Interspeech 2024
1-5 September 2024, Kos, Greece

Temporal-Channel Modeling in Multi-head Self-Attention for Synthetic Speech Detection

Duc-Tuan Truong[1], *Ruijie Tao*[2], *Tuan Nguyen*[3], *Hieu-Thi Luong*[1], *Kong Aik Lee*[4], *Eng Siong Chng*[1]
[1]Nanyang Technological University, Singapore [2]National University of Singapore, Singapore
[3]Institute for Infocomm Research (I^{2}R), A*STAR, Singapore
[4]The Hong Kong Polytechnic University, Hong Kong
truongdu001@e.ntu.edu.sg, ruijie@nus.edu.sg, stunvat@i2r.a-star.edu.sg,
hieuthi.luong@ntu.edu.sg, kong-aik.lee@polyu.edu.hk, aseschng@ntu.edu.sg

Abstract

Recent synthetic speech detectors leveraging the Transformer model have superior performance compared to the convolutional neural network counterparts. This improvement could be due to the powerful modeling ability of the multi-head self-attention (MHSA) in the Transformer model, which learns the temporal relationship of each input token. However, artifacts of synthetic speech can be located in specific regions of both frequency channels and temporal segments, while MHSA neglects this temporal-channel dependency of the input sequence. In this work, we proposed a Temporal-Channel Modeling (TCM) module to enhance MHSA's capability for capturing temporal-channel dependencies. Experimental results on the ASVspoof 2021 show that with only 0.03M additional parameters, the TCM module can outperform the state-of-the-art system by 9.25% in EER. Further ablation study reveals that utilizing both temporal and channel information yields the most improvement for detecting synthetic speech[1].

Index Terms: synthetic speech detection, attention learning, ASVspoof challenges

1. Introduction

Powered by advanced deep generative neural networks, recent text-to-speech (TTS) and voice conversion (VC) systems have the ability to generate highly realistic synthetic human voices. Although this application can benefit many areas including data augmentation [1], criminals can utilize these fake speeches for malicious purposes leading to financial fraud, political conflict, and impersonation. Due to that, synthetic speech detection has been an active research field [2, 3]. To capture local synthetic artifacts, convolutional neural networks (CNNs) have conventionally served as the foundational architecture for SSD models. This approach covers a wide range of CNNs including LCNNs [4, 5], residual-connected ResNet [6, 7], and other variants [8, 9]. However, CNN-based models exhibit limitations in capturing the long-range dependencies of the input sequence. To overcome this, numerous studies employ Transformer models [10, 11, 12], yielding improved performance over CNN-based SSD models.

Notably, the recent SSD model [13], which combines the rich sequence representation of a self-supervised learning (SSL) model XLSR and the transformer-based Conformer architecture, achieves the state-of-the-art result in the ASVspoof 2021 corpus. This improvement can be attributed to the powerful modeling capability of the multi-head self-attention (MHSA) mechanism. It is conjecture that artifact details of synthetic speech can be located in specific regions of both temporal and spectral domain [14, 15, 16]. Therefore, incorporating the relationship between temporal and spectral information can provide a more complete and accurate representation for detecting artifacts in synthetic speech. By leveraging the temporal and spectral dependencies, several SSD systems [17, 18] exhibit improved capabilities in detecting deepfake speech. However, the MHSA in transformer-based SSD systems focuses on computing dot product between input tokens along the temporal dimension, hence it may overlook the dependencies between the temporal and channel dimensions of input sequences, which can be crucial for SSD tasks.

To better leverage the temporal and channel interaction of the input sequence for the XLSR-Conformer system, we propose the Temporal-Channel Modeling (TCM) module in the multi-head self-attention of the Conformer model. Our TCM module is based on the *head tokens* design, in which each head token represents the information on the channel dimension. The idea of head tokens is first proposed in [19] to enhance the interaction between the representation of attention heads in the MHSA and have improved the performance of Vision Transformer trained in the small-scale image classification dataset. However, in this work, head tokens aim to facilitate the correlation between temporal and channel dependencies by interacting them with the temporal tokens during MHSA. We also modify the original head token design by enriching the classification token with both temporal and channel information. The proposed TCM module, wherein with a marginal increase in parameters, improves the performance of the state-of-the-art XLSR-Conformer system on the ASV2021 eval set. Through empirical evaluation of the contribution of each component in the TCM module, temporal information from input tokens and channel information from head tokens both play an important role in the improvement of the TCM module.

2. Method

2.1. Baseline XLSR-Conformer

We adopt the state-of-the-art XLSR-Conformer [13] as our baseline architecture. As illustrated in Figure 1.a, it leverages the pre-trained XLSR [20], a variant of the wav2vec 2.0 model. Benefiting from the large-scale architecture and training on extensive data in an SSL manner, SSL models including XLSR can extract rich speech representations that have been useful for numerous speech tasks [21, 22, 23, 24, 25] including synthetic speech detection [26]. XLSR comprises two main components: a CNN front-end to transfer the 1D raw waveform into 2D temporal-channel representation, and 24 transformer encoder layers for capturing the global relationship of the speech. The

The corresponding author is Ruijie Tao

[1]Code and pre-trained models are available at https://github.com/ductuantruong/tcm_add

10.21437/Interspeech.2024-659

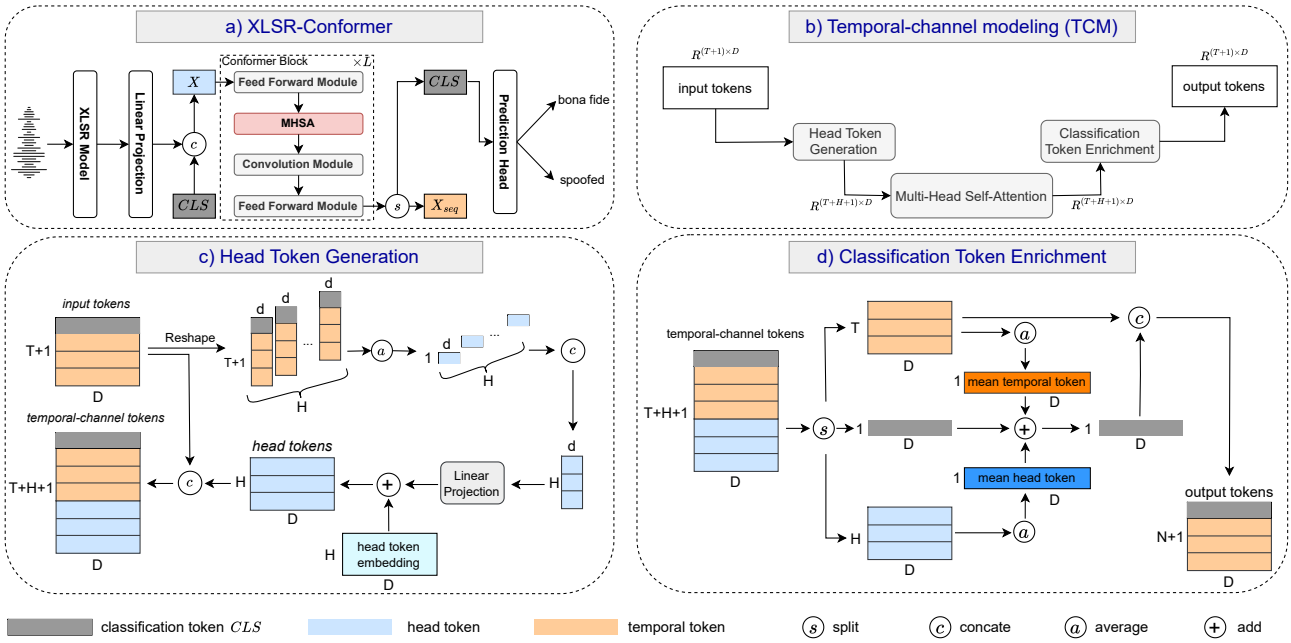

Figure 1: *The overall architecture of the baseline XLSR-Conformer and our proposed temporal-channel modeling (TCM) module. The TCM module is used to replace the multi-head self-attention (MHSA) of each Conformer block in the baseline XLSR-Conformer. The TCM module architecture includes three main parts: Head Token Generation, Multi-Head Self-Attention, and Classification Token Enrichment. The objective of TCM is to generate the head token for channel information and then integrate the temporal-channel dependency into the original temporal tokens for better synthetic speech detection.*

shape of the output speech representation is $(T \times D)$, where T denotes the temporal length and D is the channel dimension of XLSR representation.

After that, the XLSR representation is projected to D-dimensional and concatenated with the learnable classification token CLS to form an input sequence $X \in \mathbb{R}^{(T+1)\times D}$ for the Conformer model. The Conformer model consists of L Conformer blocks, each Conformer block includes the MHSA, feed-forward module, and the additional Convolutional layer to capture local dependencies within the speech representation. Finally, the CLS token is detached from the Conformer model's output representation to determine whether the input speech is bona fide or spoof.

2.2. Temporal-Channel Modeling module

The study of [19] introduces the concept of head token design, which initially focuses on fostering interaction between attention heads in multi-head self-attention (MHSA) and has improved the performance of image classification models trained on limited datasets. While the Temporal-Channel Modeling (TCM) approach is inspired by the head token design, its goal is to assist multi-head self-attention in capturing temporal-channel dependencies which can be essential for detecting synthetic speech. The proposed TCM module replaces the original MHSA of each Conformer block in the baseline model. As shown in Figure 1.b, the TCM architecture comprises three parts: Head Token Generation, Multi-Head Self-Attention, and Classification Token Enrichment. Similar to MHSA, TCM will not change the shape of the input and output token sequences for each Conformer block.

2.2.1. Head Token Generation

The Temporal-Channel Modeling module begins with the Head Token Generation component, designed to generate head tokens that represent the channel information of the input. These tokens interact with temporal information in subsequent steps. As shown in Figure 1.c, the input sequence of the Head Token Generation component consists of classification token CLS and temporal tokens $X \in \mathbb{R}^{(T+1)\times D}$. It first undergoes the head token generation process where X is reshaped into H segments of $d = D/H$ dimensions along the channel axis, where H is the number of attention heads in MHSA. Subsequently, each segment undergoes temporal average pooling and concatenates together, followed by the linear projection consisting of a fully connected layer and the GeLU function to project back to D-dimension channel representation. Since these steps are similar to the MHSA transformation process, by projecting the input sequence into distinct attention heads, these embeddings are designated as head tokens, representing different parts of the channel dimension. To distinguish head tokens from input tokens, we add a learnable head token embedding with the shape of $(H \times D)$ to head tokens. After obtaining head tokens, they are concatenated with the input sequence along the temporal dimension, forming a new temporal-channel token sequence with the length of $(T + H + 1)$ to the MHSA.

2.2.2. Multi-Head Self-Attention

The multi-head self-attention mechanism within our TCM operates similarly to conventional multi-head self-attention but with the input sequence containing both temporal and channel tokens, rather than just temporal tokens. To learn the temporal-

channel interaction for spoofing detection, the multi-head self-attention mechanism transforms the temporal-channel tokens into query Q, key K, and value V. This is achieved by projecting temporal-channel tokens H times using corresponding linear projection matrices W_i^Q, W_i^K, W_i^V, resulting in d-dimensional channel representations, where i represents the index of the head within the MHSA. With the scaled dot product, the self-attention operator then calculates appropriate weights for each token along the temporal axis based on its relevance to each other, and this process is repeated in parallel across H attention heads. Subsequently, the output of each head is concatenated and subjected to a final linear projection denoted as W^O, yielding the output embedding. The multi-head self-attention can be represented by the following equation:

$$\begin{aligned} \text{MultiHead}(X) &= \text{Concat}(\text{head}_1, \ldots, \text{head}_H)W^O \\ \text{where head}_i &= \text{softmax}(\frac{XW_i^Q \cdot (XW_i^K)^T}{\sqrt{d}}) \cdot XW_i^V \end{aligned} \quad (1)$$

Given that the self-attention of each head is independently computed along the temporal axis of the input sequence, if the input sequence only contains temporal tokens, the model may lack the interaction between the temporal and channel dimensions in the MHSA. However, in the proposed method, head tokens representing channel information are put together with temporal tokens, hence MHSA can learn temporal-channel dependencies by attending to different parts including temporal and head tokens of the input sequence.

2.2.3. Classification Token Enrichment

Although the classification token CLS can attend the information from both temporal and channel tokens during MHSA, we further enrich the CLS token with both temporal and channel tokens with the Classification Token Enrichment component module because the CLS token is directly used for the final prediction, and the information from both tokens can be both crucial for detecting artifacts. Figure 1.d illustrates the Classification Token Enrichment component of the proposed TCM module. Firstly, the temporal and head tokens are segregated from the MHSA output and subjected to average pooling to get the mean temporal token and mean head token. After that, instead of considering only the mean head token in the original head token design [19], our TCM module also enriches the classification token CLS with the mean temporal token. Finally, the enriched classification token is concatenated with the temporal tokens to form the output sequence, keeping the same shape of $(T + 1) \times D$ as the input sequence.

3. Experiments

3.1. Dataset and metrics

While the training and development data are from the ASVspoof 2019 [28] logical access (LA) track containing clean speech with text-to-speech and voice conversion attacks, we evaluated our method on the ASVspoof 2021 [29] logical access (LA) and deep fake (DF) tasks. ASVspoof 2021 LA eval set includes 2 known and 11 unknown and the speech data is distorted by various codec and compression variations, mimicking real-world scenarios. Additionally, ASVspoof 2021 introduced a new DF eval consisting of two new additional sets of source data compared to the LA set. Our primary evaluation metrics are the common-used equal error rate (EER) [30] and minimum normalized tandem detection cost function (t-DCF).

3.2. Implementation details

In the training step, the audio data are cropped or concatenated giving segments of approximately 4 seconds duration (64,600 samples). We used the Adam optimizer with a learning rate of 10^{-6} with a weight decay of 10^{-4} to optimize a weighted cross-entropy loss. The batch size for the training step is set to 20. The final result is reported using the model checkpoint created by averaging the top-5 best validation performance models. Early stopping is applied when the cross entropy loss in the validation set did not improve for 7 epochs. All of the experiments are trained with one Nvidia A40 GPU with the same random seed. In terms of model architecture, following our baseline [13], the pre-trained SSL model XLSR[2] is utilized as an upstream model to extract intermediate representation from the raw input signal.

To be comparable with [13, 26], the signal noise injection data augmentation technique RawBoost [31] is utilized in our experiments. The configuration and parameters of RawBoost used in our experiment are similar to the original paper. Following our baseline system [13], we trained two separate SSD systems with two different Rawboost settings to evaluate on the LA and DF track, respectively. In the LA track, the SSD system is trained with the RawBoost technique combining linear and non-linear convolutive noise and impulsive signal-dependent additive noise strategies. On the other hand, the stationary signal-independent additive, randomly colored noise, is added during the training in the DF track.

3.3. Results

3.3.1. Comparison with the state-of-the-art systems

Table 1 compares the performance of the proposed TCM with our reproduced state-of-the-art XLSR-Conformer and other existing competitive systems on the ASVspoof21 LA and DF evaluation set. In the fixed-length input evaluation on the LA track, adding the proposed TCM module can achieve 25% EER improvement than the baseline XLSR-Conformer for the pooled EER (1.03 % vs 1.40%). While XLSR-Conformer with TCM achieved comparable performance to the top-performing LA system XLSR-AASIST [26] in the LA track, it attained a new state-of-the-art result of 2.06% EER in the DF track, surpassing the previous best-reported result of XLSR-Conformer by 9.25% in the fixed-length input evaluation. Similar gains can be observed in variable-length utterance evaluation. Notably, while achieving noticeable improvement, our TCM module is lightweight since it adds only $0.03M$ parameters to the XLSR-Conformer system. In the following sections, we conduct experiments on the reproduced XLSR-Conformer for further analysis of the robustness and efficiency of TCM.

3.3.2. Transformer and Conformer comparison

To verify the robustness and effectiveness of the proposed TCM in the SSD task, we replaced the Conformer block with the Transformer one and conducted the study in Table 2. We notice that TCM can bring relatively stable improvement for both Conformer and Transformer structures. This can indicate that the learned temporal-channel dependency in TCM can be beneficial for detecting spoofed artifacts regardless of the transformer-based architectures. Furthermore, the Conformer-based system yielded superior performance compared to the corresponding Transformer in the baseline setting as well as with the TCM module. These demonstrate that the local information captured

[2]https://github.com/pytorch/fairseq/tree/main/examples/wav2vec

System	Params (M)	LA (Fix)		LA (Var)		DF (Fix)	DF (Var)
		EER (%)	min t-DCF	EER (%)	min t-DCF	EER (%)	EER (%)
RawNet2 [27]	25.43	9.50	0.4257	-	-	22.38	-
AASIST [17]	0.30	5.59	0.3398	-	-	-	-
RawFormer [11]	0.37	4.98	0.3186	4.53	0.3088	-	-
XLSR-AASIST [26]	317.84	**1.00**	**0.2120**	-	-	3.69	-
XLSR-Conformer [13]	319.74	1.38	0.2216	**0.97**	**0.2116**	<u>2.27</u>	<u>2.58</u>
XLSR-Conformer (reproduce)	319.74	1.40	0.2226	1.26	0.2200	2.79	2.98
XLSR-Conformer + TCM	319.77	<u>1.03</u>	<u>0.2130</u>	<u>1.18</u>	<u>0.2172</u>	**2.06**	**2.25**

Table 1: *Performance comparison with the state-of-the-art systems on the ASVspoof 2021 eval set with fixed-length (Fix) and variable-length (Var) utterance evaluation (Bold denotes the best result, underline denotes the second-best result, and dash denotes the results are unavailable).*

System	21LA		21DF	
	Fix	Var	Fix	Var
XLSR-Transformer	1.60	1.44	2.24	2.49
XLSR-Transformer + TCM	1.51	1.91	**2.02**	2.34
XLSR-Conformer	1.40	1.26	2.33	2.48
XLSR-Conformer + TCM	**1.03**	**1.18**	2.06	**2.25**

Table 2: *EER (%) results to evaluate the robustness of TCM for the Transformer and Conformer Block.*

Track	System	EER (%)		
		H=4	H=6	H=8
LA	XLSR-Conformer	1.40	1.14	1.72
	XLSR-Conformer + TCM	**1.03**	**1.13**	**1.06**
DF	XLSR-Conformer	2.79	2.87	**3.11**
	XLSR-Conformer + TCM	**2.06**	**2.84**	3.81

Table 3: *Our methods with different numbers of heads on ASV2021 LA & DF eval set.*

by the Convolution module in Conformer is important for the SSD task.

3.3.3. Multi-head attention

Table 3 further studies the effect of the different numbers of heads with and without TCM on the ASV2021 LA & DF eval set. The system with 4 heads can lead to the best performance. (1.03 % EER on LA track and 2.06 % EER on DF track.). The improvement by TCM is robust for most cases, except the DF eval track with 8 heads. It is important that an increase in the number of attention heads does not necessarily ensure improved results.

3.3.4. Ablation study

Table 4 presents an analysis of the contributions of each component within our TCM module on the ASV2021 DF evaluation set. We observed that the inclusion of head token embeddings leads to a slight improvement in system performance. Conversely, the performance of the TCM module experiences a notable decline, from 2.06% to 2.40% EER, when head tokens are excluded from the multi-head attention mechanism or when the mean head token (mean HT) is omitted from addition to the CLS token. A similar trend is observed when the mean temporal token (mean TT) is excluded from enriching the CLS token. Notably, when both mean HT and mean TT are absent, the performance drops significantly to 3.25% EER, indicating a deterioration compared to the baseline system. These findings underscore the importance of leveraging both temporal and channel information, as represented by temporal tokens and head tokens, in the task of detecting synthetic speech.

	EER (%)
XLSR-Conformer w/o TCM (baseline)	2.79
XLSR-Conformer + TCM	**2.06**
w/o HT embedding	2.08
w/o HT in MHSA	2.40
w/o adding mean HT to CLS	2.41
w/o adding mean TT to CLS	2.33
w/o adding mean HT & Mean TT to CLS	3.25

Table 4: *Ablation study of each component in our proposed TCM on ASV2021 DF evaluation set. HT and TT represent head tokens and temporal tokens, respectively.*

4. Conclusion

In this paper, we propose a Temporal-Channel Modeling module for MHSA-based synthetic speech detection systems. Our method integrates the channel representation head token into the temporal input token within the multi-head self-attention, which forces the model to learn the temporal-channel dependencies from the input sequence. The XLSR-Conformer using our TCM module outperforms the state-of-the-art performance and outperforms competing methods on the ASVspoof 2021 eval set. Additionally, the ablation study validates the effectiveness of our proposed method and demonstrates the importance of temporal-channel modeling in synthetic speech detection.

5. Acknowledgement

This research is supported by the National Research Foundation Singapore under the AI Singapore Programme (AISG Award No.: AISG-TC-2023-011-SGIL). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore. The computational work for this article was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg).

6. References

[1] K. C. Yuen, L. Haoyang, and C. E. Siong, "Asr model adaptation for rare words using synthetic data generated by multiple text-to-speech systems," in *2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2023, pp. 1771–1778.

[2] H. Wu, J. Kang, L. Meng, H. Meng, and H. yi Lee, "The defender's perspective on automatic speaker verification: An overview," 2023.

[3] A. Khan, K. M. Malik, J. Ryan, and M. Saravanan, "Voice spoofing countermeasures: Taxonomy, state-of-the-art, experimental analysis of generalizability, open challenges, and the way forward," *arXiv preprint arXiv:2210.00417*, 2022.

[4] Z. Wu, R. K. Das, J. Yang, and H. Li, "Light convolutional neural network with feature genuinization for detection of synthetic speech attacks," in *Proc. INTERSPEECH*, 2020.

[5] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "Stc antispoofing systems for the asvspoof2019 challenge," in *Proc. INTERSPEECH*, 2019.

[6] X. Li, N. Li, C. Weng, X. Liu, D. Su, D. Yu, and H. M. Meng, "Replay and synthetic speech detection with res2net architecture," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6354–6358, 2021.

[7] X. Li, X. Wu, H. Lu, X. Liu, and H. Meng, "Channel-wise gated res2net: Towards robust detection of synthetic speech attacks," *Proc. INTERSPEECH*, 2021.

[8] N. Müller, P. Czempin, F. Diekmann, A. Froghyar, and K. Böttinger, "Does Audio Deepfake Detection Generalize?" in *Proc. INTERSPEECH*, 2022, pp. 2783–2787.

[9] A. M. Rostami, M. M. Homayounpour, and A. Nickabadi, "Efficient attention branch network with combined loss function for automatic speaker verification spoof detection," *Circuits, Systems, and Signal Processing*, pp. 1 – 19, 2021.

[10] C. Li, F. Yang, and J. Yang, "The role of long-term dependency in synthetic speech detection," *IEEE Signal Processing Letters*, vol. 29, pp. 1142–1146, 2022.

[11] X. Liu, M. Liu, L. Wang, K. A. Lee, H. Zhang, and J. Dang, "Leveraging positional-related local-global dependency for synthetic speech detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[12] H. seo Shin, J. Heo, J. ho Kim, C. yeong Lim, W. Kim, and H.-J. Yu, "Hm-conformer: A conformer-based audio deepfake detection system with hierarchical pooling and multi-level classification token aggregation methods," 2023.

[13] E. Rosello, A. Gomez-Alanis, A. M. Gomez, and A. Peinado, "A conformer-based classifier for variable-length utterance processing in anti-spoofing," in *Proc. INTERSPEECH*, 2023, pp. 5281–5285.

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

[15] K. Sriskandaraja, V. Sethu, P. N. Le, and E. Ambikairajah, "Investigation of sub-band discriminative information between spoofed and genuine speech," in *Proc. INTERSPEECH*, 2016.

[16] H. Tak, J. Patino, A. Nautsch, N. W. D. Evans, and M. Todisco, "An explainability study of the constant q cepstral coefficient spoofing countermeasure for automatic speaker verification," in *The Speaker and Language Recognition Workshop*, 2020.

[17] J. weon Jung, H.-S. Heo, H. Tak, H. jin Shim, J. S. Chung, B.-J. Lee, H. jin Yu, and N. W. D. Evans, "Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6367–6371, 2022.

[18] F. Chen, S. Deng, T. Zheng, Y. He, and J. Han, "Graph-based spectro-temporal dependency modeling for anti-spoofing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[19] Z. Lu, H. Xie, C. Liu, and Y. Zhang, "Bridging the gap between vision transformers and convolutional neural networks on small datasets," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022.

[20] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. M. Pino, A. Baevski, A. Conneau, and M. Auli, "XLS-R: Self-supervised cross-lingual speech representation learning at scale," in *Interspeech*, 2021.

[21] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-task self-supervised learning for robust speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6989–6993.

[22] D.-T. Truong, R. Tao, J. Q. Yip, K. A. Lee, and E. S. Chng, "Emphasized non-target speaker knowledge in knowledge distillation for automatic speaker verification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 10 336–10 340.

[23] E. da Silva Morais, R. Hoory, W. Zhu, I. Gat, M. Damasceno, and H. Aronowitz, "Speech emotion recognition using self-supervised features," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6922–6926.

[24] D.-T. Truong, T. T. Anh, and C. E. Siong, "Exploring speaker age estimation on different self-supervised learning models," in *IEEE APSIPA ASC*, 2022, pp. 1950–1955.

[25] T. Gupta, D.-T. Truong, T. T. Anh, and C. E. Siong, "Estimation of speaker age and height from speech signal using bi-encoder transformer mixture model," in *Proc. INTERSPEECH*, 2022, pp. 1978–1982.

[26] H. Tak, M. Todisco, X. Wang, J.-w. Jung, J. Yamagishi, and N. Evans, "Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation," in *ODYSSEY 2022, The Speaker Language Recognition Workshop*, 2022.

[27] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. W. D. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6369–6373, 2021.

[28] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. W. D. Evans, T. H. Kinnunen, and K. A. LEE, "Asvspoof 2019: Future horizons in spoofed and fake audio detection," in *Proc. INTERSPEECH*, 2019.

[29] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgado, T. Kinnunen, M. Todisco, J. Yamagishi, N. Evans, A. Nautsch, and K. A. Lee, "Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, p. 2507–2522, 2023.

[30] N. Brümmer and E. de Villiers, "The bosaris toolkit: Theory, algorithms and code for surviving the new dcf," 2013.

[31] 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 International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.