Towards End-to-End Synthetic Speech Detection

Guang Hua, *Member, IEEE*, Andrew Beng Jin Teoh, *Senior Member, IEEE*, and Haijian Zhang, *Member, IEEE*

***Abstract*—The constant Q transform (CQT) has been shown to be one of the most effective speech signal pre-transforms to facilitate synthetic speech detection, followed by either hand-crafted (subband) constant Q cepstral coefficient (CQCC) feature extraction and a back-end binary classifier, or a deep neural network (DNN) directly for further feature extraction and classification. Despite the rich literature on such a pipeline, we show in this paper that the pre-transform and hand-crafted features could simply be replaced by end-to-end DNNs. Specifically, we experimentally verify that by only using standard components, a light-weight neural network could outperform the state-of-the-art methods for the ASVspoof2019 challenge. The proposed model is termed Time-domain Synthetic Speech Detection Net (TSSDNet), having ResNet- or Inception-style structures. We further demonstrate that the proposed models also have attractive generalization capability. Trained on ASVspoof2019, they could achieve promising detection performance when tested on disjoint ASVspoof2015, significantly better than the existing cross-dataset results. This paper reveals the great potential of end-to-end DNNs for synthetic speech detection, without hand-crafted features.**

***Index Terms*—Synthetic speech detection, speech forensics, ASVspoof2019, ASVspoof2015, cross-dataset testing, end-to-end.**

I. Introduction

THE success of deep learning technology has shifted the paradigm of speech synthesis from the classic hidden Markov model based framework [1] to neural speech synthesis. Equipped with powerful deep neural network (DNN) architectures e.g., [2], and fueled by massive training data, today's text-to-speech (TTS) systems could synthesize high quality speech that is hard to be distinguished from human voices. Despite the multitude of benefits, these advances have also improved the quality of voice spoofing attacks, including voice conversion [3], impersonation [4], cloning [5], etc., posing new challenges to synthetic speech detection.

For nearly a decade, the combination of a front-end feature extractor and a back-end binary classifier is the *de facto* framework for synthetic speech detection. Within this framework, an overwhelming majority of the existing works have focused on the development of hand-crafted front-end features, including fundamental frequency, power spectrum, octave spectrum, linear frequency cepstral coefficient (LFCC), mel-frequency cepstral coefficient (MFCC), cepstral mean and variance (CMVN), cochlear filter cepstral coefficient (CFCC), filter bank based cepstral coefficient, linear prediction cepstral coefficient (LPCC), modified group delay (MGD), relative phase shift (RPS), constant Q cepstral coefficient (CQCC), and many of their variations and combinations [6]–[15]. Usually, one or a few of these features are used to train a Gaussian mixture model (GMM) or a support vector machine (SVM) for classification. Taking the advantage of DNNs in classification tasks, multilayer perceptron (MLP) and convolutional neural network (CNN) based classifiers have been used to replace the conventional back-end classifiers [15]–[22]. On the other side, DNN structures have also been used at the front-end to facilitate feature extraction [22]–[25], followed by conventional classifiers. DNNs can also work across the front- and back-end, with pre-transformed features as input [26]–[30].

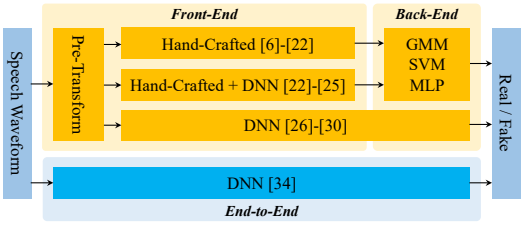


Fig. 1. Relationship between the existing front-end→back-end pipeline and the proposed end-to-end framework for synthetic speech detection.

Among hand-crafted features, CQCC has been found to be the best choice, which is also the baseline feature in the ASVspoof2019 challenge [31]. Recently, Yang *et al.* developed a set of subband CQCC features for better detection performance [19]. Subsequently, Das *et al.* [21] further fused 8 hand-crafted features, followed by an MLP classifier. For deep learning based approach, Lavrentyeva *et al.* [29] proposed the use of FFT, LFCC, and CMVN, followed by a CNN for classification, while using CQT as model input, Li *et al.* [30] incorporated the so called Res2Net structure and squeeze-and-excitation (SE) block. With score level fusion, Lavrentyeva *et al.* [29] and Li *et al.* [30] have achieved the state-of-the-art performance on ASVspoof2019 dataset.

Based on the above overview, the existing mainstream workflow for synthetic speech detection is summarized in the brown blocks of Fig. 1. It can be seen that a time-frequency transform (e.g., CQT) of the speech waveform before hand-crafted feature extraction (e.g., CQCC), or before feeding the data into a DNN, has become an implicit standard routine in the existing works. However, since DNNs are best known for their excellent capability of feature extraction, there naturally arises a question of whether it is necessary to apply these pre-transforms. In fact, these transforms usually discard some information about the observed speech signal. For example, the CQT feature, more precisely the log power spectrum of the CQT [30], does not have the phase information of the signal.

This work was supported by the 2020–2021 International Scholar Exchange Fellowship (ISEF) Program at the Chey Institute for Advanced Studies, South Korea. *(Corresponding Author: Andrew Beng Jin Teoh)*

G. Hua and H. Zhang are with the School of Electronic Information, Wuhan University, Wuhan 430072, China (e-mail: ghua@whu.edu.cn; haijian.zhang@whu.edu.cn).

A. B. J. Teoh is with the School of Electrical and Electronic Engineering, College of Engineering, Yonsei University, Seoul 120749, South Korea (e-mail: bjteoh@yonsei.ac.kr).

To further generate the CQCC, even more information will be discarded [19]. From hand-crafted feature engineering point of view, a good feature captures discriminative information between classes and is also compact in size, but the same principle may not apply to the DNN regime.

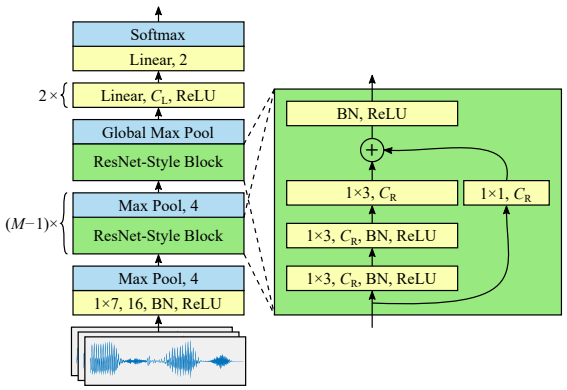
In this paper, we show that the pre-transforms, as well as the hand-crafted features, are in fact not a must for DNN based synthetic speech detection. Despite the rich hand-crafted features, we experimentally verify that via the use of standard DNN structures, an end-to-end light-weight neural network with mere speech waveform could achieve even better results. Our proposal is motivated by recent works analyzing raw-waveform based DNNs [32] and the attempt of applying end-to-end DNNs to related speech processing tasks, e.g., speech separation [33]. The proposed model is thus termed as Time-domain Synthetic Speech Detection Net (TSSDNet). We note that the first work on end-to-end synthetic speech detection was probably carried out by Muckenhirn *et al.* [34], in which a basic feedforward sequential CNN was used. It was tested on older datasets, not achieving the state-of-the-art results. In our design of the TSSDNet, two types of advanced CNN structures are considered, including ResNet-style skip connection with 1×1 kernels [35] and Inception-style parallel convolutions [36], respectively. We demonstrate that via proper training, the proposed networks outperform the state-of-the-art hand-crafted feature based detectors as well as DNN based ones on the challenging ASVspoof2019 dataset [31]. To analyze practical merits of the proposed methods, we further perform a cross-dataset evaluation between ASVspoof2019 and ASVspoof2015 [37][1], demonstrating their promising generalization capability.

II. The Proposed Models

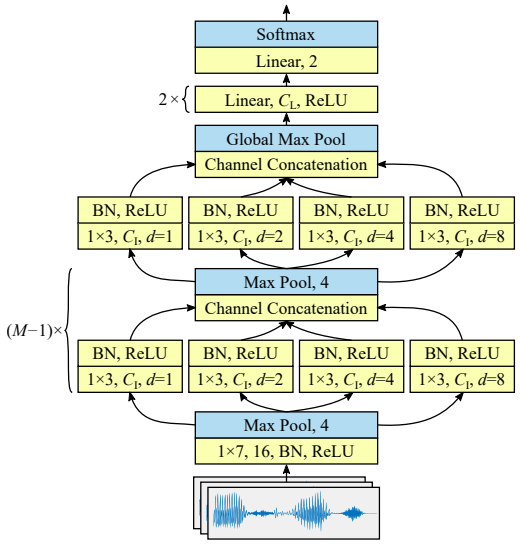
In many deep learning tasks such as object recognition or semantic understanding, it has been found that generally the deeper the network, the better the performance [35], [36], [38]. However, in synthetic speech detection, the critical feature is the artifact left behind data forgery, which may not contain any semantic information. Since deeper features are more towards higher level semantic information, which may not be suitable to represent the subtle forgery artifacts, we hypothesize that the network for synthetic speech detection should be relatively shallower. Grabbing the essence of the popular ResNet [35] and Inception network [36], the proposed end-to-end TSSDNets are designed as follows.

A. Model Structure

The proposed Res-TSSDNet and Inc-TSSDNet are depicted in Fig. 2 (a) and (b), respectively, which share the same first layer, 3 final fully-connected linear layers, and global max pooling before the linear layers. The ResNet-style and Inception-style blocks are repeated for M times, respectively, and batch normalization (BN) is applied in both networks. C_{R} and C_{I} denote the number of channels in the corresponding modules, which may vary across layers. Noticeably, to increase the receptive field and control model complexity, dilated convolutions [39] with dilation d are incorporated in the Inc-TSSDNet, which is different from the original Inception network [36]. All the convolution layers apply "SAME" padding with stride $= 1$, while for the pooling layers the stride equals to the corresponding kernel size.



(a) ResNet style, Res-TSSDNet.



(b) Inception style, Inc-TSSDNet.

Fig. 2. Structures of the proposed models, where all the conv layers apply "SAME" padding, and in all local max pooling layers, stride=kernel size. M: number of stacked ResNet- and Inception-Style modules. C_{R}: number of channels in Res-TSSDNet. C_{I}: number of channels in Inc-TSSDNet.

B. Training Strategy

1) Data Preparation: Normally, the training data contain raw speech recordings with varied durations. To align the training data, we adopt the treatment in [30]. In [30], the training examples are truncated or repeated until the duration is 6.4 seconds to generate the CQT feature, while here we keep every example with 6 seconds, with the default 16 kHz sample rate. These 6-second examples are then directly fed into the networks for end-to-end training. Since all convolution layers have "SAME" padding, the length of feature vector (9.6×10^4 at input) is reduced solely by the pooling layers.

[1]The ASVspoof2017 dataset is not considered in this paper because it only has replay attack. Although replay attack is seen to be considered together with synthesis attack, the underlying mechanism is very different. The physical access portion of ASVspoof2019 is also excluded for the same reason.

Note that hand-crafted feature based method, e.g., CQCC [19], is insensitive to the length of recording since all time slices contribute to classifier training. Batch size is set to 32. .

2) Weighted Cross-Entropy Loss: Considering the fact that in general data-driven media content forgery detection tasks, the number of genuine examples is usually much less than the number of fake ones, we apply weighted cross-entropy (WCE) loss during the training phase to cope with data imbalance. Let $\{x_i, y_i\}$ compose the labeled training set, where $\forall i$, label $y_i \in \{0, 1\}$, then the WCE loss is given by

$$\mathrm{WCE}\left(\mathbf{z}, y_i\right) = -w_{y_i} \log\left(z_{y_i}\right), \tag{1}$$

where $\mathbf{z} = [z_0, z_1]$ contains the softmax probabilities of the 2 classes, and w_{y_i} is the inverse ratio of label y_i in the training set. For all the training processes, we use the Adam [40] optimizer and default settings. Exponential learning rate decay with a multiplicative factor of 0.95 is adopted. The model yielding the lowest equal error rate (EER) on development set within 100 epochs is selected for evaluation.

3) Mixup Regularization: For practical forensic merits, the trained model is expected to generalize to unseen attacks, and the ASVspoof datasets have been specially designed for this purpose. In this paper, we consider the mixup regularization [41] as a booster to further improve the generalization capability. Specifically, it uses a set of mixed examples and labels, instead of the original set, to train the network, i.e.,

$$\tilde{x}_i = \lambda x_i + (1-\lambda)x_j, \quad \tilde{y}_i = \lambda y_i + (1-\lambda)y_j, \tag{2}$$

where $\{x_i, y_i\}$ and $\{x_j, y_j\}$ are two randomly selected training pairs, $\lambda \sim \mathrm{Beta}(\alpha, \alpha)$, and $\alpha \in (0, \infty)$ is a hyperparameter. The implementation of the mixup regularization could be carried out via the following equivalent loss function,

$$\mathrm{CE}_{\mathrm{mixup}}\left(\tilde{\mathbf{z}}, y_i, y_j\right) = \lambda \mathrm{CE}(\tilde{\mathbf{z}}, y_i) + (1-\lambda)\mathrm{CE}(\tilde{\mathbf{z}}, y_j), \tag{3}$$

where $\tilde{\mathbf{z}}$ contains the softmax probabilities from mixed examples, and $\mathrm{CE}(\cdot, \cdot)$ is the standard cross-entropy (CE) loss, equivalent to setting $w_0 = w_1$ in (1).

III. Results

We first present the main results obtained by the proposed networks in comparison with the benchmark and the state-of-the-art solutions on the latest ASVspoof2019 dataset. We then perform ablation study, followed by cross-dataset evaluation on the ASVspoof2015 dataset. All the results are generated using a single GeForce GTX 1080 or 1080Ti GPU. PyTorch implementations of the proposed TSSDNets are available at: *https://github.com/ghuawhu/end-to-end-synthetic-speech-detection*.

A. Main Results

The comparison of the results in terms of EER obtained on the logical access (LA) development and evaluation sets of the ASVspoof2019 challenge is presented in Table I, where the 2D-Res-TSSDNet is the 2D version of the Res-TSSDNet, having the same architecture except that all the convolution and pooling (2×2 pooling) layers use 2D kernels instead.

We make the following remarks from the main results. **i)** The works of [19] and [21] represent the best results of sophisticated hand-crafted feature engineering plus an MLP as the back-end classifier. **ii)** The majority of recent works belong to the type of pre-transform (or light feature engineering) plus DNNs to further perform feature extraction and classification. **iii)** All the works incorporating DNNs rely on the feature and model fusion for performance improvement, and in [29] and [30], the fused results have achieved EERs below 2%. **iv)** The 2D-Res-TSSDNet result is obtained with experimental settings identical to [30] without fusion, and it can be seen that when working with 2D pre-transform input, the use of advanced DNN components, i.e., Res2Net and SE, becomes very necessary. **v)** Most importantly, the proposed Res-TSSDNet is a single end-to-end network (no fusion, no feature engineering), containing less than a half of trainable weights than the one in [30] and only about one-tenth than in [29], but it achieves the overall lowest evaluation EER by a clear margin. **vi)** Lastly, the Inc-TSSDNet is extremely light, having only 0.09M parameters, but it could still achieve an EER lower than those from [27]–[29] heavy models.

TABLE I
EER (%) OF THE PROPOSED AND STATE-OF-THE-ART METHODS ON ASVSPOOF2019 LA DEV AND EVAL SETS, $M = 4$, $C_{\mathrm{L}} = \{64, 32\}$, $C_{\mathrm{R}} = \{32, 64, 128, 128\}$, $C_{\mathrm{I}} = \{8, 16, 32, 32\}$.

Method	#Param	Dev	Eval
Baseline LFCC+GMM [42]	-	0.43	9.57
Baseline CQCC+GMM [42]	-	2.71	8.09
Subband CQCC+MLP [19]	-	-	8.04
8 Features+MLP [21]	-	0.00	4.13
Spec+VGG+SincNet [28]	>4.32M	0.00	8.01
Spec+CQCC+ResNet+SE [27]	5.80M	0.00	6.70
FFT+CNN [29]	10.2M	0.04	4.53
3 Features+CNN [29]	30.6M	0.00	1.86
CQT+Res2Net+SE [30]	0.92M	0.43	2.50
3 Features+Res2Net+SE [30]	2.76M	0.00	1.89
CQT+2D-Res-TSSDNet	0.97M	0.59	5.89
End-to-End Res-TSSDNet	0.35M	0.74	**1.64**
End-to-End Inc-TSSDNet	**0.09**M	1.09	4.04

TABLE II
ABLATION STUDY OF RES-TSSDNET AND INC-TSSDNET, USING ASVSPOOF2019 LA EVAL EER (%), $C_{\mathrm{L}} = \{64, 32\}$.

	M	C_{R}	1×1	#Param	Eval
Res-TSSDNet	3	$\{32, 64, 128\}$	Yes	0.18M	11.37
	4	$\{32, 64, 128, 128\}$	No	0.32M	2.69
	4	$\{32, 64, 128, 128\}$	Yes	0.35M	**1.64**
	5	$\{32, 64, 128, 128, 128\}$	No	0.47M	5.14
	5	$\{32, 64, 128, 128, 128\}$	Yes	0.51M	4.58
	M	C_{I}	Dilation d	#Param	Eval
Inc-TSSDNet	3	$\{8, 16, 32\}$	$\{2^0, \ldots, 2^3\}$	0.04M	10.39
	4	$\{8, 16, 32, 32\}$	$\{2^0, \ldots, 2^3\}$	0.09M	4.04
	5	$\{8, 16, 32, 64, 64\}$	$\{2^0, \ldots, 2^3\}$	0.35M	5.31
	4	$\{8, 16, 32, 32\}$	$\{2^0, \ldots, 2^7\}$	0.34M	**3.75**
	5	$\{8, 16, 32, 64, 64\}$	$\{2^0, \ldots, 2^7\}$	1.34M	4.20

B. Ablation Study

We first perform ablation study by varying the depth or width of the networks, and the results are summarized in Table II. For the Res-TSSDNet, the column "1×1" indicates whether the "skip connection" in Fig. 2 (a) is used. It can be seen that going either shallower or deeper will result in the raise

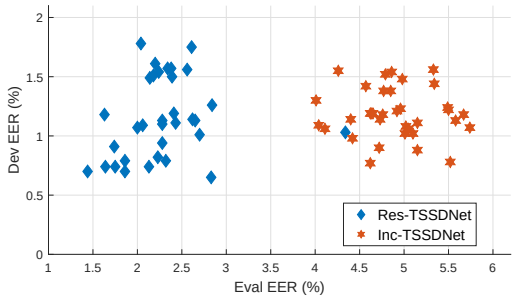


Fig. 3. Intra-model performance on ASVspoof2019.

of EER, while with the use of ResNet skip connection, the network could achieve 1.05% EER reduction over the one without using it. Similarly for the Inc-TSSDNet, the sweet spot also lies in the moderate depth or width.

We further perform intra-model sensitivity analysis using the two proposed end-to-end models in Table I. Fixing all hyperparameters, the two models are trained from scratch using ASVspoof2019 training set for over 30 times, and the dev and eval EERs are summarized in Fig. 3. It can be seen that the EERs of both models are bounded within certain ranges (except one outlier eval EER $> 4\%$ for the Res-TSSDNet). The Inc-TSSDNet yields tighter dev EERs over the Res-TSSDNet, but eval EERs of the former are clearly higher. We can see from Fig. 3 and Table II that the intra-model differences may be as significant as the differences from model configurations. Relatively lighter models are hence recommended for their better trade-offs between accuracy and efficiency.

In addition, we have also discovered that **i)** changing all the activations from ReLU to leaky or parametric ReLU does not lead to a clear performance difference; **ii)** The first layer with a 1×7 convolution kernel, adopted from ResNet setting, is slightly better than using a 1×3 kernel; **iii)** Global max pooling is found to be more effective than global average pooling before the linear layers for both networks, but for the 2D-Res-TSSDNet, we stick to global average pooling; **iv)** The EERs of using standard CE are slightly higher than those using the WCE; **v)** Duration of training example also matters. Experimental results using 5-second truncation yielded a slight performance degradation, but when 2-second truncation is applied, the EER on evaluation set increased drastically.

C. Cross-Dataset Testing

We now perform the cross-dataset experiments. Since the ASVspoof2015 training set contains relatively old speech synthesis methods, we focus on using networks trained on the training set of more advanced ASVspoof2019 to test on the dev and eval sets of ASVspoof2015. The intra- and inter-dataset EERs are presented in Table III. It can be seen that the GMMs learned from LFCC and CQCC features in ASVspoof2019 are generally inconsistent with the data in ASVspoof2015. For the best Res-TSSDNet on ASVspoof2019, it could not generalize to ASVspoof2015 either, whose EERs indicate almost indistinguishable softmax probability distributions for real and fake classes. However, by incorporating mixup regularization and increasing the level of mixup level α, we observe that the

TABLE III
EER (%) OF NETWORKS TRAINED ON ASVSPOOF2019 TRAINING SET, TESTED ON ASVSPOOF2015 DEV AND EVAL SETS.

Method	2019	2015	
	Eval	Dev	Eval
Baseline LFCC+GMM [42]	9.57	19.82	15.91
Baseline CQCC+GMM [42]	8.09	47.72	39.90
Res-TSSDNet	1.64	39.42	42.52
Mixup, $\alpha = 0.1$, Res-TSSDNet	2.07	5.48	5.46
Mixup, $\alpha = 0.5$, Res-TSSDNet	2.29	3.50	5.75
Mixup, $\alpha = 1.0$, Res-TSSDNet	2.16	**0.71**	**1.95**
$M = 3$, 4-branch, Inc-TSSDNet	10.39	5.31	5.24
$M = 4$, 4-branch, Inc-TSSDNet	4.04	2.78	3.29
$M = 4$, 8-branch, Inc-TSSDNet	3.75	1.84	2.16
$M = 5$, 8-branch, Inc-TSSDNet	4.20	**1.31**	**1.96**

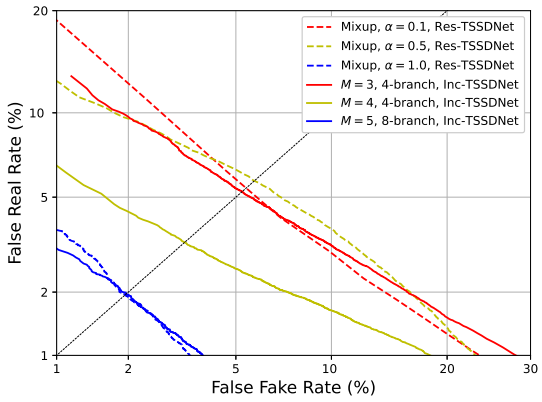


Fig. 4. DET curves of cross-dataset testing on ASVspoof2015 eval set.

Res-TSSDNet can significantly reduce the cross dataset EERs to less than 2%, while slightly sacrificing the performance on the original dataset. Further, all the Inc-TSSDNets have very attractive generalization capability even for the lightest model. The $M = 5$, 8-branch version yields the best cross-dataset performance with 1.96% eval EER. This is a significant score compared to the existing cross-dataset results as reported in [43]–[45]. Noticeably in [45], also trained on ASVspoof2019 training set and tested on ASVspoof2015, the use of the CQT based features could only achieve EERs greater than 20% (see Table 2 in [45]). For completeness, the detection error trade-off (DET) curves on ASVspoof2015 evaluation set using a few methods in Table III are provided in Fig. 4.

IV. CONCLUSION

We have shown that a light-weight end-to-end neural network, significantly different from the exiting front- and back-end pipeline, could achieve to date the best synthetic speech detection results. It reduces the ASVspoof2019 eval EER by a clear margin compared to much heavier networks fed by pre-transform inputs, sophisticated hand-crafted features plus MLP classifiers, or the fusion of many systems of such kinds. We have further shown via cross-dataset testing that the proposed networks could also generalize to unseen dataset. In the ongoing ASVspoof2021 challenge, a new speech deepfake (DF) detection task is introduced specially for synthetic deepfake speech detection, and end-to-end methods are being given more attention, e.g., the RawNet2 [46] is used as a baseline.

REFERENCES

[1] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden markov models," *Proc. IEEE*, vol. 101, no. 5, pp. 1234–1252, May 2013.

[2] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2021, pp. 1–15.

[3] X. Tian, S. W. Lee, Z. Wu, E. S. Chng, and H. Li, "An exemplar-based approach to frequency warping for voice conversion," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 10, pp. 1863–1876, 2017.

[4] Y. Gao, R. Singh, and B. Raj, "Voice impersonation using generative adversarial networks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018)*, Apr. 2018, pp. 2506–2510.

[5] S. Ö. Arık, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Proc. the 32nd International Conference on Neural Information Processing Systems (NeurIPS)*, 2018, pp. 10 019–10 029.

[6] J. Sanchez, I. Saratxaga, I. Hernáez, E. Navas, D. Erro, and T. Raitio, "Toward a universal synthetic speech spoofing detection using phase information," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 4, pp. 810–820, Apr. 2015.

[7] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Proc. Interspeech*, 2015.

[8] I. Saratxaga, J. Sanchez, Z. Wu, I. Hernaez, and E. Navas, "Synthetic speech detection using phase information," *Speech Communication*, vol. 81, pp. 30–41, 2016.

[9] T. B. Patel and H. A. Patil, "Significance of source–filter interaction for classification of natural vs. spoofed speech," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 644–659, 2017.

[10] ——, "Cochlear filter and instantaneous frequency based features for spoofed speech detection," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 618–631, 2017.

[11] D. Paul, M. Pal, and G. Saha, "Spectral features for synthetic speech detection," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 605–617, Jun. 2017.

[12] L. Wang, S. Nakagawa, Z. Zhang, Y. Yoshida, and Y. Kawakami, "Spoofing speech detection using modified relative phase information," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 660–670, 2017.

[13] M. Todisco, H. Delgado, and N. Evans, "Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification," *Computer Speech & Language*, vol. 45, pp. 516–535, 2017.

[14] M. Pal, D. Paul, and G. Saha, "Synthetic speech detection using fundamental frequency variation and spectral features," *Computer Speech & Language*, vol. 48, pp. 31–50, 2018.

[15] J. Yang, R. K. Das, and N. Zhou, "Extraction of octave spectra information for spoofing attack detection," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 12, pp. 2373–2384, Dec. 2019.

[16] X. Tian, X. Xiao, E. S. Chng, and H. Li, "Spoofing speech detection using temporal convolutional neural network," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016, pp. 1–6.

[17] H. Muckenhirn, P. Korshunov, M. Magimai-Doss, and S. Marcel, "Long-term spectral statistics for voice presentation attack detection," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 11, pp. 2098–2111, 2017.

[18] H. Yu, Z. Tan, Z. Ma, R. Martin, and J. Guo, "Spoofing detection in automatic speaker verification systems using dnn classifiers and dynamic acoustic features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4633–4644, Oct. 2018.

[19] J. Yang, R. K. Das, and H. Li, "Significance of subband features for synthetic speech detection," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 2160–2170, 2020.

[20] C. Zhang, C. Yu, and J. H. L. Hansen, "An investigation of deep-learning frameworks for speaker verification antispoofing," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 684–694, 2017.

[21] R. K. Das, J. Yang, and H. Li, "Long range acoustic features for spoofed speech detection," in *Proc. Interspeech*, 2019, pp. 1058–1062.

[22] Z. Chen, Z. Xie, W. Zhang, and X. Xu, "ResNet and model fusion for automatic spoofing detection," in *Proc. Interspeech*, Aug. 2017, pp. 102–106.

[23] Y. Qian, N. Chen, and K. Yu, "Deep features for automatic spoofing detection," *Speech Communication*, vol. 85, pp. 43–52, 2016.

[24] M. Adiban, H. Sameti, and S. Shehnepoor, "Replay spoofing countermeasure using autoencoder and siamese networks on asvspoof 2019 challenge," *Computer Speech & Language*, vol. 64, no. 101105, pp. 1–13, 2020.

[25] Y. Qian, N. Chen, H. Dinkel, and Z. Wu, "Deep feature engineering for noise robust spoofing detection," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 10, pp. 1942–1955, 2017.

[26] 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, pp. 1018–1022.

[27] C.-I. Lai, N. Chen, J. Villalba, and N. Dehak, "ASSERT: Anti-Spoofing with Squeeze-Excitation and Residual Networks," in *Proc. Interspeech 2019*, 2019, pp. 1013–1017.

[28] H. Zeinali, T. Stafylakis, G. Athanasopoulou, J. Rohdin, I. Gkinis, L. Burget, and J. Černocký, "Detecting Spoofing Attacks Using VGG and SincNet: BUT-Omilia Submission to ASVspoof 2019 Challenge," in *Proc. Interspeech*, 2019, pp. 1073–1077.

[29] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "STC antispoofing systems for the ASVspoof2019 challenge," in *Proc. Interspeech*, 2019, pp. 1033–1037.

[30] X. Li, N. Li, C. Weng, X. Liu, D. Su, D. Yu, and H. Meng, "Replay and synthetic speech detection with Res2Net architecture," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2021)*, 2021.

[31] X. Wang and *et al.*, "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, no. 101114, pp. 1–24, 2020.

[32] H. Muckenhirn, V. Abrol, M. Magimai-Doss, and S. Marcel, "Understanding and Visualizing Raw Waveform-Based CNNs," in *Proc. Interspeech 2019*, 2019, pp. 2345–2349.

[33] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[34] H. Muckenhirn, M. Magimai-Doss, and S. Marcel, "End-to-end convolutional neural network-based voice presentation attack detection," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*, 2017, pp. 335–341.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv, 1512.03385*, pp. 1–14, 2015.

[36] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[37] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilci, M. Sahidullah, and A. Sizov, "ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *Proc. Interspeech*, 2015, pp. 1–5.

[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. International Conference on Learning Representations (ICLR)*, 2015, pp. 1–14.

[39] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. International Conference on Learning Representations (ICLR)*, 2016, pp. 1–13.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. International Conference on Learning Representations (ICLR)*, 2017, pp. 1–15.

[41] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *Proc. International Conference on Learning Representations (ICLR)*, 2018, pp. 1–13.

[42] 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," in *Proc. Interspeech*, 2019, pp. 1008–1012.

[43] D. Paul, M. Sahidullah, and G. Saha, "Generalization of spoofing countermeasures: A case study with ASVspoof 2015 and BTAS 2016 corpora," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2047–2051.

[44] P. Korshunov and S. Marcel, "A cross-database study of voice presentation attack detection," in *Handbook of Biometric Anti-Spoofing–Presentation Attack Detection, 2nd Ed.* Springer, 2019, pp. 363–389.

[45] R. K. Das, J. Yang, and H. Li, "Assessing the scope of generalized countermeasures for anti-spoofing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6589–6593.

[46] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with RawNet2," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6369–6373.