



Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestch

Full length article

Enhancing voice spoofing detection in noisy environments using frequency feature masking augmentation

Soyul Han ^{a,b}, Jaejin Seo ^b, Sunmook Choi ^{c,d}, Taein Kang ^b, Sanghyeok Chung ^c, Seungeun Lee ^c, Seoyoung Park ^b, Seungsang Oh ^{c,*}, Il-Youp Kwak ^{b,*}^a Institute for Data Innovation in Science, Seoul National University, Seoul 08826, South Korea^b Department of Statistics and Data Science, Chung-Ang University, Seoul 06974, South Korea^c Department of Mathematics, Korea University, Seoul 02841, South Korea^d Center for Applied Mathematics, Cornell University, Ithaca, NY 14850, USA

ARTICLE INFO

Keywords:

Voice spoofing detection

Fake audio detection

Data augmentation

Deep learning

ABSTRACT

In the rapidly evolving landscape of voice-related technology, high-tech companies are developing multifaceted voice assistants, tailored to their specific organizational goals. This technological evolution, however, introduces heightened security vulnerabilities such as voice spoofing attacks. To address voice spoofing challenges, various competitions like ASVspoof 2015, 2017, 2019, 2021, and ADD 2022 have emerged. ADD 2022's Track 1 aimed to classify genuine and fake speech signals in the presence of noise. Our exploratory data analysis revealed that for a given speech sample, noisy signals tend to occur within similar frequency bands. If a model is heavily reliant on data within frequency ranges that contains noise, its performance will be suboptimal. To address this issue, we propose a data augmentation technique called Frequency Feature Masking (FFM), which randomly masks frequency bands. FFM helps prevent overfitting and enhances the model's robustness by avoiding reliance on specific frequency bands. Furthermore, we propose a frequency band masking method using a bell-shaped filter. This allows for smooth transitions between masked and unmasked frequencies, enabling the model to naturally mimic frequency variations in real speech signals. We compare the performance of various data augmentation methods with FFM in two spoofing detection datasets, ASVspoof 2019 LA and ADD 2022. The proposed FFM augmentation achieves state-of-the-art results in both datasets. The ADD 2022 dataset showed an improvement of approximately 51% after the application of FFM, while there was a 54% improvement in the ASVspoof 2019 LA dataset. In addition, we have made the code and demo used in the experiment publicly available.

1. Introduction

The rapid advancement of speech technology has popularized voice assistants, tailored to diverse user needs depending on business goals. For instance, Amazon's Alexa supports transactions, while Samsung's Bixby manages device operations. However, as voice assistants grow more sophisticated, security vulnerabilities, such as manipulation through background noise or voice imitation, have become critical concerns. Voice anti-spoofing, a key area in automatic speaker verification, seeks to detect spoofed utterances created using various methods. To address these challenges, initiatives like the ASVspoof challenges [1–5] (since 2015) and the recent Audio Deep Synthesis Detection (ADD 2022, 2023) competitions [6,7] have advanced research in this domain.

Especially, in the context of ADD 2022 Track 1 competition, the primary objective was to develop models capable of detecting spoofing in complex acoustic environments characterized by significant background noise. Previous studies have been limited in their ability to adequately assess spoofing detection performance under such conditions due to a lack of data collected in noisy environments. Consequently, the ADD 2022 competition was specifically focused on the critical research challenge of spoofing detection in noisy settings. The influx of environmental noise complicates the composition of complex sound conditions. Voice samples, particularly those originated from noisy conditions, frequently manifest distinct noise patterns within specific frequency ranges, notably the high or low-frequency bands. However, if

* Corresponding authors.

E-mail addresses: soyul5458@snu.ac.kr (S. Han), seojaejin@cau.ac.kr (J. Seo), sc3377@cornell.edu (S. Choi), xodls4179@cau.ac.kr (T. Kang), cshtzon@korea.ac.kr (S. Chung), duneag2@korea.ac.kr (S. Lee), sonya1214@cau.ac.kr (S. Park), seungsang@korea.ac.kr (S. Oh), ikwak2@cau.ac.kr (I.-Y. Kwak).¹ Authors contributed equally.<https://doi.org/10.1016/j.jestch.2025.101972>

Received 8 October 2024; Received in revised form 2 January 2025; Accepted 23 January 2025

Available online 5 February 2025

2215-0986/© 2025 The Authors. Published by Elsevier B.V. on behalf of Karabuk University. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

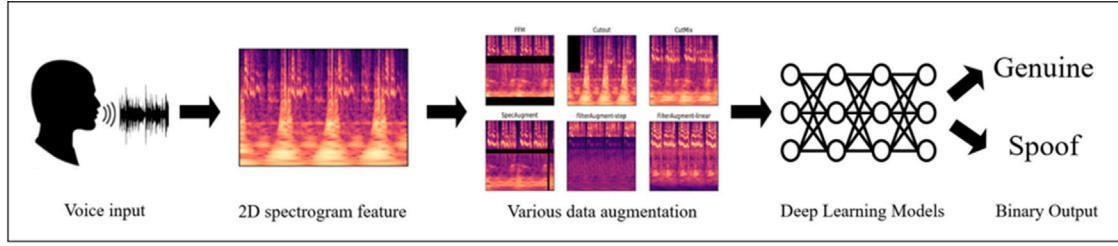


Fig. 1. Proposed frameworks.

our spoofing detection model heavily relies on these noise-prone bands, its accuracy could diminish when exposed to similar noisy data.

To address this challenge, we propose a novel data augmentation technique called Frequency Feature Masking (FFM) [8], designed to enhance the robustness of models in noisy environments. Observations of noisy audio samples revealed that the frequency bands containing human voice differ from those dominated by noise. Based on this insight, FFM selectively applies less masking to regions containing human voice and more masking to noise-dominated areas, encouraging the model to focus on voice-relevant features. Despite its simplicity of randomly masking frequency bands, the FFM technique demonstrated remarkable effectiveness on the noisy dataset from the ADD 2022 competition. Notably, our approach secured third place among 42 teams, with only three surpassing the baseline system [6], highlighting the significance of FFM in advancing the field.

All models examined within this study are based on spectrogram images. Initially, we employ the mel-spectrogram and the constant-Q transform (CQT) to convert 1D raw waveform audio input data into 2D spectrogram features. An overview of our proposed system is illustrated in Fig. 1. In assessing the potential of the FFM augmentation, six data augmentation techniques are applied to the spectrograms during the training process for comparison. These encompass Mixup [9], Cutout [10], CutMix [11], SpecAugment [12], FilterAugment [13], and our FFM augmentation. We then utilize the augmented data to train five deep neural networks: Light CNN (LCNN) [14], ResMax [15], Double Depthwise Separable (DDWS) [16], BC-ResMax [16], and Overlapped Frequency-Distributed (OFD) models [17].

Our study demonstrates the efficacy of the FFM augmentation method in significantly enhancing model performance. On the ADD 2022 competition dataset, FFM combined reduced EERs from initial values of 20.14%–22.53% to as low as 11.06%–15.16%. Similarly, for the ASVspoof 2019 LA dataset, the best EER decreased from 2.68% (baseline) to 1.49% with FFM. Comprehensive experiments showed FFM's consistent superiority across various models and conditions, underscoring its robustness in improving model accuracy.

The paper is organized into six sections. Section 1 elucidates the motivating factors behind the study and succinctly outlines the experimental outcomes designed to tackle the addressed issues. In Section 2, the paper delves into an exploration of various data augmentation techniques utilized within this study. Moving on, Section 3 meticulously delineates the FFM implementation and five different voice deep fake detection models that we used. Section 4 comprehensively expounds upon the dataset employed, the experimental intricacies, resultant findings, and conducted ablation studies. Section 5 critically examines deviations from prior research and underscores the indispensability of data augmentation within the realm of deep learning investigations. Finally, Section 6 encapsulates the essence of the proposed method and its associated experimental outcomes, while also presenting the challenges that require attention in forthcoming research endeavors. In addition, we have made the code and demo used in the experiment publicly available.²

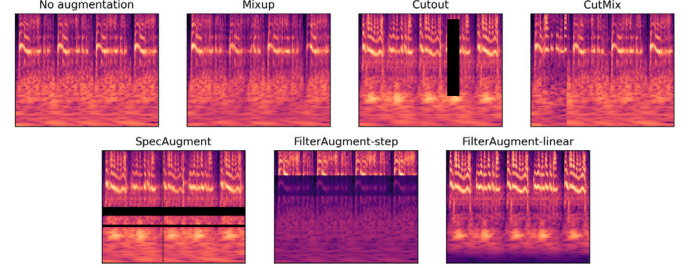


Fig. 2. Examples of various augmented spectrogram images using recent augmentation techniques.

2. Previous research on data augmentations

This section provides an overview of various data augmentation techniques that have been employed in research domains including sound event detection and audio classification. These techniques aim to mitigate overfitting and enhance the generalized performance of deep learning models during training (see Fig. 2).

2.1. Mixup

Mixup augmentation involves the linear interpolation between pairs of input samples and their corresponding labels to generate newly augmented training instances. Mixup is a widely used data augmentation method in voice classifications as well as image classifications [9]. The method mixes two different samples from the training set according to a parameter λ which is sampled from the $\text{beta}(\alpha, \alpha)$ distribution with a hyperparameter α . The method is as follows:

$$\tilde{x} = \lambda x_A + (1 - \lambda)x_B, \quad (1)$$

$$\tilde{y} = \lambda y_A + (1 - \lambda)y_B, \quad (2)$$

where (x_A, y_A) and (x_B, y_B) are two distinct samples. Here, (x_A, x_B) are two different input spectrograms, and (y_A, y_B) are the one-hot encoded labels, respectively.

To provide a more specific example of the actual application of Mixup, if the batch size is 16, eight training samples are prepared by randomly pairing them into sets of two. Suppose the labels for two arbitrary pairs are $y_A = (0, 1)$ (bonafide), $y_B = (1, 0)$ (spoof), and a random sample λ from a $\text{beta}(0.7, 0.7)$ distribution is 0.3. The label for the mixup sample would be $\hat{y} = 0.3 \times (0, 1) + 0.7 \times (1, 0) = (0.7, 0.3)$. The mixup sample $\hat{x} = 0.3x_A + 0.7x_B$.

2.2. Cutout

Cutout is a data augmentation technique in which specific regions of a training image are randomly masked [10]. The parameter λ is selected from a beta distribution to regulate the ratio between the masked region and the entire image area. This parameter decides the size of the cutout box, denoted as r_w for width and r_h for height. Its

² https://github.com/ikwak2/Paper_FFM.

center coordinate, r_x and r_y , are extracted from a uniform distribution. Here, W and H signify the size of the width (the number of temporal frames) and height (the number of frequency bins) of the given image, respectively.

$$r_x \sim \text{Unif}(0, W), \quad r_w = W\sqrt{1 - \lambda}, \quad (3)$$

$$r_y \sim \text{Unif}(0, H), \quad r_h = H\sqrt{1 - \lambda}. \quad (4)$$

Cutout technique is executed on the image x_A by

$$\tilde{x} = M \odot x_A \quad (5)$$

where M is a binary mask in $\{0, 1\}^{W \times H}$, with its zero-region denoting the cutout box region.

In this context, both the image x_A and the mask M have the same size and the $\odot$ operation denotes an element-wise multiplication.

2.3. CutMix

CutMix integrates two distinct training images to construct a novel composite image [11]. It locally removes a segment from the original image and consequently combines a patch from another image into the removed region. This seamless amalgamation of two different images ensures consistent performance, regardless of whether they possess identical or different labels. The label of the generated samples through CutMix is assigned by mixing their labels in proportion to the areas occupied by the two images.

$$\tilde{x} = M \odot x_A + (1 - M) \odot x_B, \quad (6)$$

$$\tilde{y} = \lambda y_A + (1 - \lambda) y_B, \quad (7)$$

with the ratio $\lambda = 1 - \frac{r_w r_h}{WH}$.

2.4. SpecAugment

SpecAugment is a data augmentation technique that enhances the robustness of models to variations in both time and frequency domains. It consists of three main techniques: time warping, frequency masking, and time masking [12].

Time warping involves stretching and compressing the spectrogram along the time axis. This is achieved by moving segments of the spectrogram horizontally to create a non-linear transformation of time. Frequency masking involves masking out random frequency bands to simulate variations in frequency characteristics. Finally, time masking masks out random segments of time steps to help the model handle incomplete or noisy data.

These techniques collectively provide custom-tailored spectrogram data by masking the entire frequency bands or temporal intervals. This augmentation strategy enhances the model's robustness to various types of input variations.

2.5. FilterAugment

FilterAugment exhibits a refined advancement in comparison to the frequency masking technique proposed in SpecAugment [13]. While SpecAugment simply masks a short time and frequency range in spectrograms, FilterAugment leverages acoustic filters by applying distinct weights to diverse frequency bands. FilterAugment comes in two variations: linear-type and step-type. In the linear-type FilterAugment, a weight filter with a continuous profile is multiplied to spectrograms. This process can retain the continuous nature of spectrogram features. On the other hand, the step-type FilterAugment introduces multiple random weights to apply them to different frequency band intervals, resulting in a non-linear step-like characteristic. Nam et al. [13] demonstrates that the performance of the linear-type FilterAugment surpasses that of the step-type FilterAugment.

3. Methods

In ADD 2022's Track 1, we focused on detecting spoofing in environments with challenging acoustics, where environmental noise plays

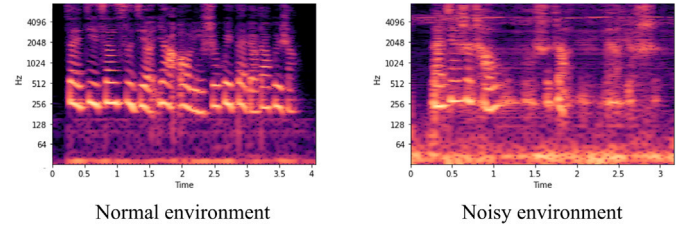


Fig. 3. Voice samples collected in normal and noisy environments.

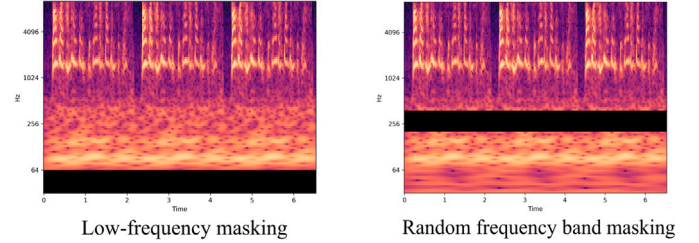


Fig. 4. Low-frequency masking and random frequency band masking.

a significant role. This noise often appears in specific frequency ranges, especially in low or high frequencies. The spectrograms in Fig. 3 visualize voice samples from an adaptation dataset obtained in both normal and noisy environments. Notably, the sample from the noisy environment exhibits a distinct long horizontal pattern at approximately 4096 Hz. Auditory analysis confirms the presence of persistent high-frequency noise infiltrating the sample. This finding is crucial for guiding the direction of our research.

In scenarios where our trained spoofing detection model places substantial reliance on these noise-incorporating frequency bands, the model's precision might exhibit degradation when confronted with the data showing similar noise attributes. To counter this, a strategic solution could involve the selective masking of frequency bands more prone to noise environment. This technique would enable the model to concentrate its attention on regions less affected by noise, thereby achieving an enhancement of its overall performance.

In response to this issue, we propose an innovative data augmentation technique "Frequency Feature Masking". This approach is designed to enhance the robustness of training for spoofing detection models within inherently noisy environments.

3.1. Frequency Feature Masking (FFM) augmentation

Building on the observation that noise is more prevalent in low-frequency and high-frequency bands than in the frequency range of human speech, we applied increased masking to noise-dominated regions. This approach ensures that less critical frequency bands are not overemphasized during the model training process. In the FFM augmentation, the following three methods are used in combination: "low frequency (LF) masking", "high frequency (HF) masking", and "random frequency band (RF) masking". Each method plays a crucial role in enhancing the model's robustness against noise by strategically masking different frequency components of the input data.

3.1.1. Low/high-frequency masking

During the training process, the augmentation using LF/HF masking to spectrogram data adheres to Algorithm 1. Given a training spectrogram image denoted as $X \in \mathbb{R}^{T \times F}$, where T and F symbolize the temporal and frequency dimensions correspondingly, the masking is executed with a probability of p . In the case of LF masking, the low-frequency range where masking is applied is restricted by limiting the mel bin within the range of 0 to l , and the minimum thickness of the

Algorithm 1 Low (or High)-frequency masking algorithm**Input:** Spectrogram X of the size $T \times F$ **Parameter:** Masking probability p , low (or high)-frequency area range l (or h), minimum band width m **Output:** Masked spectrogram $\tilde{X}$

- 1: **Low (or High) frequency masking function;**
 $LF(X)$ (or $HF(X)$)
- 2: $\tilde{X} \leftarrow X$
- 3: Generate u from $Unif(0, 1)$
- 4: **if** $u < p$ **then**
- 5: Randomly take an integer a from (m, l) (or $(h, F-m)$)
- 6: $\tilde{X}[:, :a] = 0$ (or $\tilde{X}[:, a:] = 0$)
- 7: **end if**
- 8: **return** $\tilde{X}$

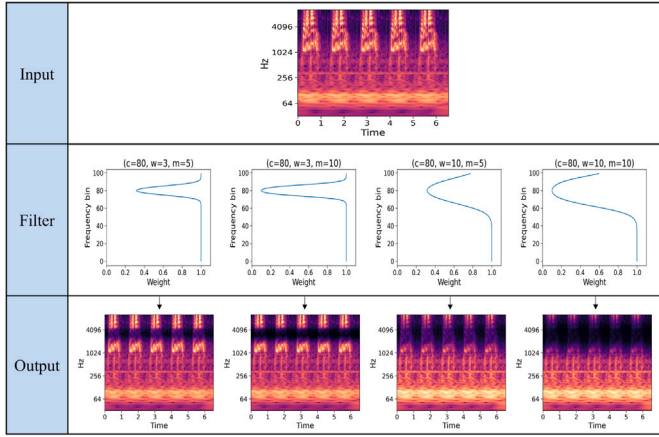


Fig. 5. Examples of various augmented spectrogram images using Bell-shaped frequency masking techniques.

masking bands is fixed as m ($m < l$). An integer a is randomly selected from (m, l) to compute the region for masking. Consequent to this, the low-frequency region $(0, a)$ is subjected to masking, with the values therein being reset to zero. LF masking is exemplified as illustrated in Fig. 4.

HF masking is the same as LF masking, but it is applied symmetrically reversed in the direction of the frequency axis. This ensures that high-frequency components are also masked, providing a balanced approach to frequency augmentation.

3.1.2. Random frequency band masking

RF masking is executed with a probability of p . The maximum number of masked frequency bands B , and both the width b_i and the lower frequency band location l_i of each band are randomly selected. A key feature of the RF masking approach is its dynamic band count. Unlike SpecAugment, which employ a fixed parameter for the number of masked bands, our method introduces variability by randomly selecting an integer from the range $(1, B)$. This variability enhances the augmentation process, thereby improving the model's capacity to generalize across diverse frequency conditions. This is visually depicted in Fig. 4. A detailed process is described in Algorithm 2.

3.2. Bell-shaped FFM augmentation

The Bell-shaped FFM augmentation method aims to improve the robustness of voice spoofing detection models in noisy environments

Algorithm 2 Random-frequency band masking algorithm**Input:** Spectrogram X of the size $T \times F$ **Parameter:** Masking probability p , maximum number of masked frequency bands B , minimum band width m , maximum band width M **Output:** Masked spectrogram $\tilde{X}$

- 1: **Random frequency band masking function;** $RF(X)$
- 2: $\tilde{X} \leftarrow X$
- 3: Generate u from $Unif(0, 1)$
- 4: **if** $u < p$ **then**
- 5: Randomly take an integer n from $(1, B)$
- 6: **for** i in range(n): **do**
- 7: Randomly take a band width b_i for i th band from integers (m, M)
- 8: Randomly take a band location l_i from integers $(0, F-b_i)$
- 9: $\tilde{X}[:, l_i : (l_i + b_i - 1)] = 0$
- 10: **end for**
- 11: **end if**
- 12: **return** $\tilde{X}$

Algorithm 3 Bell-shaped frequency masking algorithm**Input:** Spectrogram X of the size $T \times F$ **Parameter:** Masking probability p , frequency area range $L = [l_1, l_2]$, magnitude parameter of the bell function m , width parameter of the bell function U **Output:** Masked spectrogram $\tilde{X}$

- 1: **Bell-shaped frequency masking function;** $Bell(L, m, w)$
- 2: $\tilde{X} \leftarrow X$
- 3: Generate u from $Unif(0, 1)$
- 4: **if** $u < p$ **then**
- 5: Randomly select a width w from the range $(\frac{U}{2}, U)$
- 6: Randomly select c from L
- 7: $BellFilter = 10^{Bell(S, c, m, w)}$
- 8: $\tilde{X} \leftarrow \tilde{X} \cdot BellFilter$
- 9: **end if**
- 10: **return** $\tilde{X}$

by reducing information loss during frequency masking. Basic FFM augmentation randomly selects frequency bands and applies zero masking, which introduces variability but can lead to significant information loss. To address this limitation, we propose a Bell-shaped FFM augmentation that applies a continuous filter along the frequency dimension of the spectrogram. This method aims to retain more information while providing effective augmentation.

Fig. 5 illustrates an example of applying a Bell-shaped filter to the input spectrogram. To apply Bell-shaped masking along the frequency axis, we define a variant of the Gaussian function as the Bell function. Eq. (8) is used to generate the Bell-shaped filter:

$$Bell(S, c, m, w) = \frac{m}{10} \times e^{-\frac{1}{2} \left(\frac{S-c}{w} \right)^2}, \quad (8)$$

where $S = [0, 1, \dots, F-1]$ represents the frequency coordinates of the spectrogram feature. The center c of the Bell function is a randomly chosen integer from the pre-defined frequency index range $L = [l_1, l_2]$. A parameter w controlling the width of the Bell function is randomly selected within the range $(\frac{U}{2}, U)$. The magnitude of the bell-shaped augmentation is adjusted by the hyperparameter m .

The Bell function is centered around frequency c and approaches to 0 as it moves away from the center. To apply augmentation, we multiply $10^{Bell(S, c, m, w)}$ with the original spectrogram along the frequency direction. Further details can be found in Algorithm 3.

Table 1

The number of utterances in training, development, adaptation and test sets of the ADD 2022 database.

	Training	Dev.	Adaptation	Test
Genuine	3012	2307	300	–
Fake	24,072	21,295	700	–
Unlabeled	–	–	–	109,199

3.3. Voice deep fake detection models

We evaluate the efficacy of our proposed FFM augmentation technique by applying it to five distinct voice deep fake detection models: the Light CNN (LCNN) model [14], the ResMax model [15], the Double Depthwise Separable (DDWS) model [16], the BC-ResMax model [16], and the Overlapped Frequency Distributed (OFD) model [17]. All of these models are based on spectrogram images and have demonstrated their effectiveness in various ASVspoof competitions, including ASVspoof 2017, 2019, and 2021 [18–20].

4. Experiments

4.1. Feature engineering

We employed the librosa software [21] to perform CQT and mel-spectrogram feature extraction. In the case of CQT feature extraction, the parameters were configured with a minimum frequency of 5, 100 frequency bins, and a filter scale factor of 1. On the other hand, for mel-spectrogram feature extraction, we adopted 100 frequency bins, 1024 window lengths, and 512 hop sizes.

4.2. Datasets

4.2.1. ADD 2022 challenge datasets

The ADD 2022 challenge [6] introduced a more realistic approach to data generation, aiming to tackle spoofing attacks in real-world scenarios. This challenge encompassed three different tracks: the low-quality fake audio detection (LF) track, the partially fake audio detection (PF) track, and the audio fake game (FG) track. In our study, we focus specifically on track 1, which involves both genuine and spoof-attack data generated through text-to-speech (TTS) and voice conversion (VC) techniques. The LF track's objective is to classify genuine and spoofed utterances that have been mixed with various real-world noises and background music. Detailed information about the ADD 2022 challenge dataset can be found in Table 1. This dataset comprises training, development (Dev.), adaptation, and test data subsets. The training, development (Dev.), and adaptation datasets are publicly available, whereas the test data, intended for competition evaluation, does not have publicly disclosed labels. Therefore, for the evaluation in this study, we utilized only the training, development, and adaptation datasets.

4.2.2. ASVspoof 2019 challenge LA dataset

The ASVspoof 2019 challenge [4] is focused on TTS, VC and replay spoofing attack types. Its dataset comprises logical access (LA) and physical access (PA) scenarios. Both scenarios stem from the VCTK basic corpus [22], and the dataset is partitioned into three main segments: training, development, and evaluation. In this study, we focus on the LA data subset, as it aligns with the TTS and VC attack types emphasized in the ADD 2022 challenge dataset. It is important to note that the ASVspoof 2019 LA dataset is not designed for noisy environments; rather, it consists of clean speech data generated under controlled conditions. Within the LA scenario, a diverse set of 17 cutting-edge TTS and VC systems are employed to generate genuine and spoofed speech. Among these systems, six TTS and VC systems are classified by known attacks in the training and development data, while the

Table 2

The number of utterances in training, development and test sets of the ASVspoof 2019 LA database.

	Training	Dev.	Test
Genuine	2580	2548	7355
Fake	22,800	22,296	63,882

remaining eleven systems are labeled according to the unknown attacks in the test data. The ASVspoof 2019 LA dataset is detailed in Table 2, encompassing training, development (dev.), and test data. The training and development data incorporates approximately 25,000 utterances each, while the test data includes around 71,000 utterances containing unknown attacks.

4.3. Experimental setup

We conducted a thorough examination and comparison of our experimental results with five different spoofing detection models. To rigorously assess the performance, we employed FFM augmentation on two distinct databases: the ADD 2022 competition dataset for a noisy environment scenario, and the ASVspoof 2019 LA dataset, which consists of various types of synthetic speech samples without noise. Each utterance within these datasets was sampled at a sampling rate of 16 kHz. Our chosen hyperparameter configurations for FFM augmentation are as follows:

- For LF masking, the low-frequency area range parameter was set to $l = 12$, the minimum masking band width was adjusted to $m = 7$, and the probability p of applying LF masking ranged between 0.5 and 0.8.
- For HF masking, the high-frequency area range parameter was configured to $h = 80$, the minimum masking band width was set to $m = 13$, and the probability p of applying HF masking spanned from 0.5 to 0.8.
- For RF band masking, the masking was conducted randomly with a probability $p = \frac{2}{3}$. Two bands ($B=2$) were selected for masking with a probability of $1/3$, and one band was decided upon a probability of $1/3$. Each masking band width was randomly chosen within the range of $m = 8$ to $M = 12$.

Additionally, our chosen hyperparameter configurations for Bell-shaped FFM augmentation are as follows:

- The frequency range L to which the center of the Bell-shaped filter is applied was divided into three categories: *low*, *high*, and *all*. Specifically, for the *low* category, we used $L = [15, 24]$; for the *high* category, $L = [75, 84]$; and for the *all* category, the entire frequency range $L = [0, 99]$ was utilized.
- The spread parameter U of the Bell-shaped filter was set to 5 and 10, while the magnitude parameter m was configured to 5 and 10. The application probability of the Bell-shaped filter was designated to 0.5.

The Mixup hyperparameter for FFM augmentation ranged from 0.6 to 0.8, and we applied 0.7 for the Bell-shaped FFM.

The chosen evaluation metric is the equal error rate (EER), where the false acceptance rate (FAR) equals the false rejection rate (FRR). A lower EER is indicative of a superior model performance. Another evaluation metric to consider is the tandem Detection Cost Function (t-DCF). This metric is designed for the joint assessment of spoofing countermeasures and automatic speaker verification (ASV) systems [23]. It addresses the limitations of traditional EER-based evaluation by incorporating system costs and prior probabilities, making it more suitable for applications such as telephone banking, which involve varying spoofing attack and target user priors. For the ASVspoof 2019 dataset, the t-DCF score was also calculated and used for evaluation,

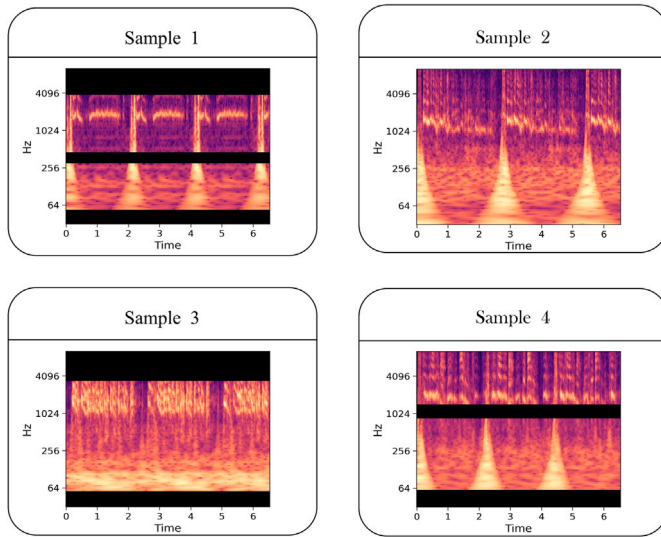


Fig. 6. Visualization of samples augmented with FFM combination ID 7.

Table 3
Augmentation ID allocation based on FFM combinations.

Aug ID	Mixup	LF	HF	RF
1	✓	✓		
2	✓		✓	
3	✓			✓
4	✓	✓	✓	
5	✓	✓		✓
6	✓		✓	✓
7	✓	✓	✓	✓

utilizing the provided ASV scores.

Training was conducted using 9-s audio samples. Audio exceeding this duration was truncated, while shorter samples were repeated to meet the required length. For training the proposed models, we selected a batch size of 16 and conducted 70 epochs of model training. We performed three separate training sessions and calculated the average EER values for the test dataset, recognizing that each experiment might converge to slightly different parameters. The learning rate was initialized to $1e-3$ and gradually decreased to $1e-5$ through a learning rate scheduler that utilized a sigmoidal decay function.

4.4. FFM technique and its implementation

Table 3 provides the identification numbers (IDs) associated with various FFM data augmentation techniques, allowing for clear differentiation between them. Fig. 6 visualizes four samples generated using Aug ID 7, which employs Mixup, LF, HF, and RF augmentations. Each masking technique is applied randomly based on a specified probability, resulting in diverse positions and widths of the masking bands, as illustrated in the sample examples in the figure.

The application of Mixup has been shown to improve performance, and it has been used in conjunction with other augmentation techniques in experiments. The process of applying Mixup and FFM, for example, involves randomly pairing 8 training samples out of a batch size of 16 into four sets of two. Assuming two pairs of labels, $y_A = (0, 1)$ (genuine) and $y_B = (1, 0)$ (spoof), and a random sample $\lambda = 0.3$ drawn from the $\text{beta}(0.7, 0.7)$ distribution, the mixed sample label becomes $\hat{y} = 0.3 \times (0, 1) + 0.7 \times (1, 0) = (0.7, 0.3)$. The mixed sample $\hat{x}$ is calculated as $\hat{x} = 0.3x_A + 0.7x_B$. Techniques such as LF, HF, and RF are then applied sequentially to $\hat{x}$.

Table 4

EER (%) on the adaptation set of ADD 2022 for each model with different combinations of augmentations.

Data	Aug ID	LCNN	ResMax	DDWS	BC-ResMax	OFD
ADD	None	20.14	21.88	22.53	22.31	21.35
	Mixup	20.14	16.77	17.44	17.59	17.12
	1	15.16	15.12	15.21	13.58	17.63
	2	18.49	17.34	15.96	15.39	16.93
	3	17.85	13.60	13.40	11.66	15.02
	4	19.31	18.02	15.33	13.24	17.56
	5	16.92	16.85	15.24	12.43	15.66
	6	17.02	17.44	13.68	13.21	15.49
	7	17.42	15.41	14.33	11.06	16.34

Table 5

EER (%) on the test set of ASVspoof 2019 LA for each model with different combinations of augmentations.

Data	Aug ID	LCNN	ResMax	DDWS	BC-ResMax	OFD
ASV	None	3.77	4.64	3.30	3.83	2.68
	Mixup	3.11	3.51	4.06	3.98	2.87
	1	2.88	3.60	3.28	3.20	2.58
	2	2.04	2.08	2.66	2.73	1.49
	3	3.52	4.57	6.65	5.17	3.25
	4	1.93	2.15	2.45	2.47	1.88
	5	3.30	4.97	4.47	4.99	3.32
	6	2.56	3.21	4.78	4.06	1.99
	7	2.45	3.78	4.33	4.34	2.49

5. Results

5.1. Results of FFM application in noisy environments

The experimental results presented in Table 4 demonstrate the effectiveness of various combinations of FFM augmentation techniques using five different spoofing detection models on the ADD 2022 dataset collected in noisy environments. This dataset, exposed to real-world noises and background music, exhibited overall high EERs. Models without any data augmentation recorded high EERs, while the application of Mixup augmentation resulted in moderate improvements. In contrast, the implementation of FFM augmentation led to significant reductions in EERs, with the BC-ResMax model achieving the most substantial improvement at 11.06%, thereby confirming the efficacy of FFM in noisy conditions.

5.2. Results of FFM application in normal environments

Table 5 presents the analysis results using the ASVspoof 2019 LA dataset, which consists of speech data from normal environments. This analysis focuses on determining the effectiveness of FFM augmentation techniques in typical conditions, serving as an ablation study.

For the ASVspoof 2019 LA dataset, the initial EERs for models without augmentations were relatively low. Mixup augmentation showed mixed results, with some EERs decreasing and others increasing. However, the integration of FFM augmentation (LF, HF) consistently enhanced performance across all models. The OFD model, in particular, achieved the lowest EER of 1.49%, highlighting the efficacy of FFM, especially when combining low-frequency and high-frequency masking. These results underscore the robustness of FFM augmentation techniques in improving the performance of spoofing detection models across different datasets and conditions.

When comparing ADD 2022 and ASVspoof 2019, we found that applying all combinations of LF, HF, and RF improved performance in ADD, compared to not applying them. However, in the general ASVspoof 2019 dataset, the use of RF actually degraded performance, while LF and HF alone resulted in improvements. Therefore, for developing a generic spoofing detection model, LF and HF can generally be expected to enhance performance. In contrast, RF might be recommended when there is a significant discrepancy between the training

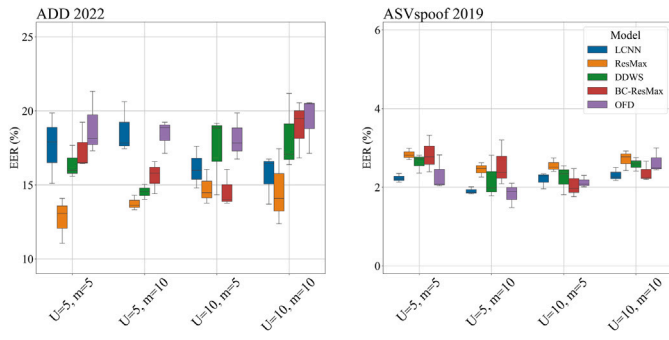


Fig. 7. EER (%) on the adaptation set of ADD 2022 and on the test set of ASVspoof 2019 LA with different combinations of Bell-shaped FFM hyperparameters.

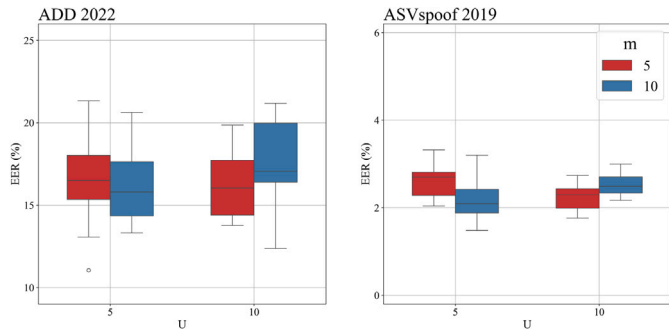


Fig. 8. Comparison of EER (%) with different combinations of Bell-shaped FFM hyperparameters.

and evaluation data, as seen with the ADD dataset, where leveraging more augmented data can help build a more robust model.

5.3. Hyperparameter search for bell-shaped FFM

The experimental results regarding Bell-shaped FFM are depicted in Fig. 7. For the ADD 2022 dataset, the LCNN, ResMax, DDWS, BC-ResMax, and OFD models using the Bell-shaped FFM showed EERs of 15.62%, 12.74%, 14.53%, 14.60%, and 18.14%, respectively. Particularly, the ResMax model demonstrated lower EERs compared to the results obtained with the conventional FFM, thereby validating the effectiveness of applying Bell-shaped FFM.

Similarly, on the ASVspoof 2019 LA dataset, the models utilizing the bell-shaped FFM achieved EERs of 1.90%, 2.45%, 2.19%, 2.07%, and 1.82%. Overall, the EERs of FFM and Bell-shaped FFM were comparable. However, Bell-shaped FFM demonstrated a lower EER compared to FFM in both LCNN and BC-ResMax models.

Fig. 8 reveals that when the parameter U is set to 5, higher values of m resulted in better performance. In contrast, when U is set to 10, lower values of m led to better performance. This suggests that stronger masking yields lower EERs when U is small, whereas weaker masking is more effective when U is large.

5.4. Comparison with other data augmentation methods

This study involves a comprehensive comparison of state-of-the-art data augmentation techniques and FFM. We employed techniques commonly used in the audio and speech-related fields, conducting experiments with the ASVspoof 2019 LA and ADD 2022 datasets. To ensure the fairness of the experiment, hyperparameter search was carried out for each data augmentation technique. The hyperparameter searches for the seven data augmentation techniques were performed based on the values presented in Table 6.

Table 6

Set hyperparameters of data augmentation.

Augmentation	Hyperparameters
Mixup, Cutout, CutMix	Parameters $\in [0.1, 0.9]$, intervals per 0.1
SpecAugment	Parameters $\in [[1, 5], [1, 5]], [[1, 10], [1, 10]], [[1, 10], [1, 15]], [[1, 15], [1, 10]], [[1, 15], [1, 15]], [[2, 5], [2, 5]], [[2, 10], [2, 10]], [[2, 10], [2, 15]], [[2, 15], [2, 10]], [[2, 15], [2, 15]]$
FilterAugment	# band = [2, 5], [3, 6], [5, 6] Minimum bandwidth = [2, 4, 6, 8, 10] dB range = [-10, 10], [-6, 6], [-1, 1]
FFM	Lowpass = [7, 8, 9, 10, 11, 12] Highpass = [80, 81, 82, 83, 84, 85, 86, 87] Randompass = [8, 9, 10, 11, 12]
Bell-shaped FFM	L = low: [15, 24], high: [75, 84], all: [0, 99] U = 5 or 10 m = 5 or 10

Table 7

Comparison of EER (%) results for different data augmentation methods in the ADD 2022 dataset.

Data augmentation	LCNN	ResMax	DDWS	BC-ResMax	OFD
None	20.14	21.88	22.53	22.31	21.35
Mixup	20.14	16.77	17.44	17.59	17.12
Cutout	17.75	20.15	19.61	19.33	19.03
CutMix	15.99	15.34	14.38	13.98	21.14
SpecAugment	19.18	18.42	19.53	16.46	19.73
FilterAug-step	16.38	16.32	13.57	17.24	16.42
FilterAug-linear	18.31	16.58	15.67	16.96	17.93
FFM (Ours)	15.16	13.60	13.40	11.06	15.02
Bell-shaped FFM (Ours)	15.62	12.74	14.53	14.60	18.14

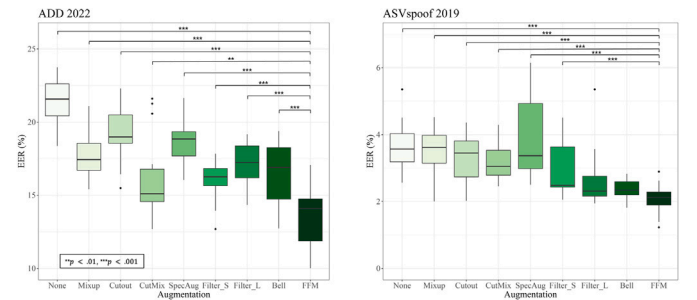


Fig. 9. Performance comparison of eight data augmentation techniques.

For Mixup, Cutout, CutMix, and SpecAugment, the most superior hyperparameters were selected by averaging the EER values obtained from three experiments for each parameter. FilterAugment (step, linear) explored optimal values by adding more parameters than those suggested by the author in the paper [13]. FFM and Bell-shaped FFM consist of three types of hyperparameters, and experiments were conducted by combining them. For each combination, experiments were performed three times, and the hyperparameters with the lowest average EER values were finally selected.

The results presented in Tables 7, 8, and Fig. 9 provide a comprehensive comparative analysis of the performance of eight data augmentation techniques applied to five deep learning models. This evaluation is designed to robustly assess the augmentation strategies across ten experimental combinations involving these five models and two distinct datasets. Notably, the proposed FFM demonstrated superior performance in the majority of scenarios, consistently outperforming all other techniques with the exception of a single case. This exceptional phenomenon may be attributed to the specific characteristics of certain models or datasets.

In each experimental scenario, experiments were conducted three times per augmentation method. The boxplot in Fig. 9 illustrates the

Table 8

Comparison of EER (%) and t-DCF results for different data augmentation methods in the ASVspoof 2019 LA dataset.

Data augmentation	LCNN		ResMax		DDWS		BC-ResMax		OFD	
	EER	t-DCF	EER	t-DCF	EER	t-DCF	EER	t-DCF	EER	t-DCF
None	3.77	0.1335	4.64	0.1299	3.30	0.0815	3.83	0.1387	2.68	0.0707
Mixup	3.11	0.0852	3.51	0.0917	4.06	0.0925	3.98	0.1235	2.87	0.0738
Cutout	3.06	0.1023	4.10	0.1434	3.76	0.0852	3.25	0.1189	2.20	0.0678
CutMix	3.56	0.122	2.68	0.0915	3.76	0.1283	2.70	0.0898	3.09	0.1003
SpecAugment	3.08	0.1168	4.55	0.1394	3.09	0.103	5.86	0.1823	3.00	0.0984
FilterAug-step	2.46	0.0858	3.76	0.1111	2.30	0.0761	2.58	0.0936	3.57	0.0857
FilterAug-linear	2.28	0.0928	3.31	0.0965	2.07	0.066	2.35	0.0865	3.21	0.0804
FFM (Ours)	1.93	0.0603	2.08	0.075	2.45	0.0738	2.47	0.0886	1.49	0.0632
Bell-shaped FFM (Ours)	1.90	0.071	2.17	0.093	2.19	0.0785	2.07	0.0865	1.82	0.0695

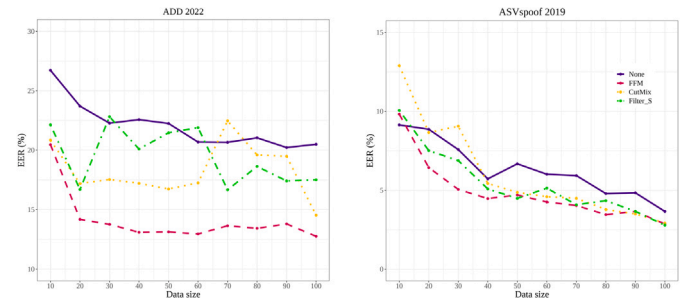
EER performance for each augmentation method. To evaluate the statistical significance of the performance difference between FFM and other methods, the Wilcoxon rank-sum test was employed with Bonferroni adjustment. In the case of the ADD dataset, FFM exhibited statistically significant performance improvements over other methods in all pairwise comparisons. Similarly, for the ASVspoof 2019 LA dataset, excluding the linear version of FilterAugment applied to DDWS, FFM consistently demonstrated significantly superior performance. The use of FFM increases masking in either low-frequency or high-frequency regions. As a result, regions outside the typical frequency range of human speech in the model are selectively masked, ultimately considered to contribute to the observed performance enhancement.

5.5. EER performance evaluation based on the data size

Data augmentation techniques such as FFM are primarily employed to enhance the diversity of training data. These are advantageous in improving the model's generalization capabilities and alleviating overfitting. In this subsection, we quantitatively evaluate the impact of FFM by applying it to datasets of varying sizes, empirically measuring its influence. The experiments were conducted using the training data from ADD 2022 and ASVspoof 2019 LA datasets, where the actual usage was scaled down to various ratios from 10% to 100% of the full dataset. In the ASVspoof 2019 LA dataset, speaker labels were available, allowing for data partitioning based on these labels. In the case of the ASVspoof 2019 LA dataset, speakers were randomly selected from 10% to 100% to adjust the dataset size. As for the ADD 2022 dataset, since speaker labels were not provided, data partitioning was conducted for samples. The experimental setup involved setting the Mixup hyperparameter α to 0.7 and applying FFM to 50% of the selected data. The BC-ResMax model was applied to these reduced datasets, and scenarios with and without data augmentation were compared. Experimental results based on data size are presented in Fig. 10.

In experiments using the ADD 2022 dataset, the performance was generally poorest when data augmentation was not applied across most data sizes. Although FilterAugment-step and CutMix showed overall superior performance compared to scenarios without data augmentation, a key finding was that FFM augmentation exhibited the most optimal EER performance. Comparing across different data sizes, using only 10% of the selected data without data augmentation resulted in a performance of 27%, which improved to 20% after applying FFM. Even when using 20% of the data, applying FFM led to a significant performance improvement from 24% to 14%. As data volume increased, performance stabilized, ultimately showing a substantial difference in performance, with approximately 20% without data augmentation and less than 13% with FFM.

In experiments utilizing the ASVspoof 2019 LA dataset, the performance consistently improved with the increase in data size, regardless of the application of data augmentation techniques. A key finding is that, when FFM augmentation was applied, it consistently demonstrated the most optimal EER performance across all data size scenarios. Overall, the model's performance showed a rapid improvement up to

**Fig. 10.** Comparing model performance based on the data size.

40% of the data size, followed by a more gradual inclination. Specifically for FFM, with only 10% of the data, the performance was around 10%, improving to 4.5% when 40% of the data was used. Ultimately, when the entire dataset was employed, the performance significantly improved to 2.9%.

In conclusion, the utilization of data augmentation techniques greatly contributed to the enhancement of spoofing detection models' performance in both ADD 2022 and ASVspoof 2019 LA datasets. Particularly in the case of ADD 2022, where the test set dealt with noisy conditions, the role of FFM augmentation in improving the model's performance was more pronounced. Therefore, in the practical implementation of spoofing detection models, it is advisable to effectively employ data augmentation techniques to account for various testing environments. Specifically, FFM augmentation shows superior performance in the experimental outcomes of spoofing detection scenarios.

6. Discussion

6.1. Comparison with the top-performing method from ADD 2022 competition

Our team CAU_KU participated in Track 1 'Low-quality Fake Audio Detection' of the Audio Deepfake Detection Challenge 2022 (ADD 2022), a grand challenge hosted by International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2022) [6]. Among 42 participating teams, we secured the 3rd position with an EER of 23.8%. Table 9 provides a detailed description of the ensemble system we submitted. For feature extraction at the front end, we utilized the CQT and mel-spectrogram techniques. For modeling, we trained five distinct architectures: LCNN, ResMax, DDWS, BC-ResMax, and OFD, leveraging FFM augmentation.

In the competition, the first-place system achieved a score of 21.7%, while the second-place system narrowly outperformed our model with a score of 23.0%. Furthermore, the fourth-place system attained a score of 25.9%. Upon examining the paper detailing the first-place system, it utilized a pretrained network on large-scale speech data, termed wav2vec 2.0, and fine-tuned it for the specific domain of voice spoofing

Table 9

EER (%) on the final evaluation data from the ADD 2022 challenge, and weights for ensemble model.

Model	Feature	Augmentation	EER	Weights
LCNN	CQT	Mixup, LF, RF	26.05%	0.20
ResMax	CQT	Mixup, RF	24.7%	0.27
DDWS	melspec	Mixup, RF	26.40%	0.20
BC-ResMax	melspec	Mixup, LF, RF	27.34%	0.13
OFD	CQT	Mixup	26.02%	0.20
Ensemble	–	–	23.8%	–

detection. A noteworthy similarity was the utilization of narrowband and wideband Finite Impulse Response (FIR) filters used by the first-place team [24]. The effects of these filters, which mask signals in the high-frequency and low-frequency bands, bear resemblance to the effects observed with our FFM's LF and HF mechanisms.

6.2. Comparison with other top-performing methods for ASVspoof 2019 LA

Table 10 displays a performance comparison with recently established spoofing detection systems. Our system employs a simple single model and achieves robust performance through our FFM augmentation technique. Our proposed FFM and Bell-shaped FFM methods have enhanced the performance of existing models, achieving low EERs ranging from 1.49 to 1.93. While their performance is higher compared to RawGAT-ST and AASIST models based on raw audio, they achieve top performance among spectrogram-based models.

6.3. Validation using the ASVspoof 2021 LA and DF datasets

To evaluate the generalization performance of the data augmentation techniques, additional experiments were conducted using the ASVspoof 2021 LA and DF datasets. These datasets are more technically challenging, providing a comprehensive evaluation of model performance. Unlike the promising results observed on the ASVspoof 2019 dataset, the additional experiments revealed that FFM and Bell-shaped FFM demonstrated comparable performance to other augmentation techniques on the ASVspoof 2021 datasets. This outcome highlights the need for further investigation into the characteristics of the ASVspoof 2021 data. Future research will focus on developing augmentation methods that perform effectively not only in noisy environments but also under general conditions.

6.4. Future works

The FFM technology has demonstrated significant performance improvements in the publicly available noise environment dataset, ADD 2022. However, performance validation across various types of noise environments has yet to be conducted. To address this gap, future research will involve testing the FFM technology on datasets that encompass a diverse range of noise types. This approach aims to comprehensively evaluate the performance and generalization capabilities of FFM under various acoustic conditions. Specifically, by systematically exploring the effectiveness of FFM across different noise scenarios, we seek to enhance its applicability and reliability in real-world applications.

This methodology not only aims to bolster the credibility of the research findings but also to demonstrate the adaptability of FFM technology in challenging environments, thereby contributing to advancements in the field of voice spoofing detection.

Table 10

Performance comparison of top-performing single systems on the ASVspoof 2019 LA evaluation.

Model	Front-End	EER (%)
SEnet [25]	CQT-MMPS-Compression, and channel	6.14
LCNN [26]	CQT-MMPS	5.99
RawNet2 [27]	SincNet filter	5.64
Resnet [25]	CQT-MMPS-Compression, and channel	5.18
LCNN [19]	LFCC	5.06
ResNet18-GAT-T [28]	LFCC-Frequency	4.71
LCNN [29]	CQT	4.07
LCNN+CE [30]	CQT-DASC	3.13
RW-ResNet [31]	Raw-audio	2.98
OFD [17] (our implementation)	CQT	2.68
LCNN (Ours)	CQT-FFM	1.93
LCNN-LSTM-sum [32]	LFCC	1.92
LCNN (Ours)	CQT-Bell-shaped FFM	1.90
OFD (Ours)	CQT-Bell-shaped FFM	1.82
OFD (Ours)	CQT-FFM	1.49
Non-OFD [17]	CQT	1.35
RawGAT-ST [33]	Raw-audio	1.06
AASIST [34]	SincNet filter	0.83

7. Conclusion

We have proposed a data augmentation technique designed to effectively enhance spoofing detection in real-world situations exposed to various noises. Our FFM augmentation introduces random masking of low-frequency or high-frequency bands, facilitating the model's focus on the frequency range specific to human speech signals, minimizing interference from environmental noises. To verify the efficacy of FFM, we conducted experiments on the ADD 2022 and ASVspoof 2019 LA datasets. FFM improved EER performance across five different deep learning models such as LCNN, DDWS, ResMax, BC-ResMax, and OFD. Applying FFM to the ADD 2022 dataset obtained in noisy environments led to an approximately 50% enhancement in the EER performance compared to the state before employing data augmentation techniques. This result indicates that FFM improves the model's capacity to cope with various noises and environmental conditions encountered in real-world scenarios.

The intriguing observation is that FFM are shown to be beneficial not only on the noisy ADD 2022 data but also on the ASVspoof 2019 LA dataset. Particularly, the HF augmentation of FFM masking of high-frequency bands played a more significant role in enhancing performance. In the case of the ASVspoof 2019 LA dataset, the high-frequency part might differ more from the training set compared to other frequency bands, or it could be less crucial for classification models. The performance improvement achieved through FFM augmentation differs from enhancing the model's performance by simply increasing the quantity of training data. It is attributed to considering the features obtained from diverse environments and implementing the model to effectively learn these features.

Additionally, we introduced Bell-shaped FFM, a novel variant of FFM employing continuous bell-shaped filters. It generated realistic augmentation samples and exhibited notable performance alongside Naive FFM. To evaluate the performance of FFM and Bell-shaped FFM augmentations, we conducted two comparative experiments. Firstly, we conducted experiments not only with cases where data augmentation was not used, but also with six commonly employed data augmentation techniques in the field of audio signal processing [10–13]. The evaluation results of this experiment demonstrated that FFM and Bell-shaped FFM exhibited significantly superior performance compared to existing data augmentation methods on the mentioned two datasets. Secondly, we conducted comparative experiments on the performance of FFM by gradually reducing the size of the training data from 100% to 10% of

the entire dataset. The results demonstrate a consistent reduction in overfitting and sustained performance improvements even with smaller training datasets.

In conclusion, the FFM and Bell-shaped FFM data augmentation techniques proposed in our study prove to be highly effective in robustly training acoustic models for spoofing detection tasks. These techniques are anticipated to continue contributing to the advancement of more accurate and reliable spoofing detection systems.

CRedit authorship contribution statement

Soyul Han: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Jaemin Seo:** Writing – original draft, Software, Methodology, Investigation, Formal analysis. **Sunmook Choi:** Writing – original draft, Software, Methodology, Investigation, Formal analysis. **Taein Kang:** Writing – original draft, Software, Methodology, Investigation, Formal analysis. **Sanghyeok Chung:** Writing – review & editing, Software, Investigation. **Seungeun Lee:** Writing – review & editing, Software, Investigation. **Seoyoung Park:** Writing – review & editing, Software, Investigation. **Seungsang Oh:** Writing – review & editing, Supervision, Methodology, Investigation, Funding acquisition, Conceptualization. **Il-Youp Kwak:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00208284, RS-2024-00341075) and the Global-LAMP Program of the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education, South Korea (No. RS-2023-00301976).

References

- [1] S.K. Ergunay, E. Khoury, A. Lazaridis, S. Marcel, On the vulnerability of speaker verification to realistic voice spoofing, in: 2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems, BTAS, IEEE, Piscataway, 2015, pp. 1–6, <http://dx.doi.org/10.1109/BTAS.2015.7358783>.
- [2] 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.
- [3] 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.
- [4] 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.
- [5] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K.A. Lee, T. Kinnunen, N. Evans, H. Delgado, ASVspoof 2021: accelerating progress in spoofed and deepfake speech detection, in: Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge, ISCA, Brno, 2021, pp. 47–54.
- [6] 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: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2022, pp. 9216–9220.
- [7] 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, 2023.
- [8] 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, 2022, pp. 9–17.
- [9] H. Zhang, M. Cisse, Y.N. Dauphin, D. Lopez-Paz, Mixup: Beyond empirical risk minimization, 2017.
- [10] T. DeVries, G.W. Taylor, Improved regularization of convolutional neural networks with cutout, 2017.
- [11] S. Yun, D. Han, S.J. Oh, S. Chun, J. Choe, Y. Yoo, CutMix: Regularization strategy to train strong classifiers with localizable features, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, IEEE, Seoul, 2019, pp. 6023–6032.
- [12] 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.
- [13] 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.
- [14] X. Wu, R. He, Z. Sun, T. Tan, A light CNN for deep face representation with noisy labels, IEEE Trans. Inf. Forensics Secur. (2018) 2884–2896, <http://dx.doi.org/10.1109/TIFS.2018.2833032>.
- [15] 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 11 (2023) 49140–49152, <http://dx.doi.org/10.1109/ACCESS.2023.3275790>.
- [16] J. Yang, S. Choi, Y. Lee, S. Oh, I.-Y. Kwak, Light-weight frequency information aware neural network architecture for voice spoofing detection, in: 26th International Conference on Pattern Recognition, ICPR, IEEE Computer Society, 2022.
- [17] S. Choi, I.-Y. Kwak, S. Oh, Overlapped frequency-distributed network: Frequency-aware voice spoofing countermeasure, in: Proc. Interspeech 2022, ISCA, Incheon, 2022.
- [18] 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.
- [19] 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.
- [20] 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.
- [21] B. McFee, C. Raffel, D. Liang, D.P. Ellis, M. McVicar, E. Battenberg, O. Nieto, Librosa: Audio and music signal analysis in python, in: Proceedings of the 14th Python in Science Conference, Vol. 8, SciPy, Austin, 2015, pp. 1–7.
- [22] 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>.
- [23] T. Kinnunen, K.A. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, D.A. Reynolds, T-DCF: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification, in: Odyssey 2018 the Speaker and Language Recognition Workshop, 2018.
- [24] 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, <http://dx.doi.org/10.1109/ICASSP43922.2022.9747768>.
- [25] A. Cohen, I. Rimon, E. Afalo, H.H. Permuter, A study on data augmentation in voice anti-spoofing, Speech Commun. 141 (2022) 56–67.
- [26] J. Yang, H. Wang, R.K. Das, Y. Qian, Modified magnitude-phase spectrum information for spoofing detection, IEEE/ACM Trans. Audio Speech Lang. Process. 29 (2021) 1065–1078.
- [27] 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, 2021, pp. 6369–6373, <http://dx.doi.org/10.1109/ICASSP39728.2021.9414234>.
- [28] H. Tak, J. Patino, A. Nautsch, N. Evans, M. Todisco, Graph attention networks for anti-spoofing, in: Proc. Interspeech 2021, 2021, pp. 2356–2360, <http://dx.doi.org/10.21437/Interspeech.2021-993>.
- [29] Z. Wu, R.K. Das, J. Yang, H. Li, Light convolutional neural network with feature genuinization for detection of synthetic speech attacks, 2020, arXiv preprint arXiv:2009.09637.
- [30] R.K. Das, J. Yang, H. Li, Data augmentation with signal companding for detection of logical access attacks, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2021, pp. 6349–6353.
- [31] Y. Ma, Z. Ren, S. Xu, RW-Resnet: A novel speech anti-spoofing model using raw waveform, in: Proc. Interspeech 2021, 2021, pp. 4144–4148, <http://dx.doi.org/10.21437/Interspeech.2021-438>.
- [32] X. Wang, J. Yamagishi, A comparative study on recent neural spoofing countermeasures for synthetic speech detection, in: Proc. Interspeech 2021, 2021, pp. 4259–4263, <http://dx.doi.org/10.21437/Interspeech.2021-702>.

- [33] H. Tak, J.-w. Jung, J. Patino, M. Kamble, M. Todisco, N. Evans, 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, 2021, pp. 1–8, <http://dx.doi.org/10.21437/ASVSPOOF.2021-1>.
- [34] 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, 2022, pp. 6367–6371.