

Betray Oneself: A Novel Audio DeepFake Detection Model via Mono-to-Stereo Conversion

Rui Liu¹, Jinhua Zhang¹, Guanglai Gao¹ and Haizhou Li^{2,3}

¹Inner Mongolia University, China

²Shenzhen Research Institute of Big Data, School of Data Science,
The Chinese University of Hong Kong, Shenzhen, China

³National University of Singapore, Singapore

liurui_imu@163.com, zjh_imu@163.com, csggl@imu.edu.cn, haizhouli@cuhk.edu.cn

Abstract

Audio Deepfake Detection (ADD) aims to detect the fake audio generated by text-to-speech (TTS), voice conversion (VC) and replay, etc., which is an emerging topic. Traditionally we take the mono signal as input and focus on robust feature extraction and effective classifier design. However, the dual-channel stereo information in the audio signal also includes important cues for deepfake, which has not been studied in the prior work. In this paper, we propose a novel ADD model, termed as M2S-ADD, that attempts to discover audio authenticity cues during the mono-to-stereo conversion process. We first projects the mono to a stereo signal using a pretrained stereo synthesizer, then employs a dual-branch neural architecture to process the left and right channel signals, respectively. In this way, we effectively reveal the artifacts in the fake audio, thus improve the ADD performance. The experiments on the ASVspoof2019 database show that M2S-ADD outperforms all baselines that input mono. We release the source code at <https://github.com/AI-S2-Lab/M2S-ADD>.

Index Terms: Audio DeepFake Detection, Stereo, Dual-Branch

1. Introduction

The Deepfake technology seeks to adopt deep learning technology to produce audio that people have never said, things that people have never done, and audio-visual content that has never existed. Examples of its applications include face-shifting technology, voice simulation, face synthesis and video generation [1]. However, such technology can be abused. More and more deepfake content has been used for malicious purposes, such as in the spreading of fake news and fraud cases [2]. This calls for effective detection of the false content.

With the rapid progress of Text-To-Speech (TTS) [3–5] synthesis and Voice Conversion (VC) techniques, one is able to impersonate another’s voice easily. The Audio Deepfake Detection (ADD) technology [6] is a task that determines whether the given audio is authentic or counterfeited, and it has attracted increasing attention recently. Note that conventional methods for ADD mainly focus on two directions, including 1) robust feature extraction and 2) effective model design. For the first direction, Gupta et al. [7] proposed a novel Cochlear Filter Cepstral Coefficients-based Instantaneous Frequency using Quadrature Energy Separation Algorithm (CFCCIF-QESA)

features, with excellent temporal resolution as well as relative phase information. Some recent works ingest raw audio [8] instead of hand-crafted acoustic feature and achieve robust ADD performance. For the second direction, researchers adopt Gaussian mixture model [9], convolution neural network [10], deep neural networks [11], recurrent neural networks [12] etc. to build the ADD architecture. Recently, the graph neural network [13] was proposed to learn the relationships between cues at temporal intervals.

All of the above work dealt with single channel (mono) speech, and shown that ADD is clearly achievable. We note that two types of audio signals, namely single and dual-channel stereo, can be converted to each other [14]. The stereo signals provide a unique perspective on the speech quality of the audio signal [15], which has not been studied in the prior work of ADD. For example, Tolooshams et al. [16] proposed a novel stereo-perception framework for speech enhancement. Compared with the traditional method with mono audio, using stereo can better preserve spatial images and enhance stereo mixture. Therefore, whether we can improve ADD performance by incorporating stereo information is the main concern of our work. Fortunately, we found some inspiration from the field of computer vision (CV), that is, a face recognition system aims to confirm the identity of a person by their face [17]. When performing face recognition, it is difficult to recognize the authenticity just rely on the realistic enough front face image [18]. However, the fake flaws will be exposed after rotating the face at a certain angle [19].

Inspired by the above research, we propose a novel ADD model, termed as M2S-ADD, that attempts to discover audio authenticity cues during the mono-to-stereo (M2S) conversion process (similar to face rotation operation). Specifically, the M2S-ADD model first converts the audio of the mono channel into stereo through a pre-trained M2S conversion model and then employs a dual-branch neural architecture to process the left and right channel signals, respectively. Finally, the information from dual channels is fused for the final decision. By incorporating two-channel stereo information compared to mono speech, the fake audio could easily reveal itself and be successfully detected by our model.

Experiments on the mainstream ASVspoof 2019 logical access (LA) database show that our M2S-ADD model with stereo speech outperforms all baselines with mono speech. The main contributions of this paper include: 1) We propose a novel audio deepfake detection model, termed as M2S-ADD, via mono-to-stereo conversion; 2) We design a dual-branch neural architecture to learn the correlation between the two-channel audio signals and conduct joint training to optimize our M2S-ADD model; To our best knowledge, this is the first study of audio deepfake detection with stereo speech data augmentation.

This research is funded by the High-level Talents Introduction Project of Inner Mongolia University (No. 10000-22311201/002), the National Natural Science Foundation of China (NSFC) (No. 62206136, 62271432), and Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funding Scheme (Project No. A18A2b0046).

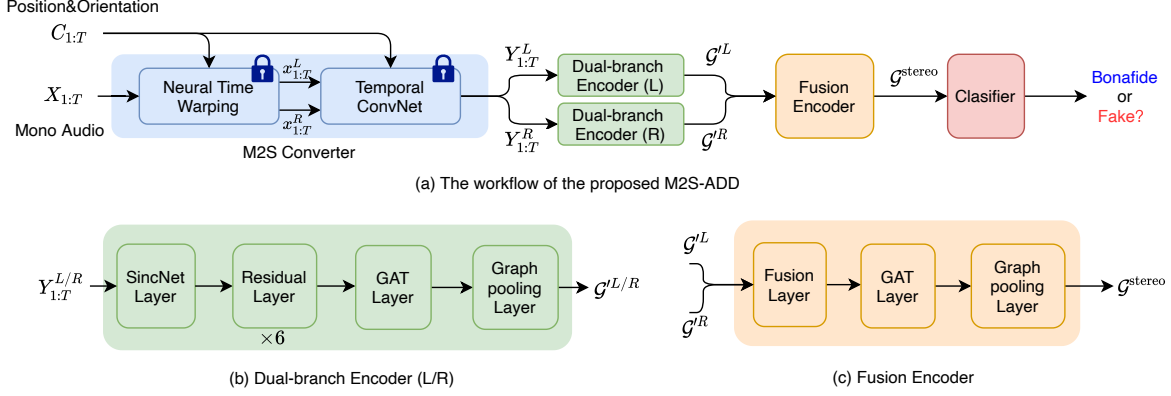


Figure 1: The proposed M2S-ADD model (a) that consists of M2S Converter, Dual-branch Encoder (b), Fusion Encoder (c), and Classifier. The blue locks mean the parameter of those modules are initialized by pretraining and fixed during M2S-ADD training.

2. M2S-ADD Neural Architecture

We propose a neural architecture, termed as M2S-ADD, as shown in Fig. 1, that consists of M2S converter, dual-branch encoder, fusion encoder, and classifier. The M2S converter converts the original mono audio into high-quality stereo, which includes two channel signals. The dual-branch encoder reads the two-channel audio and extracts the high-level feature representations for the left and right channels separately. The fusion encoder combines the two-channel feature representations together to facilitate the modeling of information between different dimensions. At last, the two-class classifier is to detect the authenticity (bonafide or fake). As M2S-ADD is expected to be better informed than the mono input system, it greatly reduces the model complexity.

2.1. M2S Converter

The M2S converter aims to transform the single-channel mono signal into dual-channel stereo signal. As shown in the blue panel of Fig.1(a), the M2S converter consists of neural time warping and temporal ConvNet. The neural time warping module learns an accurate warp from the source position to the listener's left and right ear while respecting physical properties like monotonicity and causality. The Temporal ConvNet module nuanced effects like room reverberations or head- and ear-shape related modifications to the signal.

Specifically, given a monaural (single-channel) signal $X_{1:T} = (X_1, X_2, \dots, X_T)$ of length T and its conditioning temporal signal $C_{1:T} = (C_1, C_2, \dots, C_T)$ that represents the position and orientation of source and listener respectively, neural time warping first adopts a CNN based WarpNet to read $C_{1:T}$ to predict a neural warfield ρ' and add it to the herometric warfield ρ . Then, the final warfield $\rho + \rho'$ was used to compute the two-channel warped signal $x_{1:T}^{L/R}$, where the channels represent left and right ear, by a recursive activation function. At last, the temporal ConvNet includes WaveNet like conditional temporal convolutions to input the $x_{1:T}^{L/R}$ and $C_{1:T}$ to reconstruct the expected dual-channel stereo signal $Y_{1:T}^{L/R}$.

The M2S converter share same architecture with the latest neural sound binauralization model [14], which includes more details. It's worth mentioning that we adopt such a pure end-to-end network as part of our M2S-ADD model to facilitate the entire pipeline. The parameters of the M2S converter need to be pre-trained and then fixed during M2S-ADD training to achieve a stable M2S conversion, as shown in the blue locks of Fig.1(a).

2.2. Dual-branch Encoder

Dual-branch encoder is devised to extract high-level and meaningful feature representation for dual-channel stereo signals. As shown in Fig.1(a) and (b), Dual-branch encoder (L) and Dual-branch encoder (R) share the same structure to process left and right channel data respectively. The dual-branch encoder is composed by one SincNet layer, six Residual layers, one graph attention network (GAT) layer and one Graph pooling layer. Note that one SincNet layer and six Residual layers were stacked to let the CNN based network deepen and introduce more nonlinearity to learn the detail clues in the left and right channels audio signal. The GAT mechanism has been proven to learn the relationships between artifacts in different sub-bands or temporal intervals [20]. Following this spirit, the GAT layer is used to aggregate the associated information by using the self-attention weight between information pairs and the graph pooling layer used to discard useless and duplicate information.

Specifically, the SincNet layer consists of Conv-1D layer, max-pooling layer, Batch Normalization (BN) and SeLU activation function. The residual layer consists of a Conv-2D layer with BN and SeLU and another Conv-2D layer with max-pooling. SincNet and residual layers of dual-branch encoder (L) or (R) read the left or right signal to extract high-level feature respectively:

$$\begin{aligned} h_{1:T}^{L'} &= f^L(Y_{1:T}^L) \\ h_{1:T}^{R'} &= f^R(Y_{1:T}^R) \end{aligned} \quad (1)$$

where f means the CNN network was stacked by the SincNet and the residual layers.

The GAT layer operates upon an input graph \mathcal{G} to produce an output graph \mathcal{G}' . Note that \mathcal{G} is formed from the high-level feature $h_{1:T}^{L/R'}$, and defined as $\mathcal{G}(N, \varepsilon, h')$. N is the set of nodes, also the number of temporal frames after spectral averaging. ε represents the edges between all possible node connections, including self-connections.

Then, GAT aggregates neighboring nodes using learnable weights via a self-attention mechanism. The attention weight $\alpha_{u,n}$ between nodes u and n are calculated according to:

$$\alpha_{u,n} = \frac{\exp(W(h'_n \odot h'_u))}{\sum_{w \in \mathcal{M}(n) \cup \{n\}} \exp(W(h'_n \odot h'_w))} \quad (2)$$

where $\mathcal{M}(n)$ refers to the set of neighboring nodes for the node n . W is the learnable weight and \odot means element-level multiplication.

The output graph \mathcal{G}' comprises a set of nodes o , where each node o is computed by:

$$o_n = \text{SeLU}(\text{BN}(m_n + h'_n))$$

$$m_n = \sum_{u \in \mathcal{M}(n) \cup \{n\}} \alpha_{u,n} h'_u \quad (3)$$

Graph pooling generates more discriminative graphs by selecting a subset of the most informative nodes. We follow [21] and generate the pooled graph representations \mathcal{G}'^L and \mathcal{G}'^R from the original \mathcal{G}^L and \mathcal{G}^R graphs of left and right channels, respectively. At last, both graph nodes are projected into the same dimensional space using an affine-transform to match the input node dimensionality of the fusion encoder.

2.3. Fusion Encoder and Classifier

The fusion encoder is used to exploit complementary information captured by the attention graphs of left and right channels. As shown in Fig.1(c), the fusion encoder includes fusion layer, GAT layer and the graph pooling layer. The fusion layer acts to combine \mathcal{G}'^L and \mathcal{G}'^R to a fused graph $\mathcal{G}'^{\text{stereo}}$ using element-wise multiplication:

$$\mathcal{G}'^{\text{stereo}} = \mathcal{G}'^L \odot \mathcal{G}'^R. \quad (4)$$

The GAT layer is applied to $\mathcal{G}'^{\text{stereo}}$ and the graph pooling is then used to generate a pooled graph $\mathcal{G}^{\text{stereo}}$. Unlike the GAT and graph pooling layers in dual-branch encoder were processed for left and right channel graphs, the GAT and graph pooling layers in this section performs further feature extraction on the fused stereo graph, which will further analyze useful cues for audio deepfake detection, avoiding the loss of useful information and the side effect of redundant information. At last, we use a fully-connected projection layer to obtain the final two-class prediction (bona fide or fake).

By carefully analyzing the left and right channel signals in two-channel stereo audio, our method finds more important fake audio cues than mono audio analysis methods, allowing for accurate fake audio detection.

3. Experiments

3.1. Dataset

M2S Converter Pretraining: An <mono, stereo> paired audio data with appropriately size is a prerequisite for M2S converter pretraining. We follow [14] and conduct pretraining on a total of 2 hours of paired mono and binaural data at 48kHz from eight different speakers, four male and four female. Specifically, a mannequin equipped with binaural microphones in its ears is treated as the listener. Participants were asked to walk around the mannequin in a circle with 1.5m radius and make a spontaneous conversation with it. The conditioning temporal signals $C_{1:T}$ for each audio are recorded from the realistic scenario¹ [14]. More detailed descriptions are referred to [14].

M2S-ADD Training: We use the official **ASvspoof 2019** logical access (LA) database² to validate the performance of M2S-ADD. The real audio in the LA database is directly selected from the VCTK corpus, while the fake audio in the database is obtained by using different TTS and VC systems. The sampling rate of all audios is 16kHz. The data in the database is divided into three subsets, training set, development

¹Pretraining dataset link: https://github.com/facebookresearch/BinauralSpeechSynthesis/releases/download/v1.0/binaural_dataset.zip

²<https://www.asvspoof.org/index2019.html>

Table 1: The configuration for our M2S-ADD model.

Layer	Input:64600 samples	Output shape
Dual-branch Encoder(L/R)		
SincNet Layer	Conv-1D(129,1,70) Maxpool-2D(3) BN&SeLU	(70,64472) (1,23,21490)
Residual Layer	$\left\{ \begin{array}{l} \text{Conv-2D}((2,3), 1,32) \\ \text{BN \& SeLU} \\ \text{Conv-2D}((2,3), 1,32) \\ \text{Maxpool-2D}((1,3)) \end{array} \right\} \times 2$ $\left\{ \begin{array}{l} \text{Conv-2D}((2,3), 1,64) \\ \text{BN \& SeLU} \\ \text{Conv-2D}((2,3), 1,64) \\ \text{Maxpool-2D}((1,3)) \end{array} \right\} \times 4$	(32,23,2387) (64,23,29)
GAT Layer	GAT Layer	left : (32, 23) right : (32, 29)
Graph Pooling Layer(1)	Graph pooling Projection	left : (32, 14) right : (32, 23) (32,12)
Fusion Encoder		
Fusion Layer	Element-wise multiplication	(32,12)
GAT Layer	GAT Layer	(16,12)
Graph Pooling Layer(2)	Graph Pooling Projection	(16,7) (1,7)
Classifier		
Classifier	FC(2)	2

set and evaluation set. Among them, the fake audio in the training and development set all come from 6 systems (A01-A06), while the fake audio in the evaluation set was created with 13 systems (A07-A19).

3.2. Experimental Setup

For M2S converter pretraining, we train it for 100 epochs using an Adam optimizer. It's worth mentioning that the original M2S converter takes a mono audio segment with a length of 9600 samples as input and then outputs the left and right channels with the same length. To achieve utterance-level mono-to-stereo conversion for ASVspoof 2019 LA database during M2S converter, we first split each audio into various segments with 64600 samples as the input, then merge the output to obtain the utterance-level left and right channels. Note that the conditioning temporal signals $C_{1:T}$ for each audio during mono-to-stereo conversion are randomly sampled from real data¹. For M2S-ADD model, the Dual-branch Encoder(L/R) consists of SincNet Layer, 6 Residual Layers, GAT Layer, and a Graph pooling Layer. In SincNet Layer, the input length of data is 64600 samples, and the number of filters is 70. In the Residual Layer, we use 32 and 64 filters in Conv-2D of two kinds of residual blocks respectively to further improve the generalization ability of the model to cope with unknown attacks. In the GAT Layer, we use absolute values to prevent useful information with negative values from being discarded. A more detailed summary of the configuration for our M2S-ADD model is shown in Table 1.

In order to ensure that the results of each training are consistent, we use the random number seed to initialize parameters, whose default value is 1234. The model has trained for 400 epochs, the batch size is 24; the learning rate is 0.0001; the weight decay rate is 0.0001; the loss function is Weighted Cross Entropy Loss. Our program runs in the environment of python 3.8, and the driver Version is 515.48.07, and the CUDA version is 11.7. We report the results in terms of Equal Error Rate (EER) [22] rather than the minimum normalized tandem detection cost function (min-tDCF) [23] since min-tDCF is more applicable to speaker verification task [22].

Table 2: Comparison of M2S-ADD with all baselines with mono input. (* means we retrained this graph-based model using our own GPU and hyperparameters as MS-ADD, like batch size, for a fair comparison and achieved different results from the original paper.)

System	EER (\downarrow)
ResNet [24]	3.72
LCNN [25]	5.06
LCNN-4CBAM [26]	3.67
Siamese CNN [10]	3.79
PC-DARTS [27]	4.96
ResNet18-GAT-S [20]	4.48
GMM [28]	3.50
ResNet18-GAT-T [20]	4.71
Resnet18-AM-Softmax [29]	3.26
LCNN-Dual attention [26]	2.76
SE-Res2Net50 [30]	2.50
MLCG-Res2Net50+CE [31]	2.15
Resnet18-OC-softmax [29]	2.19
Capsule network [32]	1.97
LCNN-LSTM-sum [33]	1.92
ResNet18-LMCL-FM [34]	1.81
MCG-Res2Net50+CE [35]	1.78
Res-TSSDNet [36]	1.64
* RawGAT-ST [37]	1.39
M2S-ADD (proposed)	1.34
w/o dual-branch	2.46

3.3. Comparative Study

This work is one of the first attempts to exploit stereo information in ADD field. We choose some state-of-the-art ADD systems, with mono audio as an input, as the benchmark. Table 2 illustrates a comparison of performance for the proposed M2S-ADD model and various baselines with mono input.

Note that the RawGAT-ST model is very similar to our model. For a fair comparison, we retrained it on our own GPU using the same hyperparameters as M2S-ADD, such as batch size. The results show that our model outperforms it in terms of EER. In a nutshell, the results show that our model outperforms all other models.

3.4. Ablation Study

To further validate our M2S-ADD model, we conduct an ablation experiment in this section. Specifically, to verify the validity of the dual-branch neural architecture, we reconstructed the mono signal into a stereo and then averaged the information from the left and right channels globally and fed it to the dual-channel encoder (L) or (R) for feature extraction. The results are shown in the last row of Table 2. We can find that abandoning the dual-branch mechanism to directly encode information from the left and right channels would disrupt the modeling of information interaction between them, resulting in a significant reduction in ADD performance.

3.5. Visualization Study

In order to check the effectiveness of our method more visually, we have tried to analyze it in a visualization study. As a visualization of the time domain signal is not intuitive enough, we converted the time domain signal into spectral features and then observed the fine-grained differences between bonafide and fake audio.

As shown in Fig.2(a) and (b), we list the spectral features for bonafide and fake audios in terms of mono and stereo channels. White and blue boxes are used to highlight the spectral details at specific locations for bonafide and fake audios respectively. As can be seen from the blue boxes for fake audio,

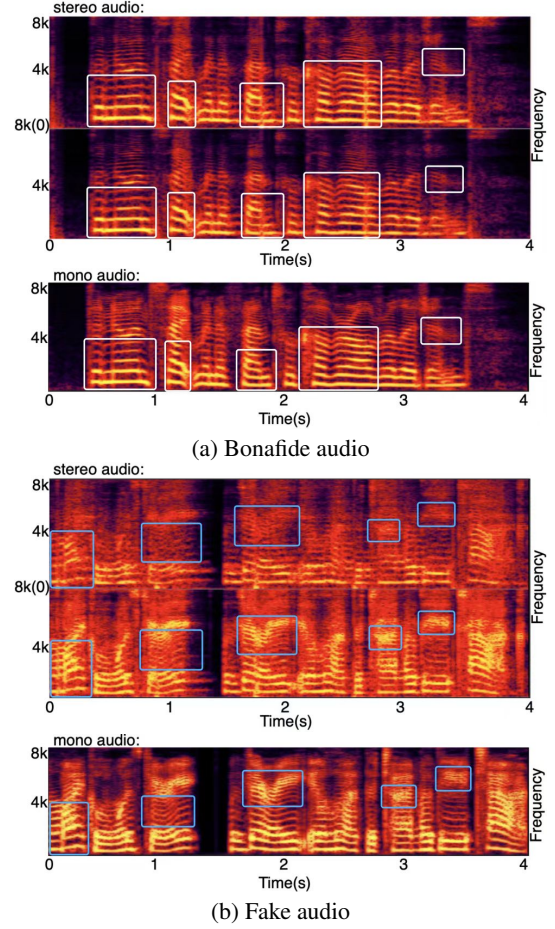


Figure 2: Comparison of visualization analysis in the spectral domain. (a) shows the spectral details of the mono and stereo signals for bonafide audio; (b) shows the spectral details for fake audio. Unlike the white boxes, the blue boxes show that the spectral artifacts are particularly exposed when the mono fake audio is converted to stereo audio.

the original spectral information is not fully reflected, and there is even a lot of noise, in the left and right channels of the stereo panel after mono-to-stereo conversion. Note that all the spectral details for bonafide audio, as in the white boxes, are preserved. In a nutshell, it was once again verified that converting mono audio to stereo audio can expose important cues among the fake audio, thus greatly reducing the difficulty of ADD.

4. Conclusions

In this work, we propose a novel audio deepfake detection model, denoted as M2S-ADD. The M2S-ADD model can handle dual-channel stereo signals and adopts dual-branch architecture to learn the subtle information among the left and right channel signals, thus effectively discovering audio authenticity cues during the mono-to-stereo conversion process. Experimental results demonstrate that our M2S-ADD can achieve accurate ADD performance for various attacks with the help of authenticity cues from dual-channel stereo signals. As per our knowledge, the proposed M2S-ADD is the first end-to-end ADD model that by exploiting stereo information. In future work, we will continue to optimize the model structure and exploit the detailed spectral information to further boost the ADD performance.

5. References

- [1] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *ArXiv*, vol. abs/2204.03458, 2022.
- [2] J. Mallet, L. Pryor, R. Dave, and M. Vanamala, "Deepfake detection analyzing hybrid dataset utilizing cnn and svm," *ArXiv*, vol. abs/2302.10280, 2023.
- [3] R. Liu, B. Sisman, G. I. Gao, and H. Li, "Decoding knowledge transfer for neural text-to-speech training," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 1–1, 2022.
- [4] R. Liu, B. Sisman, G. Gao, and H. Li, "Expressive tts training with frame and style reconstruction loss," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1806–1818, 2021.
- [5] R. Liu, B. Sisman, and H. Li, "Graphspeech: Syntax-aware graph attention network for neural speech synthesis," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6059–6063.
- [6] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, C. Wang, T. Wang, Z. Tian, Y. Bai, C. Fan *et al.*, "Add 2022: the first audio deep synthesis detection challenge," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 9216–9220.
- [7] P. Gupta, P. K. Chodingala, and H. A. Patil, "Replay spoof detection using energy separation based instantaneous frequency estimation from quadrature and in-phase components," *Computer Speech & Language*, vol. 77, p. 101423, 2023.
- [8] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6369–6373.
- [9] B. Chettri and B. L. Sturm, "A deeper look at gaussian mixture model based anti-spoofing systems," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5159–5163.
- [10] Z. Lei, Y. Yang, C. Liu, and J. Ye, "Siamese convolutional neural network using gaussian probability feature for spoofing speech detection," *Proc. Interspeech 2020*, pp. 1116–1120, 2020.
- [11] J. Li, M. Sun, and X. Zhang, "Multi-task learning of deep neural networks for joint automatic speaker verification and spoofing detection," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019, pp. 1517–1522.
- [12] Z. Chen, W. Zhang, Z. Xie, X. Xu, and D. Chen, "Recurrent neural networks for automatic replay spoofing attack detection," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 2052–2056.
- [13] H.-j. Shim, J. Heo, J.-h. Park, G.-h. Lee, and H.-J. Yu, "Graph attentive feature aggregation for text-independent speaker verification," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7972–7976.
- [14] A. Richard, D. Markovic, I. D. Gebru, S. Krenn, G. A. Butler, F. Torre, and Y. Sheikh, "Neural synthesis of binaural speech from mono audio," in *International Conference on Learning Representations*, 2021.
- [15] P. M. Delgado and J. Herre, "Objective assessment of spatial audio quality using directional loudness maps," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 621–625, 2019.
- [16] B. Tolooshams and K. Koishida, "A training framework for stereo-aware speech enhancement using deep neural networks," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6962–6966, 2021.
- [17] H. Du, H. Shi, D. Zeng, and T. Mei, "The elements of end-to-end deep face recognition: A survey of recent advances," 2020.
- [18] M. Abdollahnejad and P. X. Liu, "Deep learning for face image synthesis and semantic manipulations: a review and future perspectives," *Artificial Intelligence Review*, vol. 53, no. 3, 2020.
- [19] E. Winarno, I. Amin, S. Hartati, and P. W. Adi, "Face recognition based on cnn 2d-3d reconstruction using shape and texture vectors combining," *Indonesian Journal of Electrical Engineering and Informatics (IJEI)*, no. 2, 2020.
- [20] J. P. M. T. H. Tak, J.-w. Jung and N. Evans, "Graph attention networks for anti-spoofing," *Proc. INTERSPEECH*, 2021.
- [21] H. Gao and S. Ji, "Graph u-nets," in *international conference on machine learning*. PMLR, 2019, pp. 2083–2092.
- [22] T. Kinnunen, H. Delgado, N. Evans, K. A. Lee, V. Vestman, A. Nautsch, M. Todisco, X. Wang, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "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.
- [23] T. Kinnunen, K. A. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "t-DCF: a Detection Cost Function for the Tandem Assessment of Spoofing Countermeasures and Automatic Speaker Verification," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2018)*, 2018, pp. 312–319.
- [24] R. K. D. J. Yang, H. Wang and Y. Qian, "Modified magnitude-phase spectrum information for spoofing de- tectiion," *IEEE/ACM TASLP*, 2021.
- [25] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "Stc antispoofing systems for the asvspoof2019 challenge," *Proc. Interspeech 2019*, pp. 1033–1037, 2019.
- [26] S. Z. S. H. X. Ma, T. Liang and L. He, "Improved lightcnn with attention modules for asv spoofing detection," *2021 IEEE International Conference on Multimedia and Expo (ICME)*, vol. pp. 1–6, 2021.
- [27] W. Ge, M. Panariello, J. Patino, M. Todisco, and N. Evans, "Partially-connected differentiable architecture search for deepfake and spoofing detection," in *Interspeech 2021*. ISCA, 2021, pp. 4319–4323.
- [28] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "Spoofing attack detection using the non-linear fusion of sub-band classifiers," *Proc. Interspeech 2020*, pp. 1106–1110, 2020.
- [29] F. J. Y. Zhang and Z. Duan, "One-class learning to- wards synthetic voice spoofing detection," *IEEE Signal Processing Letters*, vol. pp. 937–941, 2021.
- [30] C. X. D. D. a. X. Li, N. Li, "Replay and synthetic speech detection with res2net architecture," *Proc. ICASSP*, vol. pp. 6354–6358, 2021.
- [31] H. L. X. L. X. Li, X. Wu and H. Meng, "Channel- wise gated res2net: Towards robust detection of synthetic speech attacks," *Proc. INTERSPEECH*, 2021.
- [32] Y. X. a. A. Luo, E. Li, "Acapsule network based approach for detection of audio spoofing attacks," *Proc. ICASSP*, vol. pp. 6359–6363, 2021.
- [33] X. Wang and J. Yamagishi, "A comparative study on re- cent neural spoofing countermeasures for synthetic speech detection," *Proc. INTERSPEECH*, 2021.
- [34] P. N. G. S. T. Chen, A. Kumar and E. Khoury, "Generalization of audio deepfake detection," *Proc. Speaker Odyssey Workshop*, vol. pp. 132–137, 2020.
- [35] X. Li, X. Wu, H. Lu, X. Liu, and H. M. Meng, "Channel-wise gated res2net: Towards robust detection of synthetic speech attacks," *ArXiv*, vol. abs/2107.08803, 2021.
- [36] G. Hua, A. B. J. Teoh, and H. Zhang, "Towards end-to-end synthetic speech detection," *IEEE Signal Processing Letters*, vol. 28, pp. 1265–1269, 2021.
- [37] H. Tak, J. W. Jung, J. Patino, M. Kamble, M. Todisco, and N. Evans, "End-to-end spectro-temporal graph attention networks for speaker verification anti-spoofing and speech deepfake detection," 2021.