CAU KU deep fake detection system for ADD 2023 challenge*

Soyul Han[1,†], Taein Kang[1,†], Sunmook Choi[2,†], Jaejin Seo[1], Sanghyeok Chung[2], Sumi Lee[1], Seungsang Oh[2,*] and Il-Youp Kwak[1,*]

[1]Chung-Ang University, 84, Heukseok-ro, Dongjak-gu, Seoul 06974, Republic of Korea
[2]Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 02841, Republic of Korea

Abstract
The paper presents the participation of the CAU_KU team in the ADD 2023 Challenge, specifically in track 1.2 (audio fake game - detection track) and track 3 (deepfake algorithm recognition track). Various deep learning models were explored using features from the pretrained wav2vec2 network, as well as CQT, mel-spectrogram, etc. We modified the representation extraction component of the AASIST model to incorporate 2D spectrograms (wav2vec2 or CQT) and attempted different deep learning models, with model ensembling employed to create the final model. For track 1.2, our submitted ensemble model for round 1 utilized the CQT-LCNN and CQT-AASIST models. For round 2, our model used the CQT-LCNN, CQT-AASIST, and W2V2-GMM models. For track 3, we ensembled the CQT-LCNN, CQT-OFD and AASIST models. Additionally, we applied the openmax algorithm to detect unknown deepfake attacks. Our best submission achieved 23.44% and 21.26% on round 1 and 2 of track 1.2, respectively, and ranked 3rd in track 1.2.

Keywords
audio deep synthesis, audio deepfake detection, deep learning, deepfake algorithm recognition

1. Introduction

This paper describes the model developed by the Chung-Ang University and Korea University (CAU_KU) team for the second Audio Deep synthesis Detection Challenge (ADD 2023) [1] in two tracks: Track 1.2 (audio fake game - detection task) and Track 3 (deepfake algorithm recognition). Our team competed in Track 1.2, which focuses on detecting fake audio in a given dataset. Participants in this track were asked to detect counterfeit audio, especially the fake samples generated from Track 1.1, in the evaluation dataset provided by the organizers. The task was evaluated by two rounds, with the second round involving a released deepfake detection model that contestants were tasked with deceiving. Track 3, which is known as Deepfake Algorithm Recognition (AR), aimed to recognize the algorithms used to generate deepfake audio utterances. The dataset provided both known and unknown algorithms associated with the fake audio. Participants were tasked with developing algorithms capable of accurately classifying the deepfake utterances into their respective categories.

In this study, we aimed to develop an advanced model by leveraging the strengths of current state-of-the-art methods, such as the Vicomtech ADD system that achieved the first-place in the previous ADD 2022 challenge [2], and the AASIST model [3] that achieved the state-of-the-art performance on the ASVspoof 2019 LA data. To incorporate the effectiveness of the Wav2Vec2 (W2V2) feature utilized in the Vicomtech ADD system, we integrated it into the representation extraction part for the AASIST model. This approach proved successful on the ASVspoof 2019 LA data, where the W2V2-AASIST model achieved a state-of-the-art performance of 0.21% EER. However, when evaluated on the ADD 2023 data, this model did not yield satisfactory results, with a 40% test EER. Thus, we considered ensemble modeling by combining multiple well-performing models for the ADD 2023 challenge.

- Modified representation extraction part of the AASIST model utilizing W2V2 and CQT.
- Experimented with models that ranked 3rd in the previous ADD 2022 challenge [4] such as LCNN, ResMax, and OFD.
- Conducted experiments using the Gaussian mixture model (GMM) with the W2V2 feature, as well as traditional features such as MFCC and CQT.

IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis (DADA 2023), August 19, 2023, Macao, S.A.R
*You can use this document as the template for preparing your publication. We recommend using the latest version of the ceurart style.
*Co-corresponding author.
†These authors contributed equally.
soyul5458@cau.ac.kr (S. Han); xodls4179@cau.ac.kr (T. Kang); felixchoi@korea.ac.kr (S. Choi); seojaejin@cau.ac.kr (J. Seo); cshzton@korea.ac.kr (S. Chung); dltnal821@cau.ac.kr (S. Lee); seungsang@korea.ac.kr (S. Oh); ikwak2@cau.ac.kr (I. Kwak)
https://ikwak2.github.io/ (I. Kwak)
0000-0003-0156-250X (S. Han); 0009-0007-4978-9101 (T. Kang); 0009-0006-3004-9222 (S. Choi); 0009-0005-3690-8680 (J. Seo); 0009-0002-2257-4729 (S. Chung); 0009-0009-8867-968X (S. Lee); 0000-0003-4975-9977 (S. Oh); 0000-0002-7117-7669 (I. Kwak)

- Applied OpenMax algorithm for track 3

For the first round of Track 1.2, the ensemble model submitted by our team comprised the CQT-LCNN and CQT-AASIST models. For the second round, the ensemble model consisted of the CQT-LCNN, CQT-AASIST, and W2V2-GMM models. These submissions achieved EERs of 23.44% and 21.26% on round 1 and 2 of Track 1.2, respectively, and ranked 3rd in this track.

In Track 3, we considered an ensemble of three models: CQT-LCNN, CQT-OFD, and AASIST. To detect new attack types, the OpenMax algorithm was applied. Our system achieved an F1-score of 0.7205 for Track 3.

2. Methods

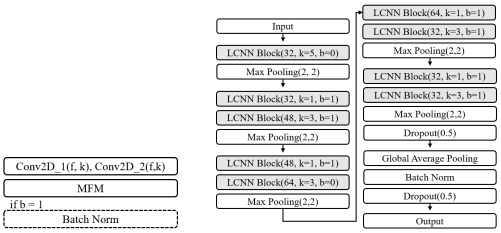
2.1. Feature engineering

In this study, we conducted experiments utilizing four widely used audio feature extraction methods: CQT, Mel-spectrogram, MFCC, and W2V2 [5]. Each method possesses distict advantages and limitations, rendering them suitable for specific applications. CQT uses a constant Q factor to ensure higher frequency resolution at low frequencies and lower resolution at high frequencies, and has demonstrated effectiveness in deepfake detection tasks. Mel-spectrogram is obtained by applying Mel-filterbanks to the power spectrum of the audio signal. MFCC is another popular feature extraction method used in speech processing and music analysis. W2V2 is a state-of-the-art speech recognition method that learns powerful representations from speech audio alone and achieves impressive results with significantly less labeled data compared to previous methods. The first-placed team in the ADD 2022 challenge at track 1 (deepfake detection track) demonstrated the usefulness of the W2V2 pretrained network [2]. By applying the discrete cosine transform (DCT) to the CQT or mel-spectrogram features, we obtain more compressed representations: Constant Q Cepstral Coefficients (CQCC) or MFCC. In deep learning scenarios, raw data such as mel-spectrogram and CQT often lead to higher accuracy. Thus, for our deep-learning models, we opted for mel-spectrogram and CQT features rather than CQCC and MFCC features.

2.2. Data augmentation

We explored several augmentation techniques such as mixup [6], SpecAugment [7], FFM [4], FilterAugment [8] and cutout [9]. These techniques have previously shown promise in improving performance in the ADD 2022 challenge [10]. However, in the context of the ADD 2023 challenge, incorporating these augmentation techniques did not yield substantial improvements in performance.

2.3. Models



(a) LCNN block (b) LCNN model

Figure 1: LCNN Model Architectures

2.3.1. LCNN model

The efficacy of the LCNN model has been demonstrated in previous research through its notable performance in the ASVspoof 2017, 2019, and 2021 challenges [11, 12, 13]. Our implementation of the LCNN model as depicted in Figure 1(b) [14], consists of 9 layers, akin to the Light CNN-9 model. However, we made modifications to the architecture by substituting the fully connected layer defined in the original Light CNN-9 model [14] with a global average pooling layer, batch normalization, and dropout layer. In Track 1.2, the final dense layer of our LCNN model outputs two values, representing the labels "spoofing" and "genuine." In Track 3, the output dense layer had a size of 7, representing the seven known deepfake algorithms, and it was activated using the softmax activation function. Figure 1(a) describes the LCNN block, where f denotes the filter size, k denotes the kernel size, and b indicates the use of batch normalization. The LCNN block performs MFM (Max-Feature-Map) operation using two convolution layers and optionally applies a batch normalization layer indicated by the dashed block when $b = 1$.

2.3.2. AASIST model and our proposed AASIST variant

AASIST is an extended version of the RawGAT-ST[15] that is based on a graph neural network [16]. AASIST has achieved state-of-the-art performance on ASVSpoof 2019 challenge dataset for logical access (LA) scenario.

We propose modifications to the representation extraction part of the AASIST model. We conducted experiments by replacing this extraction part by either a W2V2 pretrained model or CQT features, as shown in Figure 2. In the figure, the upper component of the representation extraction part depicts the original AASIST model. The middle component represents the model utilizing W2V2, with fine-tuning of the last transformer layers

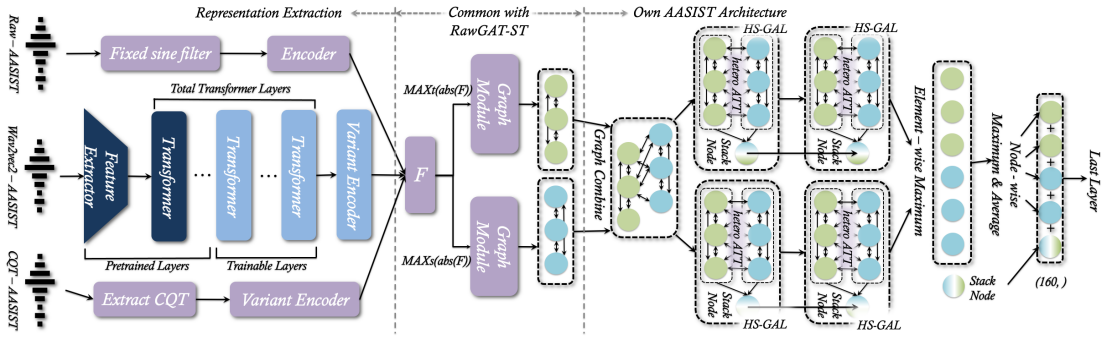


Figure 2: Our proposed AASIST variants, which are modified models in the representation extraction part, utilizing either the W2V2 pretrained model or CQT features.

in the W2V2 pretrained model. The lower component represents the model considering CQT features.

2.3.3. GMM model

The Gaussian Mixture Model (GMM) is a probabilistic model that represents data as a combination of multiple Gaussian distributions [17]. During the ADD 2023 challenge, it was observed that the performance of deep learning models on the test set did not meet the anticipated level of success. This led us to consider using the traditional machine learning-based GMM model, which has been widely employed in ASVspoof 2015 [18] and ASVspoof 2017 [19]. In addition, considering the necessity for simpler models to prevent overfitting, we recognized GMM as a suitable method to effectively model features extracted through W2V2 pretrained networks in a straightforward yet effective manner. We considered using various features such as MFCC, CQT, and W2V2 as input features for the GMM model.

2.3.4. OFD model

The Overlapped frequency-distributed (OFD) network [20] is a spoofing detection model designed to detect distinct features within different frequency ranges by dividing spectrograms along the frequency axis. There are two types of models: OFD model and Non-OFD model.

In OFD model, each block divides the feature map into multiple parts along the frequency axis allowing for overlap. In contrast, Non-OFD model partitions the feature map along the frequency axis without any overlap. Both models consist of six blocks, each characterized by three hyperparameters: the number of splits, the presence or absence of overlap, and the activation function. In OFD model, all six blocks are split with overlap, while in Non-OFD model, no overlaps occur between blocks. The activation function can be either ReLU or MFM. For instance, "OFD with $(s_1, s_2, s_3, s_4, s_5, s_6)$-ReLU" refers to OFD model with s_i number of splits in the ith block, using ReLU. If $s_i = 0$, then it implies that the ith block does not split the input feature map.

2.3.5. Other models

Additionally, we conducted experiments with several alternative methods, including ResMax [21, 22], BC-ResMax, and DDWS [23]. However, these methods exhibited comparable accuracy to the LCNN model, and due to limited time, we were unable to dedicate further investigation to these models.

2.4. OpenMax for unknown attack detection

OpenMax [24] is an algorithm designed for open set recognition, specifically targeting the identification of utterances belonging to the unknown class. The algorithm consists of two steps: preparation and inference.

During the preparation step, a model is trained using known classes from the training set. Following the training phase, final-layer logit vectors (seven-dimensional) are computed for correctly classified training data samples. The mean vector μ_j of the logit vectors corresponding to each class $j = 0, 1, \ldots, 6$ is computed. The distance between the logit vector of each correctly classified training sample and the mean vector of its class is determined. Weibull distributions are fitted using the libMR [25] FitHigh function for each class, using the η number of samples with the largest distance to the mean vector.

In the inference step, the final-layer logit vectors are obtained for all test samples. For each logit vector $v_i = (v_{i,0}, \ldots, v_{i,6})$, the probability w_j of not belonging to class j is calculated for all $j = 0, 1, \ldots, 6$. The logit vector is then updated as

$$\tilde{v}_i = \left((1 - w_0)v_{i,0}, \ldots, (1 - w_6)v_{i,6}, \sum_{j=0}^{6} w_j v_{i,j} \right),$$

and the softmax of $\tilde{v}$ serves as the output of the OpenMax algorithm.

To handle uncertain predictions a threshold θ is set. For each i, if $\max_{j\in\{0,\ldots,7\}} softmax(\tilde{v}_i)_j \leq \theta$ or the unknown class (j = 7) has the largest probability, then its predicted class is considered to be 7.

3. Experiments

3.1. Datasets

3.1.1. ADD 2023 challenge datasets

The ADD 2023 challenge consists of three tracks, and we describe the datasets for Track 1.2 and Track 3 [1]. Track 1.2 aims to detect fake audio, which refers to realistic and natural-sounding fake voice audio that can deceive deepfake detection models. This track is divided into two rounds, both featuring nearly identical detection tasks. Table 1 shows the number of samples in the training, development, and test sets (round 1 and 2). Track 3 aims to recognize deepfake speech algorithms. The training and development sets have seven categories (0, 1, 2, ..., 6) with labels, one of which is real and the other six are fake speech algorithms. Notably, the label for real speech is unknown. The test set has eight categories, but no label information is provided. Seven of them align with the "known" classes in the training and development sets, while the remaining category represents the unknown fake class labeled as 7. Table 2 shows the number of samples in the training, development, and test sets.

Table 1
The number of samples in the training, development, test sets (round 1 and 2) of the ADD 2023 track 1.2 dataset.

	Training	Dev.	R1. Test	R2. Test
Genuine	3,012	2,307	Unknown	Unknown
Fake	24,072	26,107	Unknown	Unknown
Total	27,084	28,324	111,976	118,477

3.1.2. ASVspoof 2019 challenge LA dataset

The ASVspoof 2019 challenge [26] focuses on TTS, VC, and replay spoofing attacks, and the dataset consists of logical access (LA) and physical access (PA) scenarios derived from the VCTK basic corpus [27]. Our focus primarily lies on the LA data, which uses 17 TTS and VC systems to produce both genuine and fake speech samples. The dataset is partitioned into three subsets: training, development, and evaluation. Here, the evaluation data contains approximately 71K utterances with unknown attacks.

Table 2
The number of samples in the training, development, and test sets of the ADD 2023 track 3 dataset.

Labels	Training	Dev.	Test
0	3,200	1,200	Unknown
1	3,200	1,200	Unknown
2	3,200	1,200	Unknown
3	3,200	1,200	Unknown
4	3,200	1,200	Unknown
5	3,200	1,200	Unknown
6	3,200	1,200	Unknown
7	0	0	Unknown
Total	22,400	8,400	79,490

3.2. Experimental setup

To assess the performance of our experimental models, we conducted evaluation on two databases: the ADD 2023 challenge dataset and the ASVspoof 2019 LA dataset. The model's performance was evaluated using the equal error rate (EER), which indicates the point at which the false acceptance rate (FAR) and false rejection rate (FRR) are equal. A lower EER value generally indicates better performance.

The CQT-LCNN model was trained using 9-second samples, a batch size of 16, and 10 epochs. To fit the 9-second signal, audio signals longer than 9 seconds were trimmed, and signals shorter than 9 seconds were repeated from the beginning to match the desired length. In the case of training the GMM model, the entire length of audio signals was used for extracting MFCC and CQT features, while 13.67 seconds of audio signals were used for W2V2 feature extraction to match the fixed input length of the pretrained network, which is set at 246,000. To simplify the structure of the GMM model, we assumed a diagonal covariance matrix.

In order to stabilize the convergence of model parameters, the learning rate is initially set to 1e-3 and subsequently reduced to 1e-5 using a sigmoidal decay function. For the ASVspoof 2019 dataset, we trained the models using only the training data. However, for the models submitted in the challenge, we trained using both the training and development sets for some sub-models.

3.3. Experimental results on ADD 2023 dataset for track 1.2

Many of the models exhibited favorable performance on the training and development data. However notable declines in performance were observed when evaluating the models on the actual test data. This indicates that the models suffer from overfiting for both training and development data. To address this issue, techniques such as data augmentation and reducing model size can be

considered.

3.3.1. Use of data augmentation techniques

We utilized various data augmentation techniques such as mixup, FFM (LF, HF and RF), FilterAugment (FA) [8] and cutout. However, the application of data augmentation did not yield substantial improvement when evaluated on the test data. Table 3 shows the results of applying data augmentation to the CQT-LCNN and BC-ResMax models. It was difficult to draw conclusions about the effectiveness of data augmentation based on the experimental results.

Table 3
Model performance comparison according to data augmentation.

Model	Data aug.	Dev. EER	Test EER (R1)
CQT-LCNN	None	0.14%	29.75%
CQT-LCNN	Mixup	0.18%	37.12%
BC-ResMax	None	0.12%	35.86%
BC-ResMax	Mixup	0.16%	44.63%
BC-ResMax	Mixup, FA	2.51%	42.07%
BC-ResMax	FFM (LF)	0.10%	34.00%
BC-ResMax	FFM (RF)	0.09%	39.01%
BC-ResMax	FFM (HF)	0.09%	35.44%

3.3.2. GMM based models

Deep learning-based models have been observed to suffer from serious overfitting issues in terms of test set accuracy. In order to address this concern, we experimented with GMM-based models, which had demonstrated success in prior ASVspoof challenges (2015 and 2017), and were known for their ability to handle overfitting. We created a deepfake detection model using GMM models with W2V2, CQT, and MFCC features. For W2V2 features, we experimented with two W2V2 pretrained models: one trained on the Librispeech corpus's 960 hours of audio (LS-960) and the other trained on the LibriVox 60k hours of data (LV-60k). We varied the number of components parameter for the W2V2-LV60k-GMM, exploring values of 16, 32, 64, and 128. Table 4 presents the experimental results. W2V2-LV60k-GMM demonstrated better performance than W2V2-LS960-GMM based on test EER and Dev EER. Although W2V2-LV60k-GMM models exhibited higher Dev EER compared to other deep learning-based methods, it yielded better results in terms of test EER. The simpler structure of the GMM-based model appeared to mitigate the overfitting issue to some extent. Additionally, we conducted experiments with CQT-GMM and MFCC-GMM models, but W2V2-LV60k-GMM exhibited the best performance.

Table 4
Model performance comparison for GMM-based models.

Features	N. comp.	Dev. EER	Test EER (R2)
W2V2-LV60k	16	3.81%	30.55%
W2V2-LV60k	32	3.18%	27.40%
W2V2-LV60k	64	2.58%	25.91%
W2V2-LV60k	128	2.56%	24.61%
W2V2-LS960	64	14.55%	39.23%
CQT	128	15.32%	43.02%
MFCC	16	3.28%	35.32%

3.3.3. AASIST based models

As in RawNet2, the Raw-AASIST model uses Fixed Sinc Filters to extract features from raw audio and compares them across different epochs. On the other hand, the W2V2-AASIST and CQT-AASIST models substitute Fixed Sinc Filters with W2V2 and CQT, respectively. We used the XLS-R (1B) version as the pretrained model for W2V2-AASIST [28]. Table 5 presents the performance for three AASIST-based models. It was observed that training the models for multiple epochs led to overfitting on the test data, resulting in a decrease in the dev EER but an increase in the test EER. The CQT-AASIST (10ep) model refers to the model trained for 10 epochs. Although models trained for more epochs exhibited lower dev EER, overfitting was evident in the test EER. Therefore, despite having a higher dev EER, we chose to use models trained with a small number of epochs, specifically between 5 and 10 for the AASIST based models.

Table 5
Model performance comparison for AASIST-based models.

Models	Dev. EER	Test EER (R1)	Test EER (R2)
Raw-AASIST	0.79%	37.36%	-
W2V2-AASIST	0.12%	39.83%	-
CQT-AASIST (10ep)	4.97%	30.54%	29.72%
CQT-AASIST (20ep)	1.18%	-	31.22%

3.4. Experimental results on ASVspoof 2019 LA dataset

Table 6 presents the performance of the models (developed for the ADD 2023 challenge) on the ASVspoof 2019 LA dataset (ASV2019). The EER (ASV) indicates the performance evaluated on the evaluation data after training on the training data of ASV2019. The EER (ADD-R1) and EER (ADD-R2) columns indicate the performance on the test data of round 1 and round 2 of the ADD 2023 challenge, respectively. Among our experimental models, the W2V2-GMM model showed the best performance on ADD-R2 with a 26.28% EER. However, it exhibited poor

performance on ASV2019, achieving a 9.8% EER. The CQT-LCNN and CQT-AASIST models, which performed well on ADD-R1 and ADD-R2 achieved EERs of 1.93% and 2.36%, respectively, on ASV2019. The W2V2-AASIST model showed exceptional performance with a 0.21% EER on ASV2019, but performed poorly on ADD-R1. Regarding the MFCC-LCNN model, it demonstrated good performance on ADD-R2, but showed poor performance on ASV2019.

Table 6
Model performance comparison on ASVspoof 2019 LA dataset and ADD 2023 test data (round 1 and round 2).

Models	EER (ASV)	EER (ADD-R1)	EER (ADD-R2)
CQT-LCNN	1.93%	29.75%	35.4%
W2V2-GMM	9.8%	-	26.28%
AASIST [3]	0.83%	37.36%	-
CQT-AASIST	2.36%	30.54%	29.72%
W2V2-AASIST	0.21%	39.83%	-
CQT-OFD	1.82%	35.84%	46.3%
MFCC-LCNN	10.05%	-	29.00%

3.5. Submitted ensemble system for track 1.2

Table 7 describes the details of the three top-performing single systems, including their EERs on the final evaluation data (R1 and R2) in track 1.2 of the ADD 2023 challenge as well as the EERs of our two ensemble systems. In Round 1, our final model consisted of an ensemble of CQT-LCNN and CQT-AASIST models in a 1:1 ratio, achieving 23.44% EER. In Round 2, we submitted an ensemble of CQT-LCNN, CQT-AASIST, and W2V2-GMM models, considering their respective accuracies, achieving 21.26% EER.

Table 7
EER on the final evaluation data for track 1.2.

Model	Feature	EER (R1)	EER (R2)
LCNN	CQT	29.75%	35.40%
AASIST	CQT	30.54%	29.72%
GMM	W2V2-LV60k	-	26.28%
Ensemble 1	-	23.44%	-
Ensemble 2	-	-	21.26%

3.6. Submitted ensemble system for track 3

Table 8 describes the three single models used in track 3. After training all the models, the OpenMax algorithm, with $\eta = 20$ and $\theta = 0.25$, is applied to the sum of their final logit vectors from the train and development sets, with the ratios specified in Table 8. The models were slightly modified to adapt them from spoofing detection to algorithm recognition tasks. The CQT-LCNN model remains unchanged, with an output dense layer of size 7 with softmax activation. For the OFD model, (2,2,0,0,0,0)-ReLU configuration is used, and two additional dense layers with 128 and 64 nodes are added just before the final layer to use the features from CNN backbone for classifying algorithms. Lastly, the AASIST model [3] was used with modified output dense layer of size 7 with softmax activation. The ensemble of these three models achieved a 0.7205 test F1-Score.

Table 8
F1-score on the final evaluation data for track 3.

Model	Feature	Ratio	Test F1-Score
LCNN	CQT	1.5	0.7005
OFD	CQT	1.2	0.6947
AASIST	Raw Audio	1	0.6745
Ensemble	-	-	0.7205

4. Conclusion

This paper presents the models employed by our CAU_KU team participating in Track 1.2 and Track 3 of the ADD 2023 challenge. We utilized various deepfake models, including the W2V2 pretrained model and a modified AASIST architecture. In Track 1.2, Round 1, our submission consisted of an ensemble model comprising the CQT-LCNN and CQT-AASIST models, achieving a 23.44% EER. In Round 2, our submission involved an ensemble model combining the CQT-LCNN, CQT-AASIST, and W2V2-GMM models, achieving a 21.26% EER. For Track 3, we developed an ensemble model using the CQT-LCNN, CQT-OFD, and AASIST models, achieving a 0.7205 F1-score.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science and ICT (RS-2023-00208284, 2020R1C1C1A01013020) and Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-00033, 50%, Study on Quantum Security Evaluation of Cryptography based on Computational Quantum Complexity).

References

[1] J. Yi, J. Tao, R. Fu, X. Yan, C. Wang, T. Wang, C. Y. Zhang, X. Zhang, Y. Zhao, Y. Ren, L. Xu, J. Zhou, H. Gu, Z. Wen, S. Liang, Z. Lian, H. Li, Add 2023: the second audio deepfake detection challenge, in: IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis (DADA 2023), volume 0, 2023, pp. 0–0.

[2] J. M. Martín-Doñas, A. Álvarez, The vicomtech audio deepfake detection system based on wav2vec2 for the 2022 add challenge, in: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 9241–9245. doi:10.1109/ICASSP43922.2022.9747768.

[3] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, N. Evans, 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), IEEE, Brno, 2022, pp. 6367–6371. doi:10.1109/ICASSP43922.2022.9747766.

[4] I.-Y. Kwak, S. Choi, J. Yang, Y. Lee, S. Han, S. Oh, Low-quality fake audio detection through frequency feature masking, in: Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia, DDAM '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 9–17. URL: https://doi.org/10.1145/3552466.3556533. doi:10.1145/3552466.3556533.

[5] A. Baevski, Y. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 12449–12460. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/92d1e1eb1cd6f9fba3227870bb6d7f07-Paper.pdf.

[6] H. Zhang, M. Cisse, Y. N. Dauphin, D. Lopez-Paz, mixup: Beyond empirical risk minimization, 2017.

[7] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, Q. V. Le, Specaugment: A simple augmentation method for automatic speech recognition, in: Proc. Interspeech 2019, ISCA, Graz, 2019, pp. 2613–2617.

[8] H. Nam, S.-H. Kim, Y.-H. Park, Filteraugment: An acoustic environmental data augmentation method, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 4308–4312.

[9] T. DeVries, G. W. Taylor, Improved regularization of convolutional neural networks with cutout, 2017.

[10] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, C. Wang, T. Wang, Z. Tian, Y. Bai, C. Fan, S. Liang, S. Wang, S. Zhang, X. Yan, L. Xu, Z. Wen, H. Li, Add 2022: the first audio deep synthesis detection challenge, in: 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, IEEE, Singapore, 2022, pp. 9216–9220.

[11] G. Lavrentyeva, S. Novoselov, E. Malykh, A. Kozlov, O. Kudashev, V. Shchemelinin, Audio replay attack detection with deep learning frameworks, in: Proc. Interspeech 2017, ISCA, Stockholm, 2017, pp. 82–86.

[12] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, A. Kozlov, STC Antispoofing Systems for the ASVspoof2019 Challenge, in: Proc. Interspeech 2019, ISCA, Graz, 2019, pp. 1033–1037. URL: http://dx.doi.org/10.21437/Interspeech.2019-1768. doi:10.21437/Interspeech.2019-1768.

[13] A. Tomilov, A. Svishchev, M. Volkova, A. Chirkovskiy, A. Kondratev, G. Lavrentyeva, STC Antispoofing Systems for the ASVspoof2021 Challenge, in: Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge, ISCA, Brno, 2021, pp. 61–67. doi:10.21437/ASVSPOOF.2021-10.

[14] X. Wu, R. He, Z. Sun, T. Tan, A light cnn for deep face representation with noisy labels, IEEE Transactions on Information Forensics and Security 13 (2018) 2884–2896. doi:10.1109/TIFS.2018.2833032.

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

[16] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, arXiv preprint arXiv:1710.10903 (2017).

[17] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag, Berlin, Heidelberg, 2006.

[18] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilci, M. Sahidullah, A. Sizov, Asvspoof 2015: The first automatic speaker verification spoofing and countermeasures challenge, in: Proc. Interspeech 2015, ISCA, Dresden, 2015, pp. 2037–2041.

[19] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, K. A. Lee, The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection, in: Proc. Interspeech 2017, ISCA, Stockholm, 2017, pp. 2–6.

[20] S. Choi, I.-Y. Kwak, S. Oh, Overlapped frequency-distributed network: Frequency-aware voice spoofing countermeasure, in: Proc. Interspeech 2022, ISCA, Incheon, 2022, pp. 3558–3562.

[21] I.-Y. Kwak, S. Kwag, J. Lee, J. H. Huh, C.-H. Lee,

Y. Jeon, J. Hwang, J. W. Yoon, ResMax: Detecting Voice Spoofing Attacks with Residual Network and Max Feature Map, in: 25th International Conference on Pattern Recognition (ICPR), IEEE Computer Society, Milan, 2021, pp. 4837–4844.

[22] I.-Y. Kwak, S. Kwag, J. Lee, Y. Jeon, J. Hwang, H.-J. Choi, J.-H. Yang, S.-Y. Han, J. H. Huh, C.-H. Lee, J. W. Yoon, Voice spoofing detection through residual network, max feature map, and depthwise separable convolution, IEEE Access (2023) 1–1. doi:10.1109/ACCESS.2023.3275790.

[23] S. Choi, S. Oh, J. Yang, Y. Lee, I.-Y. Kwak, Lightweight frequency information aware neural network architecture for voice spoofing detection, in: 26th International Conference on Pattern Recognition (ICPR), IEEE Computer Society, Montreal Quebec, 2022, pp. 477–483.

[24] A. Bendale, T. E. Boult, Towards open set deep networks, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015) 1563–1572.

[25] W. J. Scheirer, A. Rocha, R. J. Micheals, T. E. Boult, Meta-recognition: The theory and practice of recognition score analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (2011) 1689–1695. doi:10.1109/TPAMI.2011.54.

[26] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. H. Kinnunen, K. A. Lee, ASVspoof 2019: Future Horizons in Spoofed and Fake Audio Detection, in: Proc. Interspeech 2019, ISCA, Graz, 2019, pp. 1008–1012. URL: http://dx.doi.org/10.21437/Interspeech.2019-2249. doi:10.21437/Interspeech.2019-2249.

[27] C. Veaux, J. Yamagishi, K. MacDonald, et al., Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit, 2017. URL: https://datashare.ed.ac.uk/handle/10283/2651.

[28] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, et al., Xls-r: Self-supervised cross-lingual speech representation learning at scale, arXiv preprint arXiv:2111.09296 (2021).