# Fusing Multi-layer Features of the Pre-trained Model With Grouped Cross Attention for Spoofing Speech Detection

Yu Guan, Wu Guo, Jie Zhang, Zhijun Zhang
University of Science and Technology of China
E-mail: gypursue@mail.ustc.edu.cn, guowu@ustc.edu.cn, jzhang6@ustc.edu.cn , zhangzhijun@mail.ustc.edu.cn

***Abstract*—In this work we propose a Grouped Cross Attention (GCA) module to fuse the multi-layer features of pre-trained model (PTM) to improve the spoofing speech detection (SSD) performance. The multi-layer features of PTM are first concatenated, followed by dimensionality reduction. The GCA is then applied to focus on the prominent time-spectrum position of the features for SSD. Specifically, we divide the features into groups along the layer dimension, which serve as queries and values for the calculation of GCA. Each group is fed into the same multi-head attention module. Since the features from the topmost PTM layer is most discriminative in SSD, we set these features as the keys for GCA. The attended features are further combined with original dimension reduced features, which are finally used as the inputs of the back-end classifier. Results on three benchmark datasets (ASV2019LA, ASV2021LA, ASV2021DF) show the state-of-the-art performance achieved by the proposed method.**

## I. Introduction

With the applications of advanced deep learning techniques in text-to-speech (TTS) and voice conversion (VC), the quality of generated spoof speech has improved significantly. This poses a severe threaten to automatic speaker verification and other speech processing systems. To address this problem, spoofing speech detection (SSD) has recently emerged as a popular research topic. The SSD generally falls into two categories: hand-crafted feature based methods and end-to-end methods. The former typically follows the pipeline comprising a front-end feature extractor and a back-end classifier. Commonly-utilized features include short time Fourier transform (STFT) [1], linear-frequency cepstrum coefficients (LFCC) [2], constant-Q transform (CQT) [3], etc. The latter directly feed raw audio into a classification network, such as RawNet2 [4] and AASIST [5].

Recently, pre-trained models (PTMs), e.g., Wav2Vec2 [6], HuBERT [7], WavLM[8], have significantly boosted the performance of many downstream speech processing tasks, including speech recognition, speaker recognition and SSD [9], [10]. It was shown that a PTM trained on the diverse speech datasets can learn general representations beneficial for downstream tasks. In case PTMs are applied to downstream tasks, they can be used as feature extractor to replace traditional hand-crafted features and cascaded with task-specific backends (such as classifiers or embedding extractors) [9]. For example, previous work integrated multi-scale feature aggregation (MFA) and dynamic convolution operations into the anti-spoofing task using the features from the PTM [11].

However, these systems only utilized the features from the topmost layer of the PTMs. Generally speaking, PTMs consist of 12 or 24 transformer layers, and the feature representations vary across different layers. Intuitively, the features below the topmost layer can also provide complementary information for SSD. Inspired by this, Martin et al. fused the features from all layers (including those extracted by CNNs) via weighted integration, which are subsequently fed directly into the back-end classifier [12]. Pan et al. proposed an attentive merging method to leverage the multi-layer features of WavLM for SSD[13]. Further, Alvarez et al. proposed an adapter which combines the sequence representations from different Wav2Vec2 layers into a single embedding per frame for better SSD [14].Furthermore, recent studies on PTM feature fusion have also demonstrated the excellent performance of this approach[15], [16].

The aforementioned fusion methods can exploit the multi-layer features of PTM to some extent, but one can do more. Compared with natural speech, synthesized speech exhibits artifacts on some specific time-spectrum positions, which however vary across different TTS and VC algorithms. The methods in [11]–[14] calibrate the features from a layer using same value (or weight),in other words, these models focus on the features on each layers equally. which cannot attend those more discriminative time-spectrum positions.

In this work, we propose a Grouped Cross Attention (GCA) module that can both fuse multi-layer features and attend the discriminative time-spectrum positions. Specially, the proposed method also uses a PTM as feature extractor and the backend classifier is same with that in [11]. Similar with [17], we first collect the features from all layers of the PTM through concatenation, followed by dimensionality reduction and normalization. To focus on the prominent time-spectrum positions, GCA is applied to the dimension-reduced features. Since the features of the topmost PTM layer is most discriminative in SSD, we set these features as the key value for cross attention calculation. To reduce memory requirement and speedup the convergence of model training, we divide the transformer layers of PTM in groups, each of which is fed into the same multi-head attention module in GCA. The attended features are further combined with original dimension-reduced features by

979-8-3315-7206-8/25/$31.00 

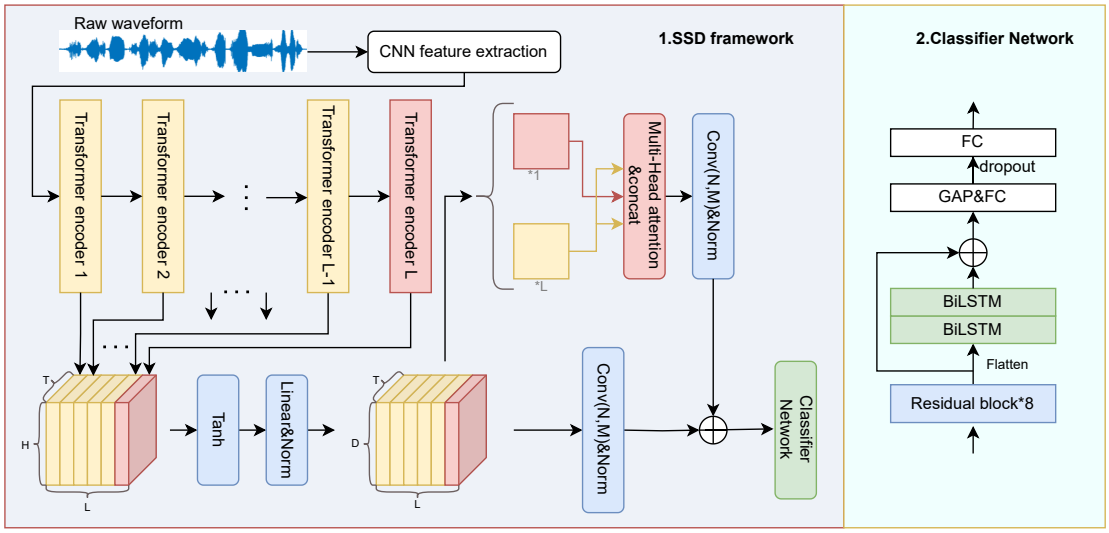


Fig. 1. The overall structure of the proposed spoofing speech detection (SSD) model

element-wise summation, which are finally used as features of the back-end classifier. We carry out experiments on three benchmark datasets, i.e., ASV2019LA[18], ASV2021LA and ASV2021DF [19], all achieving state-of-the-art performance.

The rest of this paper is structured as follows. Section 2 presents the proposed method in detail. Section 3 describes experimental setup, followed by results in Section 4. Finally, Section 5 concludes this work.

## II. Methodology

The overall architecture of the proposed SSD model is depicted in Figure 1, which includes a front-end feature extractor using the PTM and the GCA module and a cascaded backend classifier. The backend classification network is similar to that in [9]. The main contribution of this work lies in the proposed GCA module for fusing the multi-level features from different layers of the PTM.

### A. Pre-trained Model

We use the XLS-R [20] to extract multi-level features in this work. XLS-R is a self-supervised cross-lingual speech representation model based on wav2vec2, which scales the number of languages, the amount of training data as well as the model size. XLS-R is pre-trained on 436K hours of unannotated speech in 128 languages. Despite sharing the same architecture as wav2vec2, XLS-R benefits from a richer data foundation, enabling to extract more general features. This results in enhanced feature reliability and greater domain robustness. Therefore, we utilize XLS-R as the feature extractor for SSD. XLS-R combines a CNN encoder as a cochlear filter bank, with 24 transformer encoders that extract multi-level speech features.

### B. Grouped Cross Attention

The multi-layer features of the PTM capture various levels of speech clues, ranging from acoustic to linguistic levels. The proposed GCA can effectively use all these information for SSD. As shown in Figure 1, we first concatenate features from all the layers of the PTM, thereby obtaining $F \in \mathbb{R}^{N\times T\times H}$, where $N$, $T$ and $H$ represent the dimension of layer, time and channel (or frequency), respectively. To reduce the computation overhead and enhance the representational power of the features for subsequent GCA, $F$ is reduced from $H$ to $D$ in channel dimension to obtain $X$, given by

$$X = BN(tanh(F)W_f) \tag{1}$$

where $W_f \in \mathbb{R}^{H\times D}$ are learnable parameters and $BN(\cdot)$ stands for batch normalization. Furthermore,the function $tanh$ is not used as activation function, but are used to perform a kind of regularization of the data. In practice, the model is difficult to converge if we separately use cross attention operation to each layer, implying that we must have $N$ sets of parameters for attention calculation. To address this problem, we evenly divide $X$ into $N/K$ groups along the layer dimension and feed each group into the same multi-head attention module. Specifically, let $X_i \in \mathbb{R}^{T\times D}$ denote the dimension-reduced feature of each layer, and we look on $\{X_{K*n+1}, X_{K*n+2}, ..., X_{K*n+K}, n = 0, 1, ..., N/K-1\}$ as features of group $n$, which are used as query and value in attention calculation. Furthermore, we select the features in topmost layer as the key in attention calculation. The attended features $y_i$ can be obtained as

$$Q_{nk} = X_{K*n+k}W_{qn}, \tag{2}$$

$$K_{nk} = X_N W_{kn}, \tag{3}$$

$$V_{nk} = X_{K*n+k}W_{vn} \tag{4}$$

$$y_{K*n+k} = \sigma(Q_{nk}K_{nk}^T/\sqrt{d_k})V_{nk}, \tag{5}$$

TABLE I
THE PERFORMANCE ON THE ASV2021-LA DATASET, WHERE A07 TO A19 REFER TO 13 ATTACK ALGORITHMS USED FOR LA EVALUATION.

| Model | DA | ASV2021-LA | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | A07 | A08 | A09 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | pooled EER (%) / t-DCF |
| wav2vec2+AASIST[9] | × | 2.55 | 2.55 | 2.17 | 6.16 | 5.64 | 4.22 | 1.59 | 2.77 | 3.77 | 2.65 | 2.22 | 3.37 | 2.27 | 4.51/0.3114 |
| wav2vec2+MFA[11] | × | 0.90 | 1.14 | 0.82 | 3.97 | 4.91 | 1.90 | 0.34 | 1.43 | 2.23 | 1.06 | 1.30 | 1.71 | 1.14 | 2.56/0.2541 |
| Proposed | × | 1.19 | 0.74 | 0.66 | 1.84 | 1.52 | 1.53 | 0.80 | 1.15 | 1.65 | 0.97 | 0.80 | 0.89 | 1.09 | 1.47/0.2240 |
| wav2vec2+AASIST[9] | ✓ | 0.39 | 0.56 | 0.31 | 1.17 | 1.07 | 0.59 | 0.21 | 0.32 | 0.41 | 0.49 | 0.48 | 1.94 | 0.45 | 0.87/0.2081 |
| wav2vec2+MFA[11] | ✓ | 0.28 | 0.45 | 0.16 | 0.63 | 0.86 | 0.36 | 0.18 | 0.27 | 0.32 | 0.40 | 0.38 | 1.52 | 0.51 | 0.62/0.2019 |
| Proposed | ✓ | **0.27** | **0.23** | **0.05** | **0.23** | **0.25** | **0.18** | **0.02** | **0.08** | **0.16** | **0.16** | **0.22** | **0.60** | **0.26** | **0.32/0.1928** |

for $n = 0, \cdots, N/K - 1, k = 1, \cdots, K$, where $W_{qn} \in \mathbb{R}^{D \times d_k}$,$W_{kn} \in \mathbb{R}^{D \times d_k}$ and $W_{qn} \in \mathbb{R}^{D \times d_v}$ are learnable parameters, where $d_k$ and $d_v$ are set as the same value in the experiment, and $\sigma$ represents the softmax function. The attended features $Y = \{y_1, y_2, ..., y_N\}$ are further combined with the dimension-reduced features $X$ as

$$Z = f_1(X) \oplus f_2(Y), \tag{6}$$

where both $f_1(\cdot)$ and $f_2(\cdot)$ include a convolutional layer and a batch normalization layer, and $\oplus$ is element-wise addition. $f_1$ introduces a residual connection, enabling the multi-head attention module to learn residuals and thereby bringing the expected distribution closer to the initialized parameters. which can balance the convergence rates across different regions of the model. Finally, the composite feature $Z$ is fed into the cascaded classifier.

### C. Classification Network

The back-end classification network is shown in the right of Figure 1. The residual block is similar to the Res2NeXt block in [17], and we set the number of channels to 32 for implementation and append a squeeze-and-excitation (SE) [21] block after each residual block. This is followed by a two-layer BiLSTM with residual connections. The output layer consist of a global pooling layer and two fully-connected layers, together with an intervening dropout layer.

## III. EXPERIMENTAL SETUP

### A. Dataset and evaluation metrics

To evaluate the effectiveness and generalization of our proposed method, experiments are conducted on three distinct datasets, e.g., ASV2019LA, ASV2021LA, ASV2021DF. ASV2019LA is selected as the training set due to its widespread use in the field of SSD. Notably, ASVspoof 2021LA and DF only released the evaluation data, and the training and development sets are the same as that of ASVspoof 2019LA. The ASVspoof 2021 logical access (21LA) evaluation set comprises spoofed voices generated by various advanced TTS and VC algorithms, mixed with relatively clean speech. It contains about 180K utterances added with multiple codecs and transmission dynamics. ASV2021DF, on the other hand, incorporates in excess of 100 different spoofing attack algorithms and introduces a more sophisticated class of spoofing attacks generated by deep synthesis techniques, It collects about 600K utterances processed by various lossless codecs used for media storage. Equal error rate (EER) and minimum tandem detection cost function (min t-DCF) [22] are used as metrics to evaluate the SSD performance.

### B. Data augmentation

In order to improve the robustness in different scenarios, we incorporate recorded audio disturbances from the MUSAN [23] corpus and employ the RawBoost [24] of three noise algorithms to introduce various interference noises in accordance with the respective audio inputs. To simulate various room acoustics, we select room impulse responses (RIR) [25] from a public RIR dataset. These tricks are employed for training, with random selections and combinations.

In addition, we employ distinct data augmentation strategies for different tasks in our training sets. We utilize various encoding methods, e.g., MP3, OGG, AAC, a-law, μ-law and VoIP. Further, we apply speed perturbations to increase audio diversity. We randomly select these methods to generate corresponding speech files, ultimately obtaining a training set that is three times the size of the original training set, where 1/3 of the speech samples are raw audio, another 1/3 are processed with a random encoding methods, and the remaining 1/3 are subjected to random rate speed perturbations. Data enhancement during training is used on all datasets. For ASV2019LA, we only use raw audio file, while for ASV2021LA and ASV2021DF, the expanded training set are utilized.

### C. Implementation details

Experiments are conducted using PyTorch on RTX3090 GPUs. All the input speech signals are truncated or repeated into a fixed length of 80000 samples. The GCA reduces the original dimension of PTM from $H$ = 1024 to $D$ = 256, and utilizes the multi-head attention module with 8 heads and an embedding dimension of 256 to extract multi-scale features. The probability of dropout is 0.5. We employ the Adam [26] optimizer for model training, with a batch size of 16, an initial learning rate of 1e-6 and a weight decay of 1e-4. Considering the imbalance between the genuine and fake utterances in the training set, the training loss is computed using a weighted cross-entropy with weights of 0.9 for fake samples and 0.1 for real samples.

## IV. EXPERIMENTAL RESULTS

### A. Comparison with other SOTA methods

*1) Results on ASVspoof 2019LA and 2021LA datasets:* In Table 2, we show the performance of the proposed approach in comparison with other state-of-the-art (SOTA) methods on the

TABLE II
COMPARISON WITH SOTA SINGLE SYSTEMS ON THE ASVSPOOF 2019 AND 2021 LA EVALUATION SETS.

| Systems | ASV2021 | | ASV2019 |
|---|---|---|---|
| | EER (%) | min t-DCF | EER (%) |
| RawNet2[3] | 5.31 | 0.3099 | 4.62 |
| SE-Rawformer[27] | 4.98 | 0.3186 | 1.05 |
| DFSincNet[28] | 3.38 | 0.2732 | 0.52 |
| WavLM+AttM-LSTM[13] | 3.50 | - | 0.65 |
| W2v2+AASIST[9] | 0.87 | 0.2081 | 0.22 |
| W2v2+AASIST2[29] | 1.61 | - | 0.15 |
| W2v2+Conformer[10] | 0.87 | 0.2092 | - |
| W2v2+MFA[11] | 0.62 | 0.2019 | - |
| W2v2+Dual branch[30] | 0.56 | 0.2000 | **0.06** |
| Proposed | **0.32** | **0.1928** | **0.06** |

TABLE III
THE PERFORMANCE ON THE ASVSPOOF 2021 DF EVALUATION.

| Systems | EER (%) |
|---|---|
| RawNet2[3] | 22.38 |
| W2v2+LCNN[31] | 4.75 |
| W2v2+FC[32] | 4.12 |
| WavLM+AttM-LSTM[13] | 3.19 |
| W2v2+AASIST[9] | 2.85 |
| W2v2+AASIST2[29] | 2.77 |
| W2v2+Conformer[10] | 2.58 |
| W2v2+Dual branch[30] | 1.89 |
| Proposed | **1.56** |

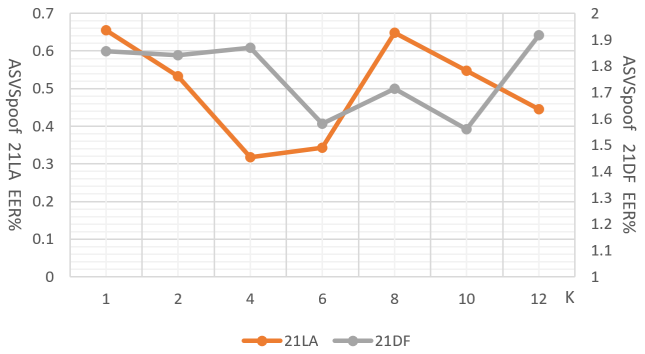


Fig. 2. Impact of GCA groups on ASVspoof2021 LA

TABLE IV
ABLATION STUDY ON DIFFERENT CLASSIFIER NETWORK.

| Systems | EER (%) | |
|---|---|---|
| | ASV2021LA | ASV2021DF |
| W2v2 + AASIST | 0.87 | 2.85 |
| + GCA | 0.58 | 1.84 |
| W2v2 + Res2Net | 0.73 | 2.52 |
| + GCA | 0.55 | 1.90 |
| W2v2 + Res2NeXt | 0.65 | 2.86 |
| + GCA | **0.32** | **1.56** |

ASV2019LA and ASV2021LA datasets. Since the pre-trained self-supervised front-end contains extra knowledge from massive out-of-database bonafide utterances, the Wav2vec2-based methods generally perform better in terms of EER and mint-DCF. In comparison with these Wav2vec2 based methods, It can be observed that our method still achieves the best results on the ASV2021LA dataset, with an EER of 0.32% and min t-DCF of 0.1928. Compared with our previous work [11], which follows the same setup except for the GCA fusion module, the proposed method achieves a 48% EER reduction (from 0.62% to 0.32%). Additionally, our approach can also obtain the SOTA performance on the ASV2019 dataset, with an EER of 0.06%.

Furthermore, we evaluate the performance of the proposed method across different spoofing algorithms in Table 1. It is evident that the proposed approach outperforms all other methods across all spoofing types, thereby validating its ability to enhance the backend model's utilization of features from PTM, enabling a more comprehensive exploitation of effective information. Moreover, our model also exhibits a lower EER without the use of data augmentation (DA), demonstrating the generalization capability of the proposed method.

*2) Results on ASVspoof 2021DF datasets:* In Table 3, we show the performance of the proposed method in comparison with SOTA approaches on ASV2021DF datasets. Notably, the DF evaluation set is a collection of bonafide and spoofing speech utterances processed by different lossless codecs that are typically used for media storage, and this set contains spoofed utterances generated with more than 100 different algorithms.Therefore,the Wav2vec2-based method s can substantially obtain better performance. The proposed method achieves an EER of 1.56%. As far as we know,this is the best result on this dataset.In summary, our approach attains the best performance to date across three datasets. The model demonstrates robustness in the face of varying deep fake algorithms, different transmission channels, and compression algorithms.

In the context of the ASVspoof2019 LA training dataset, which excludes variations related to codec, transmission (as in the ASVspoof2021 LA evaluation set), and compression (as in the ASVspoof2021 DF evaluation set), the experimental results demonstrate that the proposed method consistently achieves robust performance. This indicates the effectiveness and reliability of the approach under the specified conditions.

### *B. Ablation study*

In this section, we apply the proposed GCA module on other mainstream backend classifiers. In addition to Res2Next, we also consider AASIST and Res2Net. As shown in Table 4, we can also observe obvious improvements with AASIST and Res2Net backend classifiers. The results indicate that even though the structures of the two backend networks presented in the table are fundamentally distinct, the incorporation of the GCA module significantly enhances the model's capability to discriminate between genuine and spoofed voices. This reveals the universality of the proposed GCA module.

Finally, we analyze the impact of the number of groups in the proposed GCA method. For simplicity, only the results on ASV2021LA and ASV2021DF datasets are presented in Fig. 2, where $K$ represents the number of layers per group used for fusion. We can observe that the performance varies in terms of $K$, and the best performance can be achieved with $K$ =

4 and 10 on the ASV2021LA and ASV2021DF, respectively. We can then draw that multi-layer features of PTM can always provide complementary information for SSD. However, the optimal grouping configurations for SSD tasks vary significantly across different scenarios, suggesting that employing adaptive grouping strategies or incorporating modules such as Mixture of Experts (MoE)[33] could potentially enhance the performance of SSD model.

## V. Conclusion

In this paper, we proposed the GCA based fusion method to fully exploit the multi-layer features of the pre-trained model for spoofing speech detection. To facilitate the backend classification network in distinguishing task-beneficial feature information, we designed the GCA module that utilizes the top-level features as keys. Our method significantly outperform other SOTA methods on three public datasets, showing the generalization capacity and robustness against various spoofing attacks and scenarios. In the future, we will consider more powerful feature extraction and fusion methods for SSD.

## References


[1] A. Tomilov, A. Svishchev, M. Volkova, A. Chirkovskiy, A. Kondratev, and G. Lavrentyeva, "Stc antispoofing systems for the asvspoof2021 challenge," in *Proc. ASVspoof 2021 Workshop*, 2021, pp. 61–67.

[2] S. Cui, B. Huang, J. Huang, and X. Kang, "Synthetic speech detection based on local autoregression and variance statistics," *IEEE Signal Processing Letters*, vol. 29, pp. 1462–1466, 2022.

[3] X. Li, N. Li, C. Wengo, X. Liu, D. Su, D. Yu, *et al.*, "Replay and synthetic speech detection with res2net architecture," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2021, pp. 6354–6358.

[4] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2021, pp. 6369–6373.

[5] J.-w. Jung, H.-S. Heo, H. Tak, *et al.*, "Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2022, pp. 6367–6371.

[6] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems (NIPS)*, vol. 33, pp. 12 449–12 460, 2020.

[7] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 29, pp. 3451–3460, 2021.

[8] S. Chen, C. Wang, Z. Chen, *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[9] H. Tak, M. Todisco, X. Wang, M. Todisco, J. Jung, J. Yamagishi, *et al.*, "Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation," *arXiv preprint arXiv:2202.12233*, 2022.

[10] E. Roselló, A. Gómez-Alanís, A. Peinado, and Á. Gómez-García, "A conformer-based classifier for variable-length utterance processing in anti-spoofing," in *Proc.Interspeech*, 2023, pp. 5281–5285.

[11] H. Wu, J. Zhang, Z. Zhang, W. Zhao, B. Gu, and W. Guo, "Robust spoof speech detection based on multi-scale feature aggregation and dynamic convolution," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2024, pp. 10 156–10 160.

[12] J. M. Martín-Doñas and A. Álvarez, "The vicomtech audio deepfake detection system based on wav2vec2 for the 2022 add challenge," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP))*, IEEE, 2022, pp. 9241–9245.

[13] Z. Pan, T. Liu, H. B. Sailor, and Q. Wang, "Attentive merging of hidden embeddings from pre-trained speech model for anti-spoofing detection," *arXiv preprint arXiv:2406.10283*, 2024.

[14] J. M. Martín-Doñas, A. Álvarez, E. Rosello, A. M. Gomez, and A. M. Peinado, "Exploring self-supervised embeddings and synthetic data augmentation for robust audio deepfake detection," in *Interspeech 2024*, 2024, pp. 2085–2089.

[15] A. Guragain, T. Liu, Z. Pan, H. B. Sailor, and Q. Wang, "Speech foundation model ensembles for the controlled singing voice deepfake detection (ctrsvdd) challenge 2024," in *2024 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, 2024, pp. 774–781.

[16] J. Peng, O. Plchot, T. Stafylakis, L. Mošner, L. Burget, and J. Černockỳ, "An attention-based backend allowing efficient fine-tuning of transformer models for speaker verification," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, 2023, pp. 555–562.

[17] S. Gao, M. Cheng, K. Zhao, K. Zhao, X. Zhang, M. Yang, *et al.*, "Res2net: A new multi-scale backbone architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 43, no. 2, pp. 652–662, 2019.

[18] X. Wang, J. Yamagishi, M. Todisco, A. Nautsch, H. Delgado, N. Evans, *et al.*, "Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, p. 101 114, 2020.

[19] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgdo, T. Kinnunen, *et al.*, "Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild," *IEEE/ACM*

*Transactions on Audio, Speech, and Language Processing(TASLP)*, vol. 31, pp. 2507–2522, 2023.

[20] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, *et al.*, "Xls-r: Self-supervised cross-lingual speech representation learning at scale," in *Proc.Interspeech*, 2022, pp. 2278–2282.

[21] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7132–7141.

[22] T. Kinnunen, H. Delgado, N. Evans, *et al.*, "Tandem assessment of spoofing countermeasures and automatic speaker verification: Fundamentals," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2195–2210, 2020.

[23] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[24] H. Tak, M. Kamble, J. Patino, M. Todisco, and N. Evans, "Rawboost: A raw data boosting and augmentation method applied to automatic speaker verification anti-spoofing," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2022, pp. 6382–6386.

[25] T. Ko, V. Peddinti, D. Povey, M. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2017, pp. 5220–5224.

[26] Kingma and P. Diederik, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[27] X. Liu, M. Liu, L. Wang, K. Lee, H. Zhang, and J. Dang, "Leveraging positional-related local-global dependency for synthetic speech detection," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP))*, IEEE, 2023, pp. 1–5.

[28] B. Huang, S. Cui, J. Huang, and X. Kang, "Discriminative frequency information learning for end-to-end speech anti-spoofing," *IEEE Signal Processing Letters*, vol. 30, pp. 185–189, 2023.

[29] Y. Zhang, J. Lu, Z. Shang, W. Wang, and P. Zhang, "Improving short utterance anti-spoofing with aasist2," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2024, pp. 11 636–11 640.

[30] H. Wu, W. Guo, Z. Zhang, *et al.*, "Spoofing speech detection by modeling local spectro-temporal and long-term dependency," in *Proc. Interspeech 2024*, 2024, pp. 507–511.

[31] X. Wang and J. Yamagishi, "Investigating self-supervised front ends for speech spoofing countermeasures," *arXiv preprint arXiv:2111.07725*, 2021.

[32] X.Wang and J.Yamagishi, "Spoofed training data for speech spoofing countermeasure can be efficiently created using neural vocoders," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, IEEE, 2023, pp. 1–5.

[33] D. Lepikhin, H. Lee, Y. Xu, *et al.*, "Gshard: Scaling giant models with conditional computation and automatic sharding," *arXiv preprint arXiv:2006.16668*, 2020.