Multi-View Collaborative Learning Network for Speech Deepfake Detection

Kuiyuan Zhang[1], **Zhongyun Hua**[1*], **Rushi Lan**[2], **Yifang Guo**[3], **Yushu Zhang**[4], **Guoai Xu**[1]
[1] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
[2] School of Computer Science and Information Security, Guilin University of Electronic Technology, China
[3] Alibaba Group, China
[4] School of Computing and Artificial Intelligence, Jiangxi University of Finance and Economics, China
zkyhitsz@gmail.com, huazyum@gmail.com, rslan2016@163.com, guoyifang@gmail.com
yushu@nuaa.edu.cn, xga@hit.edu.cn

Abstract

As deep learning techniques advance rapidly, deepfake speech synthesized through text-to-speech or voice conversion networks is becoming increasingly realistic, posing significant challenges for detection and raising potential threats to social security. This growing realism has prompted extensive research in speech deepfake detection. However, current detection methods primarily focus on extracting features from either the raw waveform or the spectrogram, often overlooking the valuable correspondences between these two modalities that could enhance the detection of previously unseen types of deepfakes. In this work, we propose a multi-view collaborative learning network for speech deepfake detection, which jointly learns robust speech representations from both raw waveforms and spectrograms. Specifically, we first design a **D**ual-**B**ranch **C**ontrastive **L**earning (DBCL) framework for learning different view features. DBCL consists of two branches that learn representations from the raw waveform or the spectrogram and utilizes contrastive learning to enhance inter- and inner-view correlations. Additionally, we introduce a **W**aveform-**S**pectrogram **F**usion **M**odule (WSFM) to exchange multi-view information for collaborative learning. In the feature learning process, WSFM converts features between views and merges them adaptively using waveform-spectrogram cross-attention. The final detection is conducted based on the concatenation of the waveform and spectrogram features. We conduct extensive experiments on four benchmark deepfake speech detection datasets, and the experimental results demonstrate that our method can achieve better detection performance than current state-of-the-art detection methods.

Introduction

The detection of deepfake speech has attracted increasing attention recently. With the rapid development of Text-To-Speech (TTS) and Voice Conversion (VC) techniques, one can easily use these tools to generate counterfeit speech that is hard for humans to recognize. This poses significant threats to multimedia security, as deepfake speech can be abused for impersonation attacks, reputation damage, online harassment, and other malicious activities (Franceschi-Bicchierai 2020; Burgess 2020). Therefore, it is urgent to develop robust methods for deepfake speech detection.

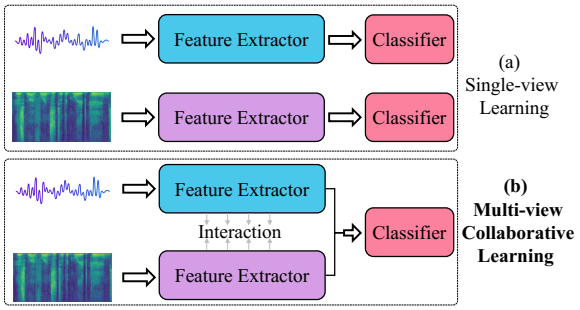

Figure 1: Illustration of two feature learning frameworks in speech deepfake detection. Previous single-view learning methods only learn speech representation from raw waveform or spectrogram, ignoring the correlation between two views. In contrast, our multi-view collaborative learning framework can learn speech representation from two views.

Recently, researchers have designed many elaborate networks for detecting deepfake speech. These methods have achieved remarkable performance by modeling speech forgery cues from either raw waveform or speech spectrogram, using features such as Mel-frequency cepstral coefficient (MFCC) (Wang et al. 2017), linear frequency cepstral coefficients (LFCC) (Ding et al. 2021), and Fourier bispectrum (AlBadawy, Lyu, and Farid 2019). However, existing detection approaches rely on single-view input for feature learning, ignoring the potential correlations between waveform and spectrogram features. This limitation may undermine the overall performance of these detection methods.

The complementary correlation between waveform and spectrogram features in speech signals can benefit feature learning for deepfake speech detection. Waveform features capture the temporal dynamics and amplitude variations inherent in speech, while spectrogram features model time-frequency characteristics in the frequency domain. By integrating information from both views, a more comprehensive understanding of the underlying signal properties can be achieved, thereby improving the discriminative power of the detection system. The work (Yang et al. 2024) incorporated multi-view representation using hand-craft spectrogram features and learning-based waveform features for

*Corresponding author

speech deepfake detection. The waveform features are obtained from speech models pre-trained on other speech tasks, such as speaker recognition or automatic speech recognition (ASR). However, the two sets of features are directly combined without any interaction, and their learning process remains independent. To date, multi-view learning directly from the waveform and spectrogram inputs has not been explored in previous works.

In this work, we propose a multi-view collaborative learning network for speech deepfake detection, which jointly learns features from raw waveform and speech spectrogram. Specifically, we design a **D**ual-**B**ranch **C**ontrastive **L**earning (DBCL) framework for learning different features from different views. The DBCL framework consists of two branches: a Transformer branch for learning waveform features and a convolutional neural network (CNN) branch for learning spectrogram features. To enhance both inter- and inner-view correlations, DBCL employs contrastive learning to the latent features produced by the branches. Inter-view contrastive learning emphasizes the contextual similarity between different view features of the same sample, while inner-view contrastive learning clusters real speech features and isolates fake speech features within each view. Besides, we propose a **W**aveform-**S**pectrogram **F**usion **M**odule (WSFM) to adaptively fuse multi-view features for collaborative learning. During the feature learning process within each branch, WSFM takes intermediate features from each view as input and applies cross-attention mechanisms to extract complementary information between the two views.The final classification feature for detection is obtained by combining the latent feature from two branches. Our contributions are:

- We develop a multi-view learning network for deepfake speech detection. Unlike previous methods that rely on a single view or simply combine pre-trained multi-view features, our approach collaboratively learns features from both the waveform and spectrogram views.
- We design a dual-branch contrastive learning framework that separately learns features from different views while enhancing both inter- and inner-view correlations. Additionally, we introduce a waveform-spectrogram fusion module to adaptively fuse the waveform and spectrogram features.
- Experiment results on three benchmark datasets demonstrate that our method outperforms state-of-the-art (SOTA) baselines across various synthesizer methods and datasets. We further validate the effectiveness of our approach through comprehensive ablation studies.

Related Works

Speech Deepfake Generation

Speech synthesis has been around for a long time. Currently, the most effective tools to generate deepfake speech are deep learning-based TTS and VC models. VC methods (Casanova et al. 2022; Wang et al. 2022b) take the speech as input and convert the voice style, such as timbre or pitch, to make the input speech sound like another person. TTS methods (Jiang et al. 2023; Oh et al. 2023) take the text as input and synthesize speech waveform at a specific voice style.

In the typical architecture of TTS and VC methods, vocoders are commonly employed to synthesize speech waveform from the input speech spectrogram. According to the model architecture, neural vocoders can be roughly divided into three categories: auto-regressive models (Oord et al. 2016; Kalchbrenner et al. 2018), diffusion models (Kim, Kim, and Yoon 2022; Popov et al. 2021), and GAN-based models (Kumar et al. 2019; Song et al. 2023). Recently, variational autoencoder (VAE) based TTS (Kim, Kong, and Son 2021; Lee et al. 2022) and VC (Lei et al. 2023) methods have been proposed. The most notable characteristic is that VAE-based methods can directly synthesize speech waveform from the feature embedding, which gives more considerable flexibility to control the voice style. In the future, more innovative synthesizer methods will emerge. Hence, it becomes imperative to devise robust deepfake speech detection methods capable of addressing potential challenges posed by emerging synthesizer methods.

Speech Deepfake Detection

The conventional speech deepfake detection approaches can be summarized into two categories: (1) learning feature from speech spectrogram and (2) learning feature from raw waveform. For the speech spectrogram, the commonly used spectrogram types include MFCC (Altalahin et al. 2023), LFCC (Ding et al. 2021), and so on. Using these handcrafted acoustic features, one can easily employ various machine learning methods or 2-Dimensional (2D) computer vision models, e.g., ResNet (Hua, Teoh, and Zhang 2021) and Transformer (Ulutas, Tahaoglu, and Ustubioglu 2023), to detect deepfake speech. To directly learn speech representation from raw waveform, some works designed elaborate 1-Dimensional (1D) CNN networks (Jung et al. 2020) or graph networks (Tak et al. 2021; Jung et al. 2022) and achieve robust detection performance. Besides, unsupervised pretrained audio models usually take the raw waveform as input to learn robust speech representation and can be applied to various downstream tasks. Therefore, some works (Lv et al. 2022; Wang et al. 2022a) combine unsupervised pretraining models to build speech deepfake detection systems.

Multi-View Learning

Multi-view learning can also be dubbed as multi-modal learning or multi-space representation learning. Recent research across various multi-modal tasks indicates that models gain advantages from multi-view learning, encompassing areas such as audio-visual deepfake detection (Zhang, Lin, and Xu 2024), specific person deepfake detection (Cozzolino et al. 2023), and multi-modal sentiment analysis (Zuo et al. 2023), etc. For example, AVA-CL (Zhang, Lin, and Xu 2024) designs an audio-visual attention layer with contrastive learning to learn audio-visual features for deepfake detection jointly. The work (Cozzolino et al. 2023) maps visual and audio features into different embedding spaces and fuses them for person-of-interest deepfake detection. IF-MMIN (Zuo et al. 2023) learns modality-invariant features for multi-modal emotion recognition.

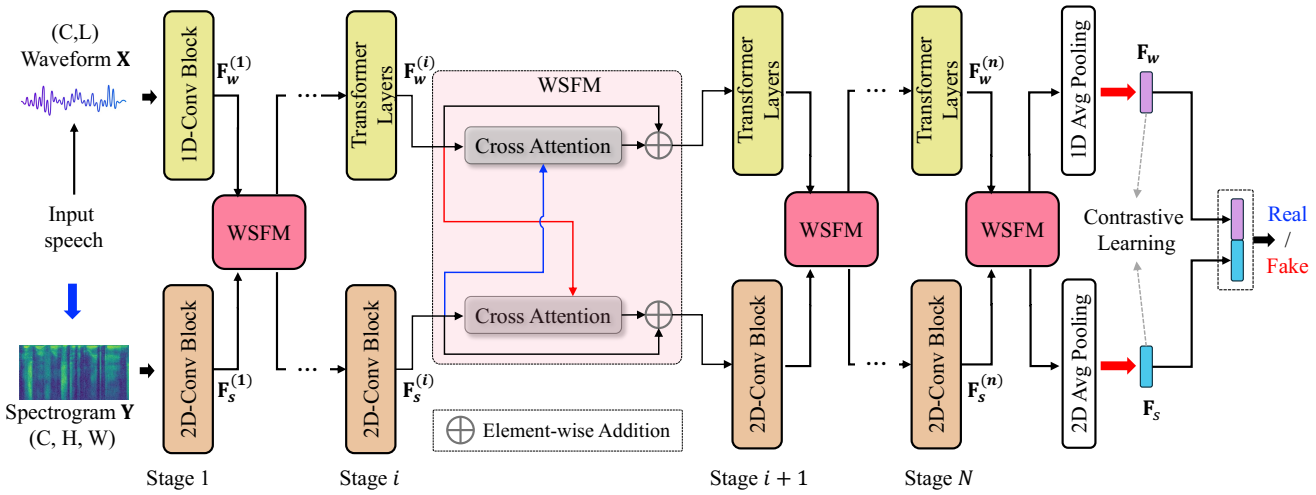

Figure 2: Network architecture of our method. We utilize two branches to learn waveform and spectrogram features, and we employ contrastive learning and WSFM to enhance the correlation between different views. The final classification is conducted based on the fusion of waveform and spectrogram features.

As for speech deepfake detection, existing multi-view methods only combine the latent feature of waveform and spectrogram views from separate audio models (Yang et al. 2024) or learn the dual-channel information just from the waveform view (Liu, Zhang, and Gao 2024). They cannot jointly learn speech representation from the waveform and spectrogram views. In contrast, we propose a multi-view learning framework in this work that collaboratively learns waveform and spectrogram features from different views.

Methodology

This section overviews our method's pipeline and details its components. We denote the input speech as $\mathbf{X} \in \mathbb{R}^{C \times L}$ and the speech spectrogram as $\mathbf{Y} \in \mathbb{R}^{C \times H \times W}$, where C is the number of channels, L represents the waveform length, and (H, W) correspond to the spectrogram's dimensions.

Overview Pipeline

Fig. 2 illustrates the overall pipeline of our method. Given the input speech $\mathbf{X}$, we apply spectrogram transform to it to obtain the frequency spectrogram $\mathbf{Y}$ as the supplementary feature of the raw waveform. Our network consists of two branches: one Transformer branch to learn waveform features and one 2D CNN branch to learn spectrogram features. Following the common design of network architecture (He et al. 2016), we divide each branch into N sequential processing stages, where each stage composes some Transformer layers or convolutional blocks. Between stages, we use the WSFM to learn complementary information from different views. To do so, the WSFM uses waveform-spectrogram cross-attention operations to fuse the input features of two views. After the stage N, we use the 1D and 2D average pooling layers to separately pool the fused features from N-th WSFM and flatten the pooling results to get the final waveform feature $\mathbf{F}_w$ and spectrogram feature $\mathbf{F}_s$. Finally, we concatenate the $\mathbf{F}_w$ and $\mathbf{F}_s$ to classify whether the input speech is bonafide or fake.

Spectrogram Transform

Some spectrogram-based detection methods (Lavrentyeva et al. 2019; Altalahin et al. 2023) use LFCC or MFCC spectrograms for feature learning. However, these spectrogram methods apply filter banks that discard specific frequency coefficients, potentially missing helpful information. To address this, our work utilizes log-scale spectrograms to retain and learn from all frequency coefficients. Concretely, the log-scale spectrogram is calculated as follows:

$$\mathbf{Y} = \ln(\mathrm{STFT}(\mathbf{X}) + 1e^{-7}), \tag{1}$$

where STFT denotes short-time Fourier transform. It should be noted that the resolution of $\mathbf{Y}$ is determined by the window size and the hop length of STFT.

Dual-Branch Contrastive Learning

Using average pooling and flattening operation, we can obtain the final waveform feature $\mathbf{F}_w$ and spectrogram feature $\mathbf{F}_s$ from two branches. To further enhance inter- and inner-view correlations, our DBCL introduces two kinds of contrastive learning tasks on $\mathbf{F}_w$ and $\mathbf{F}_s$.

Inner-View Contrastive Learning We utilize the inner-view contrastive learning to highlight the commonalities between speeches of the same category, i.e., bonafide or fake. Given a set of features $\mathbf{F} = \{\mathbf{f}^1, \mathbf{f}^2, \cdots, \mathbf{f}^n\}$ and their speech labels $\mathbf{y} = \{y^1, y^2, \cdots, y^n\}$, the inner-view contrastive task is defined as follows:

$$\begin{aligned} CL(\mathbf{F}, \mathbf{y}) = & \frac{1}{n^2} \sum_{i=1}^{n} \left(\sum_{j:y^i=y^j}^{n} \Big(1 - cos(\mathbf{f}^i, \mathbf{f}^j)\Big) \right. \\ & \left. + \sum_{j:y^i \neq y^j}^{n} \max\Big(cos(\mathbf{f}^i, \mathbf{f}^j) - \eta, 0\Big) \right), \end{aligned} \tag{2}$$

where $cos(a,b) = \frac{a \cdot b}{\|a\|\|b\|}$ donates the cosine similarity function, and η is the margin parameter to control the similarity for label-unmatched sample pairs. The contrastive loss of the inner-view contrastive learning task is calculated as follows:

$$\mathcal{L}_{inner} = CL(\{\mathbf{F}_w\}_0^B, \{y\}_0^B) + CL(\{\mathbf{F}_s\}_0^B, \{y\}_0^B) \quad (3)$$

where B is the batch size.

Inter-View Contrastive Learning We utilize inter-view contrastive learning to align the semantics between waveform and spectrogram features of the same sample. Concretely, the inter-view contrastive loss is defined as follows:

$$\mathcal{L}_{inter} = L(f(\mathbf{F}_w), \mathbf{F}_s) + L(\mathbf{F}_w, g(\mathbf{F}_s)) \quad (4)$$

where f and g are multi-layer perception (MLP) layers. The $L(\mathbf{p}, \mathbf{q})$ is computed as:

$$L(p,q) = -\frac{1}{B}\sum_{i=1}^{B}\left(\log \frac{e^{cos(\mathbf{p}_i, \mathbf{q}_i)/\tau}}{\sum_{k=1}^{N} e^{cos(\mathbf{p}_i, \mathbf{q}_k)/\tau}}\right), \quad (5)$$

where τ is the temperature coefficient. Note that these two views of features are still in different feature spaces. We only use the transform layers f and g in loss calculation.

Waveform-Spectrogram Fusion Module

We design the WSFM to enhance the connection between waveform and spectrogram features and learn robust speech representation from different views. Specifically, the WSFM has two adaptive fusion operations as follows.

Feature Fusion for Waveform Branch We first convert the input 2D spectrogram feature $\mathbf{F}_s^i \in \mathbb{R}^{C_i \times h_i \times w_i}$ into 1D features $\mathbf{W}^i$ since the waveform branch can only handle waveform-like 1D features. Specifically, we flatten the last two dimensions of $\mathbf{F}_s^i$ into a 1D vector and employ linear interpolation to the 1D vector to make it have equal length to waveform feature $\mathbf{F}_w^i$. Then, the feature fusion process for the waveform branch is calculated as follows:

$$\mathbf{F}_s^i = \text{softmax}\left(\frac{Q_{\mathbf{W}^i} K_{\mathbf{F}_s^i}^\top}{\sqrt{d_s}}\right)\mathbf{F}_s^i + \mathbf{F}_s^i, \quad (6)$$

where (Q, K) are linear transformation layers to make $\mathbf{W}^i$ and $\mathbf{F}_s^i$ have the same dimension d_s.

Feature Fusion for Spectorgram Branch We first convert the input 1D waveform feature $\mathbf{F}_w^i \in \mathbb{R}^{C_i \times l_i}$ into 2D spectrogram feature $\mathbf{S}^i$ since the spectrogram branch can only handle image-like 2D features. Specifically, we use a linear layer to project the time dimension of $\mathbf{F}_w^i$ into the spatial dimension of the spectrogram feature $\mathbf{F}_s^i$. We then regard the spatial dimension as the number of tokens for attention. The feature fusion process for the spectrogram branch is calculated as follows:

$$\mathbf{F}_w^i = \text{softmax}\left(\frac{Q_{\mathbf{S}^i} K_{\mathbf{F}_w^i}^\top}{\sqrt{d_w}}\right)\mathbf{F}_w^i + \mathbf{F}_w^i, \quad (7)$$

where (Q, K) are linear transforms to make $\mathbf{W}^i$ and $\mathbf{F}_s^i$ have the same dimension d_w.

Computational Efficiency The cross-attention fusion inherently has quadratic complexity, meaning longer sequences significantly increase training costs. To mitigate this issue, it is possible to choose sparse attention techniques as an alternative to reduce computation.

Speech Classification

We concatenate the waveform feature $\mathbf{F}_w \in \mathbb{R}^{C_n}$ and spectrogram feature $\mathbf{F}_s \in \mathbb{R}^{C_n}$ for the final classification:

$$\hat{y} = \sigma(\mathbf{H}_{cls}([\mathbf{F}_w || \mathbf{F}_s])). \quad (8)$$

where $||$ denotes the concatenation operation, $\mathbf{H}_{cls}$ represents a linear classification head, and σ is the sigmoid function that outputs probabilities in the range [0,1].

Loss Function

The classification loss $\mathcal{L}_{cls}$ is defined as follows:

$$\mathcal{L}_{cls} = \text{BCE}(\hat{y}, y), \quad (9)$$

where BCE denotes the binary cross-entropy loss. To impose supervisory guidance directly at each branch, we also add two supplementary classification losses based on the single-view feature:

$$\begin{aligned} \mathcal{L}_{cls}^1 &= \text{BCE}(\sigma(\mathbf{L}_1(\mathbf{F}_w)), y), \\ \mathcal{L}_{cls}^2 &= \text{BCE}(\sigma(\mathbf{L}_2(\mathbf{F}_s)), y), \end{aligned} \quad (10)$$

where $\mathbf{L}_1$ and $\mathbf{L}_2$ are two linear classifiers for the waveform and spectrogram branches. The total loss for model training is defined as:

$$\mathcal{L} = \gamma_1 * (\mathcal{L}_{cls} + \mathcal{L}_{cls}^1 + \mathcal{L}_{cls}^2) + \gamma_2 * (\mathcal{L}_{inner} + \mathcal{L}_{inter}) \quad (11)$$

where (γ_1, γ_2) are adjustment parameters for sub-losses.

Experiment Setting

Datasets

We evaluate our proposed method using deepfake speech datasets: ASVspoof2019 logical access (LA) (Wang et al. 2020) dataset, MLAAD (Müller et al. 2024) dataset, ASVspoof2021 LA dataset and ASVspoof2021 deepfake (DF) dataset (Liu et al. 2023). Table 1 and Table 2 list the number of used synthesizer methods and the number of real and fake samples of these datasets.

In the ASVspoof2021 DF dataset, the synthesizer methods are divided into five categories: neural vocoder autoregressive (AR), neural vocoder non-autoregressive (NAR), traditional vocoder (TRD), unknown (UNK), and waveform concatenation (CONC). Considering that there are too many fake samples in the testing split, we use all the real samples but only partial fake samples for evaluation. Specifically, the number of fake samples per category is equal to the number of bonafide samples in the testing subset we used.

Comparison Methods

We compare our method with the following baselines:

Details	ASVspoof 2019 LA dataset			ASVspoof 2021 DF dataset			ASVspoof 2021 LA dataset		
	Train	Validation	Test	Train	Validation	Test	Train	Validation	Test
Syn.	A01-A06	A01-A06	A07-A19	A07-A19	A07-A19	A07-A19 + Other 92s	A07-A19	A07-A19	A07-A19
Real	2580	2548	7355	4795	973	14869	1484	192	14816
Fake	22800	22296	63882	44530	9027	65273	12980	1808	133360
Total	25380	24844	71237	49325	10000	80142	14464	2000	148176

Table 1: The number of synthesizers (Syn.), real samples, and fake samples of deepfake speech datasets: ASVspoof2019 LA, ASVspoof2021 LA, and ASVspoof2021 DF. Note that we use only partial testing samples in ASVspoof2021 DF.

	EN	DE	ES	FR	IT	PL	RU	UK	Other
Syn.	20	7	5	7	6	5	5	5	8
Real	31239	5856	3913	5821	6708	3808	3710	3820	0
Fake	19000	6000	4000	6000	7000	4000	4000	4000	22000
Total	50239	11856	7913	11821	13708	7808	7710	7820	22000

Table 2: Details of each subset in the MLAAD speech deepfake datasets.

- LCNN (Lavrentyeva et al. 2019) and RawNet2 (Jung et al. 2020): They serve as baselines in the ASVspoof 2021 challenge. LCNN employs a lightweight 2D CNN architecture with LFCC features as input for detection, while RawNet2 utilizes a 1D CNN architecture to learn features directly from the waveform.
- AASIST (Jung et al. 2022): This graph network uses a novel heterogeneous stacking graph attention layer to learn speech representations from the raw waveform.
- LibriSeVoc (Sun et al. 2023): Taking RawNet2 (Jung et al. 2020) as the backbone, this method appends an auxiliary task to classify vocoders as regularization.
- Wav2Clip (Wu et al. 2022) and AudioClip (Guzhov et al. 2022). They are Contrastive Language-Image Pretraining (CLIP) based pre-trained models. Wav2Clip distills CLIP by projecting audio into a shared embedding space with images and text, while AudioClip incorporates the ESResNeXt audio model into CLIP.
- ABC-CapsNet (Wani, Gulzar, and Amerini 2024): This approach uses VGGNet (Simonyan and Zisserman 2015) to extract speech features from Mel-spectrogram and applies cascaded capsule networks to analyze complex speech data patterns.
- MPE (Wang et al. 2024): This method designs a multi-scale permutation entropy (MPE) feature to present the statistic characteristic and then employ an LCNN backend for classification based on the combination of MPE and LFCC features.
- ASDG (Xie et al. 2024): This method aims to enhance generalizability by learning an ideal feature space. To do so, it clusters real speech while isolating fake speech from various domains using single-side domain adversarial learning and triplet loss.

We train and test all baselines mentioned above, using their publicly available code or building the models exactly as described in their paper if no publicly available code.

Data Preprocessing

We preprocess the speech data to ensure consistency by converting all clips to mono-channel speech with a 16 kHz sampling rate. Each clip is standardized to 48,000 samples (three seconds). Clips shorter than three seconds are padded by repeating the speech, while longer clips have a three-second segment randomly extracted during training, and the middle segment is used for validation and testing. To balance bonafide and fake samples, we apply oversampling to the bonafide samples during training.

Implementation Details

Our spectrogram branch is the feature extractor of the ResNet18 (He et al. 2016), and our waveform branch is a 12-layer Transformer (Chen et al. 2022). We split the two branches into five stages, and the number of feature channels is set to (64, 128, 256, 512, 512) in the four stages of the spectrogram branch, while the number of feature channels is 768 in the Transfomer branch. We set the window size and hop length to 512 and 187 in the STFT, and the resulting transformed spectrogram has the size of (257, 257).

The η and τ in inner- and inter-view contrastive losses are set to 0.4 and 1.0 by default, respectively. The loss weights in the loss function are set to $\gamma_1 = 1.0$ and $\gamma_2 = 1.0$. During training, the batch size is set to 64, and the Adam optimizer with a weight decay of 0.01. The learning rate of the parameters of the classifiers is set to $1e^{-4}$, while that of other parameters is set to $5e^{-4}$.

To enhance feature diversity, we implement the following data augmentation strategies for all detection methods: randomly adding noise with signal-to-noise ratio (SNR) ranging from 10 to 120 dB and randomly applying pitch shifts. We employ the early stopping strategy to halt model training when there is no further improvement in the area under the ROC Curve (AUC) performance within three epochs for all detection methods. All experiments are conducted using the PyTorch framework on a computer equipped with a GTX 4090 GPU device.

Experiment Results

Cross-Method Evaluation

We conduct a cross-method evaluation on the ASVspoof2021 DF dataset. Specifically, all detection methods were trained and validated on the ASVspoof2021 LA dataset and subsequently tested on both the LA and DF datasets. The cross-method evaluation results, presented in

Method	LA Testing Subset	DF Testing Subset	Unseen Synthesizers in DF Testing Subset			
			AR	NAR	TRD	UNK
LCNN	98.16 / 6.53	83.22 / 26.00	81.31 / 27.65	82.63 / 26.26	84.09 / 25.56	84.75 / 24.51
RawNet2	99.06 / 4.17	83.91 / 25.37	78.89 / 29.11	81.19 / 26.94	91.25 / 16.47	84.98 / 25.04
RawGAT	99.13 / 4.11	88.86 / 20.01	81.57 / 26.44	88.85 / 19.11	**96.69** / 9.31	88.94 / 20.67
LibriSeVoc	99.39 / 3.17	82.90 / 25.79	79.14 / 29.15	80.67 / 27.30	89.66 / 18.14	82.87 / 25.85
AudioClip	98.09 / 6.35	79.95 / 28.48	77.90 / 30.14	79.62 / 29.08	83.90 / 24.80	78.86 / 28.84
Wav2Clip	98.31 / 7.01	84.52 / 24.33	80.36 / 27.69	78.84 / 28.87	90.78 / 17.77	86.90 / 22.05
AASIST	98.50 / 6.19	86.42 / 22.09	78.22 / 28.79	86.77 / 21.01	95.11 / 11.75	86.23 / 22.99
MPE	92.10 / 16.11	76.97 / 29.92	74.17 / 32.07	78.81 / 28.57	80.63 / 26.89	74.73 / 31.27
ABCNet	91.83 / 15.62	73.50 / 33.66	74.40 / 33.02	75.33 / 32.30	72.69 / 34.23	71.63 / 34.91
ASDG	97.85 / 6.09	80.51 / 28.66	79.62 / 29.45	80.44 / 28.01	81.66 / 28.19	80.43 / 28.82
Ours	**99.64 / 1.56**	**95.26 / 9.33**	**95.35 / 10.32**	**96.10 / 8.45**	95.52 / **8.62**	**94.17 / 10.00**

Table 3: AUC(↑) / EER(↓) (%) performances on the ASVspoof2021 LA and DF testing subsets. Note that we remove all the samples synthesized by A07-A19 synthesizers in the ASVspoof2021 DF subset, so there is no category CONC.

Model	MLAAD Full	MLAAD subsets				
		FR	IT	PL	RU	UK
LCNN	96.27 / 9.42	98.69 / 6.12	97.24 / 9.57	99.70 / 2.50	89.42 / 19.08	98.50 / 5.94
RawNet2	85.52 / 22.71	86.19 / 21.08	83.97 / 24.00	94.49 / 13.16	94.32 / 12.83	82.51 / 26.40
RawGAT	93.71 / 14.33	94.66 / 13.69	98.43 / 6.26	99.74 / 1.53	92.91 / 15.42	84.76 / 25.08
LibriSeVoc	83.90 / 23.48	80.17 / 28.35	87.87 / 20.47	96.23 / 11.06	92.32 / 15.60	88.75 / 17.83
AudioClip	92.21 / 16.80	93.13 / 13.43	91.81 / 17.89	98.93 / 6.02	88.64 / 20.35	91.29 / 17.60
Wav2Clip	94.79 / 12.85	97.89 / 5.98	99.32 / 3.81	98.13 / 7.67	77.82 / 31.78	96.74 / 10.98
AASIST	92.87 / 14.88	93.77 / 14.80	98.17 / 5.90	99.68 / 1.81	90.97 / 18.60	83.90 / 27.43
MPE	94.83 / 12.47	93.56 / 14.32	96.90 / 9.11	97.33 / 8.08	87.75 / 20.94	96.90 / 8.50
ABCNet	64.11 / 40.54	70.18 / 35.94	72.09 / 36.91	70.06 / 35.45	60.70 / 41.43	60.40 / 43.20
ASDG	94.75 / 10.73	97.27 / 8.37	97.37 / 7.24	99.22 / 3.53	81.86 / 28.75	96.74 / 7.23
Ours	**99.17 / 4.73**	**99.39 / 4.00**	**99.65 / 2.82**	**99.98 / 0.45**	**98.62 / 5.31**	**99.57 / 2.35**

Table 4: AUC(↑) and EER(↓) (%) performances on the unseen dataset and languages. All the models are trained on the EN, ES, and DE subsets in the MLAAD dataset.

Table 3, clearly demonstrate that our method outperforms the competition significantly in nearly all categories. On the LA dataset's testing subset, our method achieved an EER of 1.56%, surpassing all other models. When tested on the DF dataset, our method also records the best EER score of 9.15%. Moreover, our method consistently exhibited superior detection performance across all categories for the unseen AR, NAR, TRD, UNK, and CONC synthesizers on the DF dataset. These superior detection scores validate the proposed method's effective generalization capability in identifying both seen and unseen deepfake methods compared to other models.

Cross-Language and Cross-Dataset Evaluation

In this evaluation task, we train and validate all the detection models on the EN, DE, and ES subsets of the MLAAD dataset and test them on the remaining languages.

Table 4 shows the cross-language evaluation results, where our method outperforms the competition significantly in nearly all categories. When tested on the full testing set, our method can achieve an EER of just 4.73%, which outperforms other methods. Our method can still obtain superior performance when tested on the FR, IT, PL, RU, and UK subsets. These cross-evaluation results highlight the effective generalization ability of our method.

Robustness Evaluation

Real-world speech often contains noise and compression artifacts, affecting audio quality. Therefore, detection models must remain robust against these distortions.

Background Noises To evaluate the robustness against noise, we conducted experiments by introducing random background noise into each input speech sample. For this purpose, we utilized the Musan (Snyder, Chen, and Povey 2015) dataset, which offers a wide range of both technical and non-technical noise types. During testing, a noise file was randomly selected from the Musan dataset and added to each speech sample at a signal-to-noise ratio (SNR) of 20 dB. We present the evaluation results in Table 5. One can see that our method maintains a consistent level of performance even in the presence of noise.

Compression Artifacts We employ the ASVspoof2019 LA dataset in this task for training since its samples do not involve any compression. For testing, we select the ASVspoof 2021 DF dataset that contains samples com-

Model	Seen	Unseen Methods				Whole
		AR	NAR	TRD	UNK	
LCNN	26.34	31.57	32.76	26.48	27.62	29.70
RawNet2	20.89	28.42	26.41	19.67	24.52	25.17
RawGAT	13.92	22.23	16.97	10.23	16.38	17.16
Wave2Vec2	42.97	30.89	31.23	28.11	24.74	28.76
WaveLM	32.62	22.58	17.55	10.58	14.26	16.52
LibriSeVoc	18.79	28.90	25.47	13.84	22.53	24.04
AudioClip	19.90	26.64	27.58	20.00	25.36	25.37
Wav2Clip	19.76	28.19	28.89	21.43	27.15	26.76
AASIST	27.00	29.72	24.84	18.12	25.72	25.27
MPE	28.14	34.30	32.01	29.81	34.16	32.59
ABCNet	19.46	26.70	21.24	15.59	21.92	21.81
ASDG	26.58	35.60	36.15	31.26	34.11	34.70
Ours	**8.91**	**14.37**	**11.94**	**7.86**	**11.89**	**11.58**

Table 5: Robustness evaluation results (EER%) against background noise on the ASVspoof2021 DF dataset. Note that we remove all the samples synthesized by A07-A19 synthesizers when testing.

Model	2019 LA	2021 DF					
		Whole	AR	NAR	TRD	UNK	CONC
LCNN	11.37	23.90	29.56	29.67	23.25	13.30	12.56
RawNet2	11.38	25.31	32.36	29.10	14.61	23.88	15.50
RawGAT	4.88	20.35	26.43	22.70	8.90	17.80	17.39
LibriSeVoc	11.27	24.57	30.67	28.40	15.91	22.40	10.45
AudioClip	11.05	24.19	28.16	28.67	17.99	23.02	14.52
Wav2Clip	8.80	22.54	31.41	30.94	17.49	12.46	6.19
AASIST	4.13	19.02	26.46	20.12	9.31	17.39	14.51
MPE	15.22	30.02	33.55	30.48	26.47	32.08	21.35
ABCNet	6.88	21.24	28.57	22.19	11.02	20.79	13.58
ASDG	12.83	25.52	31.83	32.43	25.15	14.58	16.84
Ours	**1.22**	**10.20**	**14.27**	**12.07**	**4.98**	**7.99**	**6.01**

Table 6: Robustness evaluation results (EER%) against compression artifacts. All the models are trained and validated on the ASVspoof2019 LA dataset but tested on the ASVspoof2021 DF dataset.

pressed using various methods at different bitrates. The testing results on both ASVspoof2019 LA and ASVspoof2021 DF datasets are presented in Table 6. First, our method performs best with an EER of 1.22% on the ASVspoof2019 LA dataset. In addition, our method achieves a 10.20% EER on the ASVspoof2021 DF dataset. It also outperforms other methods in the AR, NAR, TRD, UNK, and CONC categories, demonstrating superior robustness against compression artifacts compared to the comparing approaches.

Ablation Study and Discussion

In this section, we conduct ablation studies to verify the effectiveness of some components of our method.

Contrastive Loss

In our DBCL, we employ two contrastive losses, $\mathcal{L}_{inner}$ and $\mathcal{L}_{inter}$, to enhance the inner- and inter-view correlations between the waveform and spectrogram features. The inner-view contrastive loss highlights feature similarity between same-class samples, while the inter-view contrastive loss aligns the semantic similarity for each specific sample. The ablation studies on these two losses are shown in Table 7. Without the inner-view contrastive loss $\mathcal{L}_{inner}$ and the inter-view contrastive loss $\mathcal{L}_{inter}$, the average detection EER performance will drop by about 4.9% and 5.5% respectively on the ASVspoof2021 DF dataset, showing the importance of inner- and inter-view correlations in model training.

Setting	$\mathcal{L}_{inner}$	$\mathcal{L}_{inter}$	WSFM	$\mathcal{L}^1_{cls}$ + $\mathcal{L}^2_{cls}$	EER
(a)	×	×	●	●	13.65
(b)	×	●	●	●	12.71
(c)	●	×	●	●	13.52
(d)	●	●	×	●	10.47
(e)	●	●	●	×	12.21
(f)	●	●	●	●	**7.77**

Table 7: Ablation studies of the constrastive losses, WSFM, and single-view classification losses. The EER (%) performance on the ASVspoof2021 DF dataset is reported.

WSFM

Embedded after each process stage, our WSFM allows adaptive interactions between waveform and spectrogram features. As shown in Table 7, it brings about 2.7% (10.47% vs. 7.77%) EER improvement on the ASVspoof2021 DF dataset, which proves the necessity of feature interaction in feature learning.

Single-View Detection Losses

Our loss function also calculates the detection loss $\mathcal{L}^1_{cls}$ and $\mathcal{L}^2_{cls}$ based on the single-view features. The introduction of these two losses can be regarded as imposing supervisory guidance directly at each branch. Benefiting from it, our method achieves about 4.5% (12.21% vs. 7.77%) improvement on the ASVspoof2021 DF dataset, as shown in Table 7.

Conclusion

In this work, we proposed a multi-view collaborative learning network for speech deepfake detection. Our method collaboratively learns multi-view features from the waveform and spectrogram views rather than just combining pretrained multi-view features for detection. Specifically, we utilized two branches to learn the waveform and spectrogram features separately and employed contrastive learning to highlight the inter- and inner-view correlations between different views. To directly fuse multi-view features, we further designed the WSFM that interchanges and adaptively integrates features from different views. Finally, we concatenate the waveform and spectrogram features learned from two CNN branches to build the final classification feature. Experiments on three public benchmarks demonstrated that our method has superior detection performance and generalization ability than all baselines.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB3104400, by the Shenzhen Higher Education Stability Support Program under Grant GXWD20231130114233001, by the National Natural Science Foundation of China under Grant 62071142, and by the Guangdong Basic and Applied Basic Research Foundation under Grant 2024A1515012299.

References

AlBadawy, E. A.; Lyu, S.; and Farid, H. 2019. Detecting AI-Synthesized Speech Using Bispectral Analysis. In *CVPR workshops*, 104–109.

Altalahin, I.; AlZu'bi, S.; Alqudah, A.; and Mughaid, A. 2023. Unmasking the Truth: A Deep Learning Approach to Detecting Deepfake Audio Through MFCC Features. In *2023 International Conference on Information Technology (ICIT)*, 511–518.

Burgess, M. 2020. Telegram Still Hasn't Removed an AI Bot That's Abusing Women. *Wired*.

Casanova, E.; Weber, J.; Shulby, C. D.; Junior, A. C.; Gölge, E.; and Ponti, M. A. 2022. Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone. In *International Conference on Machine Learning*, 2709–2720.

Chen, S.; Wang, C.; Chen, Z.; Wu, Y.; Liu, S.; Chen, Z.; Li, J.; Kanda, N.; Yoshioka, T.; Xiao, X.; Wu, J.; Zhou, L.; Ren, S.; Qian, Y.; Qian, Y.; Wu, J.; Zeng, M.; Yu, X.; and Wei, F. 2022. WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing. *IEEE Journal of Selected Topics in Signal Processing*, 1505–1518.

Cozzolino, D.; Pianese, A.; Nießner, M.; and Verdoliva, L. 2023. Audio-Visual Person-of-Interest DeepFake Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 943–952.

Ding, Y.-Y.; Lin, H.-J.; Liu, L.-J.; Ling, Z.-H.; and Hu, Y. 2021. Robustness of speech spoofing detectors against adversarial post-processing of voice conversion. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29: 3415–3426.

Franceschi-Bicchierai, L. 2020. Listen to This Deepfake Audio Impersonating a CEO in Brazen Fraud Attempt.

Guzhov, A.; Raue, F.; Hees, J.; and Dengel, A. 2022. Audioclip: Extending Clip to Image, Text and Audio. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 976–980.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hua, G.; Teoh, A. B. J.; and Zhang, H. 2021. Towards end-to-end synthetic speech detection. *IEEE Signal Processing Letters*, 28: 1265–1269.

Jiang, C.; Gao, Y.; Ng, W. W.; Zhou, J.; Zhong, J.; and Zhen, H. 2023. SeDepTTS: Enhancing the Naturalness via Semantic Dependency and Local Convolution for Text-to-Speech Synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 12959–12967.

Jung, J.-w.; Heo, H.-S.; Tak, H.; Shim, H.-j.; Chung, J. S.; Lee, B.-J.; Yu, H.-J.; and Evans, N. 2022. Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6367–6371.

Jung, J.-w.; Kim, S.-b.; Shim, H.-j.; Kim, J.-h.; and Yu, H.-J. 2020. Improved RawNet with Feature Map Scaling for Text-independent Speaker Verification using Raw Waveforms. *Proc. Interspeech*, 3583–3587.

Kalchbrenner, N.; Elsen, E.; Simonyan, K.; Noury, S.; Casagrande, N.; Lockhart, E.; Stimberg, F.; Oord, A.; Dieleman, S.; and Kavukcuoglu, K. 2018. Efficient neural audio synthesis. In *International Conference on Machine Learning*, 2410–2419.

Kim, H.; Kim, S.; and Yoon, S. 2022. Guided-tts: A diffusion model for text-to-speech via classifier guidance. In *International Conference on Machine Learning*, 11119–11133.

Kim, J.; Kong, J.; and Son, J. 2021. Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech. In *Proceedings of the 38th International Conference on Machine Learning*, 5530–5540.

Kumar, K.; Kumar, R.; De Boissiere, T.; Gestin, L.; Teoh, W. Z.; Sotelo, J.; De Brebisson, A.; Bengio, Y.; and Courville, A. C. 2019. MelGAN: Generative adversarial networks for conditional waveform synthesis. *Advances in Neural Information Processing Systems*, 32.

Lavrentyeva, G.; Novoselov, S.; Tseren, A.; Volkova, M.; Gorlanov, A.; and Kozlov, A. 2019. STC antispoofing systems for the ASVspoof2019 challenge. *arXiv preprint arXiv:1904.05576*.

Lee, S.-H.; Kim, S.-B.; Lee, J.-H.; Song, E.; Hwang, M.-J.; and Lee, S.-W. 2022. HierSpeech: Bridging the Gap between Text and Speech by Hierarchical Variational Inference using Self-supervised Representations for Speech Synthesis. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 16624–16636. Curran Associates, Inc.

Lei, Y.; Yang, S.; Wang, X.; Xie, Q.; Yao, J.; Xie, L.; and Su, D. 2023. UniSyn: an end-to-end unified model for text-to-speech and singing voice synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 13025–13033.

Liu, R.; Zhang, J.; and Gao, G. 2024. Multi-space channel representation learning for mono-to-binaural conversion based audio deepfake detection. *Information Fusion*, 105: 102257.

Liu, X.; Wang, X.; Sahidullah, M.; Patino, J.; Delgado, H.; Kinnunen, T.; Todisco, M.; Yamagishi, J.; Evans, N.;

Nautsch, A.; and Lee, K. A. 2023. ASVspoof 2021: Towards Spoofed and Deepfake Speech Detection in the Wild. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31: 2507–2522.

Lv, Z.; Zhang, S.; Tang, K.; and Hu, P. 2022. Fake Audio Detection Based On Unsupervised Pretraining Models. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 9231–9235.

Müller, N. M.; Kawa, P.; Choong, W. H.; Casanova, E.; Gölge, E.; Müller, T.; Syga, P.; Sperl, P.; and Böttinger, K. 2024. MLAAD: The Multi-Language Audio Anti-Spoofing Dataset. arXiv:2401.09512.

Oh, S.; Noh, H.; Hong, Y.; and Oh, I. 2023. RWEN-TTS: Relation-Aware Word Encoding Network for Natural Text-to-Speech Synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 13428–13436.

Oord, A. v. d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016. WaveNet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Popov, V.; Vovk, I.; Gogoryan, V.; Sadekova, T.; and Kudinov, M. 2021. Grad-tts: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning*, 8599–8608.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556.

Snyder, D.; Chen, G.; and Povey, D. 2015. MUSAN: A Music, Speech, and Noise Corpus. arXiv:1510.08484.

Song, K.; Zhang, Y.; Lei, Y.; Cong, J.; Li, H.; Xie, L.; He, G.; and Bai, J. 2023. DSPGAN: a GAN-based universal vocoder for high-fidelity TTS by time-frequency domain supervision from DSP. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5.

Sun, C.; Jia, S.; Hou, S.; and Lyu, S. 2023. AI-Synthesized Voice Detection Using Neural Vocoder Artifacts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 904–912.

Tak, H.; weon Jung, J.; Patino, J.; Kamble, M.; Todisco, M.; and Evans, N. 2021. End-to-end spectro-temporal graph attention networks for speaker verification anti-spoofing and speech deepfake detection. In *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 1–8.

Ulutas, G.; Tahaoglu, G.; and Ustubioglu, B. 2023. Deepfake audio detection with vision transformer based method. In *2023 46th International Conference on Telecommunications and Signal Processing (TSP)*, 244–247.

Wang, C.; He, J.; Yi, J.; Tao, J.; Zhang, C. Y.; and Zhang, X. 2024. Multi-Scale Permutation Entropy for Audio Deepfake Detection. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1406–1410.

Wang, C.; Yi, J.; Tao, J.; Sun, H.; Chen, X.; Tian, Z.; Ma, H.; Fan, C.; and Fu, R. 2022a. Fully Automated End-to-End Fake Audio Detection. In *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*, 27–33.

Wang, L.; Nakagawa, S.; Zhang, Z.; Yoshida, Y.; and Kawakami, Y. 2017. Spoofing speech detection using modified relative phase information. *IEEE Journal of selected topics in signal processing*, 11(4): 660–670.

Wang, Q.; Zhang, X.; Wang, J.; Cheng, N.; and Xiao, J. 2022b. Drvc: A framework of any-to-any voice conversion with self-supervised learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3184–3188.

Wang, X.; Yamagishi, J.; Todisco, M.; Delgado, H.; Nautsch, A.; Evans, N.; Sahidullah, M.; Vestman, V.; Kinnunen, T.; Lee, K. A.; Juvela, L.; Alku, P.; Peng, Y.-H.; Hwang, H.-T.; Tsao, Y.; Wang, H.-M.; Maguer, S. L.; Becker, M.; Henderson, F.; Clark, R.; Zhang, Y.; Wang, Q.; Jia, Y.; Onuma, K.; Mushika, K.; Kaneda, T.; Jiang, Y.; Liu, L.-J.; Wu, Y.-C.; Huang, W.-C.; Toda, T.; Tanaka, K.; Kameoka, H.; Steiner, I.; Matrouf, D.; Bonastre, J.-F.; Govender, A.; Ronanki, S.; Zhang, J.-X.; and Ling, Z.-H. 2020. ASVspoof 2019: A Large-Scale Public Database of Synthesized, Converted and Replayed Speech. *Computer Speech & Language*, 64: 101114.

Wani, T. M.; Gulzar, R.; and Amerini, I. 2024. ABC-CapsNet: Attention Based Cascaded Capsule Network for Audio Deepfake Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2464–2472.

Wu, H.-H.; Seetharaman, P.; Kumar, K.; and Bello, J. P. 2022. Wav2CLIP: Learning Robust Audio Representations from Clip. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4563–4567.

Xie, Y.; Cheng, H.; Wang, Y.; and Ye, L. 2024. Domain Generalization via Aggregation and Separation for Audio Deepfake Detection. *IEEE Transactions on Information Forensics and Security*, 19: 344–358.

Yang, Y.; Qin, H.; Zhou, H.; Wang, C.; Guo, T.; Han, K.; and Wang, Y. 2024. A Robust Audio Deepfake Detection System via Multi-View Feature. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 13131–13135.

Zhang, Y.; Lin, W.; and Xu, J. 2024. Joint Audio-Visual Attention with Contrastive Learning for More General Deepfake Detection. *ACM Trans. Multimedia Comput. Commun. Appl.*, 20(5).

Zuo, H.; Liu, R.; Zhao, J.; Gao, G.; and Li, H. 2023. Exploiting Modality-Invariant Feature for Robust Multimodal Emotion Recognition with Missing Modalities. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5.