing model. (a2) presents our proposed *Conv1D Resblock* by adding the *Conv1D(3)* layer and the followed *BN* layer as a residual connection. The dilation of the *Conv1D(1)* and *Conv1D(3)* layer is set to 1, while that of the *Conv1D(2)* layer is set to 2. All kernel size of the convolution layers in the *Conv1D Resblock* is set to 3. (b) shows the details of the original *Conv1D* block.

2. Model Architecture

Our anti-spoofing model mainly includes two parts: the feature extraction (Wavegram or *ResWavegram*) and the backbone (Resnet34). Before training, the time length of input audio is limited to 8 seconds by concatenating or cutting operations to ensure the same size of the input data. All audio samples are divided by 32768 (2^{15}), ensuring that their numerical range is restricted in $[-1, 1]$. Finally, the raw waveform is directly fed to the neural network and the whole model is trained end-to-end.

2.1. Wavegram

Since the time-domain CNN models [16, 17] cannot capture the frequency relationships within an audio signal, it is important to construct a frequency axis from the one-dimensional CNNs based feature. The extraction process of Wavegram is shown in Table 1. The key structure to extract the Wavegram feature is the *Conv1D* block (Figure 1 (b)). To improve the receptive field of the network, the *Conv1D* block consists of two convolution layers whose dilations are set to 1 and 2, respectively. Each *Conv1D* block contains a *MaxPooling* layer to down-sample the original audio signal.

The output size of an audio signal is changed from 1×128000 to $C \times T$ in a mini-batch. C represents the final output channels, which equals C_3 in Table 1. The number of the output time frame T is $128000 / (5 \times 4^3) = 400$, while the stride of the *Conv1D* layer is set to 5 and the pooling size of the *MaxPooling* layer in each *Conv1D* block is set to 4. Then we reshape the feature size to $\frac{C}{F} \times T \times F$ by splitting the channels to $C_g = \frac{C}{F}$ group(s), where each group can be regarded as having F

frequency bins. In this way, each audio finally is transformed to a feature map whose size is of $C_g \times 400 \times F$. Then the feature is sent to the backbone Resnet34 to predict the input audio is either bonafide or spoofed.

Table 1: *The extraction process of Wavegram. We can get different Wavegram based features by setting (C_1, C_2, C_3) to different values. In our experiments, Wavegram-S, Wavegram-M and Wavegram-L features are got by setting (C_1, C_2, C_3) to $(64, 64, 64)$, $(64, 128, 128)$ and $(64, 128, 256)$, respectively.*

Layer	Output Size
Input	1×128000
Conv1D	64×25600
Conv1D block	$C_1 \times 6400$
Conv1D block	$C_2 \times 1600$
Conv1D block	$C_3 \times 400$
Output(Wavegram)	$C_g \times 400 \times F$

2.2. ResWavegram

Inspired by the residual connection proposed in the Resnet network [19], we add a residual connection between the input and output of the two convolution layers in each *Conv1D* block. The modified block is shown in Figure 1 (a2), and we call it *Conv1D Resblock*. The residual connection we add consists of a convolution layer and a batch normalization (BN) layer. In each *Conv1D Resblock*, the receiving field becomes larger with the kernel size of the *Conv1D(3)* layer is set to 3. A BN layer is added after the *Conv1D (3)* layer to speed up the training pro-

cess and reduce the over-fitting effect of the network.

We get the ResWavegram feature by replacing the Conv1D block in the Wavegram extraction process with the Conv1D Resblock, as shown in Figure 1 (a1). By adding such a residual connection, the perceptual information range of our feature extraction process is deepened and the information of gradient update can be better transmitted. The neural network ultimately learns a better feature representation from the raw waveform, which is proved by the experimental results.

2.3. Resnet

Resnet is widely used in sound-related tasks, and it is also a popular choice in the field of anti-spoofing. We believe that the Resnet can learn a more distinctive feature from the raw waveform. So, we use Resnet from [19], but the channels of each ResBlock are changed to 1/4 of the original ones. Two fully connected layers are added after the pooling layer, as shown in Table 2. A skip connection is added between the outputs of the pooling layer and the FC2 layer to ensure that the gradient update information can be better passed through the network. We also add the Rectified Linear Unit (ReLU) activation function after the FC1 layer.

Table 2: Detailed architecture of Resnet. The Adaptive Average Pooling is selected as the pooling layer and the output size is set to (1,1), making each channel has one output fed to the fully connected layers.

Layer	Output Size	Channels	Blocks
Conv&BN	$T \times F$	16	-
Res1	$T \times F$	16	3
Res2	$T/2 \times F/2$	32	4
Res3	$T/4 \times F/4$	64	6
Res4	$T/8 \times F/8$	128	3
Pooling	128	-	-
FC1	128	-	-
FC2	128	-	-
Output	2	-	-

3. Experiments Setup

3.1. Database

The database we use is the logical access (LA) corpus of ASVspoof2019 Challenge [20], which is based upon a multi-speaker speech synthesis database called VCTK [21]. All audio samples in the database are sampled at 16 kHz and 16-bit quantization. The database is divided into three subsets, namely train, development and evaluation, and the details are shown in Table 3. The evaluation subset contains a set of unseen bonafide and spoofed speech collected from multiple speakers, and the spoofed speech is generated by diverse unseen spoofing algorithms. We use the train and development subsets for joint training and the evaluation subset is used for evaluation, as many researchers do.

3.2. Model training

We use Pytorch as the training platform and all the experiments are implemented with Python language. We train each model with 50 epochs, while the batch size is set to 16. We select Adam [22] as the optimizer with the initial learning rate of $1E-4$ and the weight decay of 0. The Cosine Annealing Warm

Table 3: Detailed information of the ASVspoof2019 LA corpus

Subset	Male	Female	Bonafide	Spoofed
Train	8	12	2580	22800
Development	4	6	2548	22296
Evaluation	21	27	7355	63882

Restarts algorithm [23] is used for updating the learning rate and the learning rate is gradually down to $1E-8$ during the training process. Cross entropy (CE) loss is chosen as the loss function. Kaiming initialization [24] is used for all the convolution layers while the weight and bias of the batch normalization (BN) layers are set to 1 and 0 respectively.

3.3. Evaluation metrics

We adopt the same two metrics that are used in ASVspoof2019 Challenge [10]. The primary metric used for evaluation is tandem detection cost function (t-DCF), ensuring the evaluation results will show the performance of spoofing countermeasures (CMs) on the reliability of an ASV system. Equal error rate (EER) is selected as the second evaluation metric, corresponding to the CM threshold where the miss and false alarm rates equal each other.

When doing evaluation, the score is calculated using the log-likelihood ratio.

$$score(U) = \log(p(bonafide/U; \theta)) - \log(p(spoof/U; \theta)) \quad (1)$$

where U represents the given audio signal to be tested and θ represents the parameters of our trained model.

4. Results and Discussion

4.1. Baselines

ASVspoof2019 Challenge introduces two baselines with the Gaussian mixture model (GMM) as the backend classifier. The two baselines use different acoustic frontends, namely linear frequency cepstral coefficients (LFCC) and constant Q cepstral coefficients (CQCC) [10]. The scores of these two official systems on the ASVspoof2019 LA evaluation set are shown in Table 6.

4.2. Ablation studies

To avoid the over-fitting and under-fitting situations during training, different channel combination configurations are explored to find a better combination for extracting the Wavegram. (C_1, C_2, C_3) in Table 1 are set to (64, 64, 64), (64, 128, 128) and (64, 128, 256) respectively. According to the combination configurations of the selected channels, we respectively named the three corresponding features as Wavegram-S, Wavegram-M and Wavegram-L. We select different C_g in each model to explore its impact on the performance of our proposed anti-spoofing model. In our experiment, the values of C_g are set to 1, 2, and 4, respectively. The ablation results are shown in Table 4.

In Table 4, we can see that setting $C_g = 1$ usually achieves better results for the three models. One possible explanation is that all the extracted information can be placed in the dimension of the frequency when C_g is set to 1, obtaining a more complete feature representation for the backbone Resnet34. The Wavegram-M based model always achieves the best results

when select the same C_g for these models, showing the channel combination of (64, 128, 128) prevents our proposed model from the over-fitting and under-fitting situations. All scores got from these models are better than the official baseline systems.

Table 4: Wavegram based models tested on the ASVspoof2019 LA evaluation set. Wavegram-S, Wavegram-M and Wavegram-L with different values of C_g are tested with the backbone Resnet34.

Feature	C_g	t-DCF	EER(%)
Wavegram-S	1	0.1074	4.34
	2	0.1240	5.52
	4	0.1121	4.56
Wavegram-M	1	0.0849	3.39
	2	0.0922	3.44
	4	0.0938	3.68
Wavegram-L	1	0.1007	4.01
	2	0.0996	3.99
	4	0.1183	5.05

4.3. Comparative results

Based on the experimental results on Wavegram, we set the value of C_g to 1 to make better use of the learned raw waveform features. Comparative experiments are done with the value of (C_1, C_2, C_3) changes. Like the extraction process of Wavegram, we can get the ResWavegram-S, ResWavegram-M and ResWavegram-L features by setting (C_1, C_2, C_3) to (64, 64, 64), (64, 128, 128) and (64, 128, 256), respectively. The final results of our experiments upon the Wavegram and our proposed ResWavegram are shown in Table 5.

Table 5: Comparative experiments of the Wavegram and ResWavegram tested on the ASVspoof2019 LA evaluation set. The value of C_g is set to 1 in all the following models.

Feature	t-DCF	EER(%)
Wavegram-S	0.1074	4.34
ResWavegram-S	0.0818	3.25
Wavegram-M	0.0849	3.39
ResWavegram-M	0.0817	2.98
Wavegram-L	0.1007	4.01
ResWavegram-L	0.0930	3.56

Compared to the Wavegram, ResWavegram achieves better performance under different configurations, especially for the ResWavegram-S feature. The network structure with the ResWavegram has much power to extract feature information from the waveform by using the residual connection in the Conv1d Resblock. Compared with the original Wavegram, the ResWavegram feature has approximately 12% improvement on the EER evaluation metric. Our ResWavegram-Resnet (RW-Resnet) model improves the t-DCF and EER metrics of the best baseline LFCC-GMM by 61% and 63% respectively.

As shown in Table 6, the proposed Wavegram-Resnet and RW-Resnet models for synthetic speech attacks detection are compared with other advanced systems tested on the ASVspoof2019 LA evaluation set [20]. Compared with neural networks that use traditional methods to extract features [11, 12, 15], the performances of our proposed models are greatly improved. Compared with the RawNet2 that only uses

Table 6: Performance comparison of the proposed Wavegram-Resnet and RW-Resnet systems to some known single systems on the ASVspoof2019 LA evaluation set.

System	t-DCF	EER(%)
CQCC-GMM (Baseline1) [10]	0.237	9.57
LFCC-GMM (Baseline2) [10]	0.212	8.09
MFCC-Resnet [12]	0.204	9.33
Spec-Resnet [12]	0.274	9.68
CQCC-Resnet [12]	0.217	7.69
LFCC-Siamese CNN [13]	0.093	3.79
Optimised LFCC (full-band) [14]	0.090	3.50
LFCC-LCNN [11]	0.100	5.06
FFT-LCNN [11]	0.103	4.53
FG-LCNN [15]	0.102	4.07
S1-RawNet2 [16]	0.130	5.64
S2-RawNet2 [16]	0.118	5.13
S3-RawNet2 [16]	0.129	4.66
Wavegram-Resnet (Ours)	0.085	3.39
RW-Resnet (Ours)	0.082	2.98

one-dimensional convolution [16], our method of using two-dimensional convolution has better feature performance capabilities. Compared with the well-designed system with the Siamese CNN [13], our proposed models achieve better performance using a slightly modified Resnet34 as the backbone. The performances of the two models we proposed far exceed the two baselines [10] and other recently proposed single systems on the ASVspoof2019 LA evaluation set. Finally, we compare the score of our proposed RW-Resnet single system with the top5 fusion systems in ASVspoof2019 Challenge, as shown in Table 7.

Table 7: Performance comparison of our proposed RW-Resnet single system to the top5 fusion systems in ASVspoof2019 Challenge on the ASVspoof2019 LA evaluation set.

System	t-DCF	EER(%)
T05	0.0069	0.22
T45 [11]	0.0510	1.86
T60 [25]	0.0755	2.64
RW-Resnet (Ours)	0.0817	2.98
T24	0.0953	3.45
T50 [26]	0.1118	3.56

5. Conclusions

A novel speech anti-spoofing model named RW-Resnet is introduced in this paper. We propose the Conv1D Resblock with a residual connection, which allows the model to learn a better feature representation from the raw waveform. The effectiveness of directly feeding the raw waveform into a neural network, with no information of the raw waveform is discard, has been verified by comparing with other advanced speech anti-spoofing systems. Our RW-Resnet improves the t-DCF and EER metrics of the best baseline LFCC-GMM by 61% and 63% respectively. In the future, we intend to adapt our proposed method to other sound fields, and explore more expressive raw-waveform based models.

6. References

- [1] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *speech communication*, vol. 66, pp. 130–153, 2015.
- [2] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.
- [3] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.
- [4] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3617–3621.
- [5] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," *Proc. Interspeech 2017*, pp. 4006–4010, 2017.
- [6] K. Tanaka, H. Kameoka, T. Kaneko, and N. Hojo, "Atts2s-vc: Sequence-to-sequence voice conversion with attention and context preservation mechanisms," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6805–6809.
- [7] J.-X. Zhang, Z.-H. Ling, and L.-R. Dai, "Non-parallel sequence-to-sequence voice conversion with disentangled linguistic and speaker representations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 540–552, 2019.
- [8] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, "Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.
- [9] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, and K. A. Lee, "The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. Interspeech 2017*, 2017, pp. 2–6. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2017-1111>
- [10] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. H. Kinnunen, and K. A. Lee, "ASVspoof 2019: Future Horizons in Spoofed and Fake Audio Detection," in *Proc. Interspeech 2019*, 2019, pp. 1008–1012. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-2249>
- [11] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "STC Antispoofing Systems for the ASVspoof2019 Challenge," in *Proc. Interspeech 2019*, 2019, pp. 1033–1037. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-1768>
- [12] M. Alzantot, Z. Wang, and M. B. Srivastava, "Deep Residual Neural Networks for Audio Spoofing Detection," in *Proc. Interspeech 2019*, 2019, pp. 1078–1082. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-3174>
- [13] Z. Lei, Y. Yang, C. Liu, and J. Ye, "Siamese Convolutional Neural Network Using Gaussian Probability Feature for Spoofing Speech Detection," in *Proc. Interspeech 2020*, 2020, pp. 1116–1120. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2020-2723>
- [14] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "Spoofing Attack Detection Using the Non-Linear Fusion of Sub-Band Classifiers," in *Proc. Interspeech 2020*, 2020, pp. 1106–1110. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2020-1844>
- [15] Z. Wu, R. K. Das, J. Yang, and H. Li, "Light Convolutional Neural Network with Feature Genuinization for Detection of Synthetic Speech Attacks," in *Proc. Interspeech 2020*, 2020, pp. 1101–1105. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2020-1810>
- [16] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," *arXiv preprint arXiv:2011.01108*, 2020.
- [17] J. weon Jung, H. jin Shim, H.-S. Heo, and H.-J. Yu, "Replay Attack Detection with Complementary High-Resolution Information Using End-to-End DNN for the ASVspoof 2019 Challenge," in *Proc. Interspeech 2019*, 2019, pp. 1083–1087. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-1991>
- [18] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.
- [19] 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.
- [20] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee *et al.*, "Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, p. 101114, 2020.
- [21] J. Yamagishi, C. Veaux, K. MacDonald *et al.*, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92)," 2019.
- [22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015.
- [23] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.
- [24] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.
- [25] B. Chettri, D. Stoller, V. Morfi, M. A. M. Ramírez, E. Benetos, and B. L. Sturm, "Ensemble Models for Spoofing Detection in Automatic Speaker Verification," in *Proc. Interspeech 2019*, 2019, pp. 1018–1022. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-2505>
- [26] Y. Yang, H. Wang, H. Dinkel, Z. Chen, S. Wang, Y. Qian, and K. Yu, "The SJTU Robust Anti-Spoofing System for the ASVspoof 2019 Challenge," in *Proc. Interspeech 2019*, 2019, pp. 1038–1042. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2019-2170>