





# Delayless Generative Fixed-Filter Active Noise Control Based on Deep Learning and Bayesian Filter

Zhengding Luo , *Student Member, IEEE*, Dongyuan Shi , *Senior Member, IEEE*,  
Woon-Seng Gan , *Senior Member, IEEE*, and Qirui Huang , *Senior Member, IEEE*

**Abstract**—The selective fixed-filter active noise control (SFANC) method can select suitable pre-trained control filters to attenuate incoming noises. However, the limited number of pre-trained filters is insufficient to effectively control various forms of noise, especially when the incoming noise differs much from the filter-training noises. To address this limitation and generate more appropriate control filters, a generative fixed-filter active noise control approach based on Bayesian filter (GFANC-Bayes) is proposed in this paper. The GFANC-Bayes method can automatically generate suitable control filters by combining sub control filters. The combination weights of sub control filters are predicted via a one-dimensional convolutional neural network (1D CNN). Based on prior information and predicted information, Bayesian filtering technique is applied to decide the combination weights. By considering the correlation between adjacent noise frames, the Bayesian filter can enhance the accuracy and robustness of predicting combination weights. Simulations on real-world noises indicate that the GFANC-Bayes method achieves superior noise reduction performance than SFANC and a faster response time than FxLMS. Moreover, experiments on different acoustic paths demonstrate its robustness and transferability.

**Index Terms**—Active noise control, Bayesian filter, convolutional neural network, deep learning, generative fixed-filter ANC.

## I. INTRODUCTION

ACOUSTIC noise problems are becoming more and more common with the increased number of industrial equipment in our daily lives, such as engines, fans, transformers, compressors, and other equipment [1], [2]. The attenuation of low-frequency noises is expensive and not practical for conventional passive noise control methods [3]. In comparison, active noise control (ANC) involves the electro-acoustic generation of an anti-noise to suppress unwanted noise. Ideally, the anti-noise is of equal amplitude and opposite phase to the disturbance, thus resulting in the cancellation of both signals [4]. Owing to its compact size and effectiveness in attenuating low-frequency

Manuscript received 28 March 2023; revised 23 October 2023; accepted 26 November 2023. Date of publication 4 December 2023; date of current version 12 January 2024. This was supported by the Singapore Ministry of Education Academic Research Fund Tier 2 under Grant MOE-T2EP50122-0018. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Stefan Goetze. (*Corresponding author: Dongyuan Shi.*)

The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: luoz0021@e.ntu.edu.sg; dongyuan.shi@ntu.edu.sg; ewsgan@ntu.edu.sg; huang.qirui@ieee.org).

Digital Object Identifier 10.1109/TASLP.2023.3337632

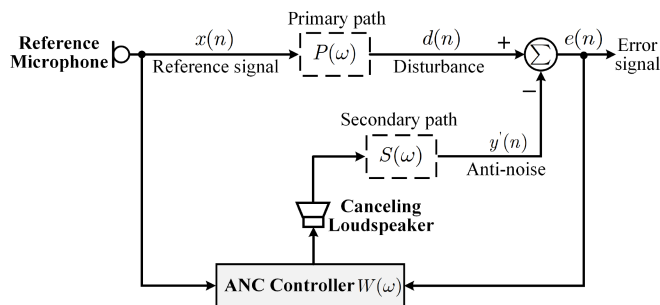


Fig. 1. Block diagram of a traditional feedforward ANC system, where  $\Sigma$  refers to the acoustic suppression.

noises, ANC has been widely used in many commercial products [5], [6], [7], [8] including headphones [9], ventilation duct, automobiles [10], etc.

Fig. 1 illustrates the block diagram of a single-channel feedforward ANC system, which comprises a reference microphone, a loudspeaker, an error microphone, and a controller. Traditionally, the control filter is updated based on the reference and error signals, which are respectively sensed by the reference microphone and error microphone [11]. To minimize the error signal, the coefficients of the control filter can be adjusted through adaptive algorithms [12]. Among adaptive ANC algorithms, the filtered-X least mean square (FxLMS) algorithm and its extensions are commonly used, as they can compensate for the secondary path delays and achieve a high computational efficiency [13].

However, conventional adaptive ANC algorithms are susceptible to instability due to imperfect practical factors such as inappropriate step size, output-saturation distortion, and feedback path effect [14], [15], [16]. Furthermore, the slow response speed of LMS-based ANC algorithms may negatively affect customers' perception of the noise reduction effect [6]. To alleviate these issues, many mature commercial products have adopted fixed-filter methods, in which the control filter coefficients are pre-trained and do not require updating based on error signals [17]. Although fixed-filter ANC can significantly improve response speed and robustness, it is only suitable for a specific noise type, resulting in degraded noise reduction performance for other types of noises [18].

As an improvement over traditional fixed-filter ANC methods, a selective fixed-filter active noise control (SFANC) method [19]

based on frequency band matching was proposed to choose different pre-trained control filters given different noises. Some crucial parameters of this method, however, can only be determined through trial and error. To overcome this limitation, deep learning techniques, particularly convolutional neural networks (CNNs) [20], [21], have emerged as promising methods to improve ANC performance. Leveraging the powerful learning capabilities of deep neural networks, deep learning-based ANC shows great potential in reducing various types of noise. Zhang et al. [22], [23] utilized a recurrent network to estimate cancelling signals for different primary noises and achieve low-latency noise control.

To enhance the stability and practicality of ANC, Shi et al. [24] proposed a deep learning-based SFANC algorithm. This SFANC method selects a suitable pre-trained control filter for the input noise via CNNs implemented in a co-processor, which can achieve a rapid response time and good robustness. Also, all the parameters are learned automatically from the noise dataset without additional human effort. However, the number of pre-trained control filters in the SFANC method is limited, which may result in unsatisfactory noise control performance for some noises, particularly those that differ much from the filter-training noises [25].

This paper proposes a generative fixed-filter active noise control approach with Bayesian filtering (GFANC-Bayes) to overcome the limitation of SFANC [24] and improve noise reduction performance. Initially, a pre-trained broadband control filter is decomposed into multiple sub control filters. During noise reduction, the co-processor generates a new control filter for real-time noise cancellation by adaptively combining sub control filters. A lightweight one-dimensional convolutional neural network (1D CNN) is developed to automatically predict the combination weights of sub control filters given the input noise. The predicted combination weights are then filtered by a Bayesian filtering module, which exploits the correlation information between adjacent noise frames to improve the prediction accuracy and robustness [26].

Unlike adaptive ANC systems, GFANC-Bayes can avoid instability and slow convergence since it does not require error signals to update. The GFANC-Bayes system can achieve delayless noise reduction through the efficient coordination between the co-processor and real-time controller. Furthermore, an adaptive labelling mechanism is proposed to reduce the manual effort required for labelling the noise dataset. Two types of combination weights including hard weights and soft weights are investigated in numerical simulations. The experimental results on real-recorded noises demonstrate that the GFANC-Bayes method achieves fast response speeds, low noise reduction errors, and high degrees of robustness.

This paper is structured as follows. Section II introduces the construction of sub control filters in the GFANC-Bayes method. The 1D CNN module and Bayesian filtering module are described in Sections III and IV, respectively. Section V details the process of real-time noise control. Section VI presents the simulation results of the proposed method. The final section concludes the paper.

## II. SUB CONTROL FILTERS IN GFANC-BAYES METHOD

The overall architecture of the proposed GFANC-Bayes approach is illustrated in Fig. 2. The co-processor operates at the frame rate and meanwhile the real-time controller operates at the sampling rate, which is referred to as the delayless structure. The input noise signal is initially captured by a reference microphone and then processed by an analog front end. Given each noise frame, the co-processor can generate a corresponding control filter by adaptively combining sub control filters. Thus, the construction of sub control filters is an essential stage in the GFANC-Bayes approach.

To obtain the sub control filters, we employed a practical filter decomposition method based on the theory of filter perfect-reconstruction [27]. Initially, we utilized the target ANC system to cancel a broadband primary noise  $\mathbf{x}(n)$ , whose frequency band contained the frequency components of interest. The FxLMS algorithm was adopted to derive the optimal control filter due to its low computational complexity [28]. The obtained optimal control filter is the pre-trained broadband control filter, which is the sole required prior data for the GFANC-Bayes system.

It is assumed that the pre-trained control filter has  $N$  taps, which is represented as  $\mathbf{c} = [c(0), \dots, c(n), \dots, c(N-1)]^T$ .  $\mathbf{c}$  can be decomposed into a perfect-reconstruction filter bank as outlined below. Through the discrete Fourier transform (DFT), its frequency spectrum is derived as

$$\mathbf{C} = \mathbf{F}_N \mathbf{c} = [C(0), \dots, C(k), \dots, C(N-1)]^T, \quad (1)$$

where  $\mathbf{F}_N$  denotes the DFT matrix. The control filter  $\mathbf{C}$  can be divided into  $M$  sub control filters as

$$\mathbf{C} = \sum_{m=1}^M \mathbf{C}_m, \quad (2)$$

in which the frequency spectrum of the  $m$ -th sub control filter can be expressed as

$$\mathbf{C}_m = [C_m(0), \dots, C_m(k), \dots, C_m(N-1)]^T, m \in [1, M]. \quad (3)$$

When  $m \neq M$ , the elements in the above vector are obtained from

$$C_m(k) = \begin{cases} C(k) & k \in [(m-1)I+1, mI] \\ & \cup [N-mI, N-1-(m-1)I] \\ 0 & \text{others;} \end{cases} \quad (4)$$

Differently, the frequency spectrum of the last sub control filter ( $m = M$ ) is computed as

$$C_M(k) = \begin{cases} C(k) & k \in [(M-1)I+1, N-1-(M-1)I] \\ 0 & \text{others,} \end{cases} \quad (5)$$

where the bandwidth of sub control filter is  $I = \lfloor \frac{N}{2M} \rfloor$ , where  $\lfloor \cdot \rfloor$  denotes the floor operation of the argument.

The time-domain representation of the  $m$ -th sub control filter can be expressed as

$$\mathbf{c}_m = \mathbf{F}_N^{-1} \mathbf{C}_m, \quad (6)$$

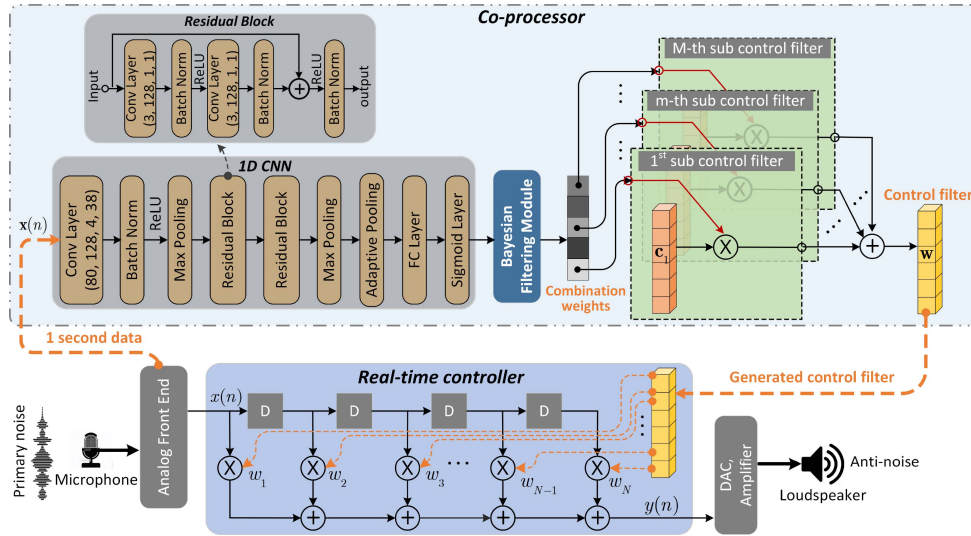


Fig. 2. Block diagram of the proposed GFANC-Bayes approach mainly consists of two modules: the generation of control filters in the co-processor and noise cancellation in the real-time controller. The configuration of the convolution layer is denoted as: (kernel size, number of channels, stride, padding).

where  $\mathbf{F}_N^{-1}$  denotes the inverse DFT matrix. The control signal  $y(n)$  of the ANC system is obtained from

$$\begin{aligned} y(n) &= \mathbf{x}^T(n) \mathbf{c} \\ &= \mathbf{x}^T(n) \mathbf{F}_N^{-1} \mathbf{F}_N \mathbf{c}. \end{aligned} \quad (7)$$

According to (1),  $y(n)$  can be rewritten to be

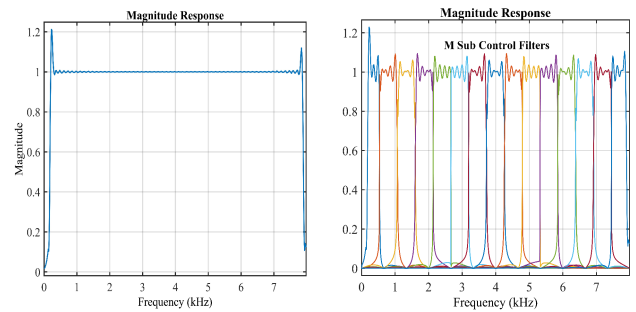
$$y(n) = \mathbf{x}^T(n) \mathbf{F}_N^{-1} \mathbf{C}. \quad (8)$$

Substituting (2) and (6) into (8) yields the control signal rewritten with the sub control filters  $\mathbf{c}_m$

$$\begin{aligned} y(n) &= \mathbf{x}^T(n) \mathbf{F}_N^{-1} \sum_{m=1}^M \mathbf{C}_m \\ &= \mathbf{x}^T(n) \sum_{m=1}^M \mathbf{c}_m. \end{aligned} \quad (9)$$

The above analysis indicates that the control filter  $\mathbf{c}$  can be perfectly reconstructed in the time domain by its all sub control filters  $\mathbf{c}_m$ . Therefore, by decomposing the pre-trained broadband control filter based on the theory of filter perfect reconstruction, multiple fixed sub control filters can be obtained. The frequency responses of the pre-trained broadband control filter and its  $M$  sub control filters are depicted in Fig. 3.

Through the weighted sum of sub control filters, the GFANC-Bayes method can generate various control filters with respect to the incoming noises. Compared to SFANC [19] with a fixed number of pre-trained control filters, GFANC-Bayes is able to generate a control filter with higher similarity in frequency content to the primary noise and thus achieve superior noise reduction performance, as demonstrated in Appendix A. Moreover, only one pre-trained control filter is required as the prior data in GFANC-Bayes, significantly reducing the effort for pre-training control filters.



(a) The pre-trained control filter

(b) Sub control filters

Fig. 3. Frequency bandwidths of the pre-trained broadband control filter and its sub control filters in the GFANC-Bayes method.

### III. THE PROPOSED ONE-DIMENSIONAL CNN

A lightweight 1D CNN is constructed in the co-processor to automatically predict the combination weights of sub control filters for various source noises. All parameters can be learned from noise data using the data-driven method. Also, the small size of the 1D CNN enables it to operate on less powerful co-processors such as mobile phones. In some audio applications [29], [30], it has been shown that CNNs employing time-domain waveforms can achieve comparable performance to those utilizing frequency spectrograms. Directly employing time-domain waveforms is more practical and convenient than employing frequency-domain data. Hence, the 1D CNN in GFANC-Bayes is designed to take the noise waveform as input and output its corresponding combination weight vector.

The input of the 1D CNN is a normalized one-second-long noise waveform. The input noise length of the network can be adjusted given specific application scenarios to achieve a balance between computational load and noise reduction performance. The normalization operation is defined as follows:

$$\hat{x}(n) = \frac{x(n)}{\max[\mathbf{x}(n)] - \min[\mathbf{x}(n)]}, \quad (10)$$

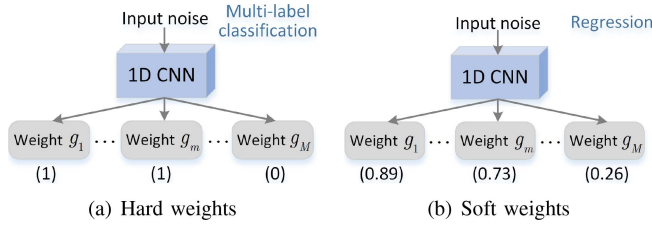


Fig. 4. Examples of hard weights and soft weights used to combine  $M$  sub control filters. Generating the hard weights and soft weights belong to multi-label classification and regression tasks, respectively.

where  $\max[\cdot]$  and  $\min[\cdot]$  represent the maximum and minimum values of the input vector  $\mathbf{x}(n)$ , respectively. The operation rescales the input range into  $(-1, 1)$  and retains the signal's negative part containing phase information, which is important for ANC applications [31].

The 1D CNN employs convolutional kernels of varying sizes to simulate a variety of bandpass filters to effectively extract features from the noise signal. With the ability to extract features, we can use the 1D CNN to obtain two types of combination weights, including hard weights and soft weights, as shown in Fig. 4. How to obtain the hard and soft weights using the 1D CNN will be explained below.

#### A. Hard Weights

The proposed 1D CNN, illustrated in Fig. 2, comprises a convolutional layer, two residual blocks, multiple pooling layers, and a fully connected (FC) layer. Each residual block consists of two convolutional layers, batch normalization, and ReLU non-linearity. Note that a shortcut connection is utilized to add the input with the output in each residual block, as the residual architecture is demonstrated to be easily optimized [32]. Additionally, to fully leverage both global and local information, the 1D CNN employs a broad receptive field (RF) in the first convolutional layer and narrow RFs in the subsequent layers.

The use of hard weights implies that the combination weights of sub control filters are binary, which helps to reduce the computational burden and memory requirements. The generation of hard weights belongs to a multi-label classification problem [33], in which a given instance may be associated with multiple labels. Deep learning approaches have demonstrated promising performance for multi-label classification [34]. To complete the multi-label classification task, the proposed 1D CNN has the following special configurations:

- The number of nodes in the output layer is  $M$ , which matches the number of labels.
- The Sigmoid layer is added to the output layer to perform element-wise function for all values:

$$\sigma(z) = \frac{1}{1 + \exp(-z)}, \quad (11)$$

where  $z$  refers to the input value, and the output value of Sigmoid function ranges from 0 to 1.

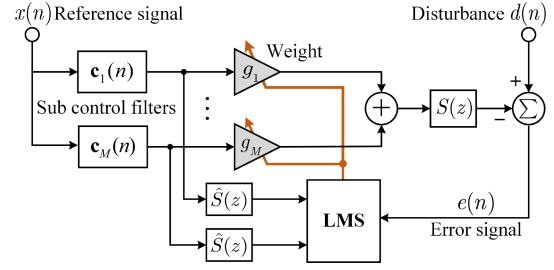


Fig. 5. Block diagram of adaptive labelling mechanism, in which the LMS algorithm is used to automatically label the training noise with the weight vector used for combining sub control filters.

- The Binary Cross Entropy (BCE) loss is used to train the proposed 1D CNN and expressed as:

$$\mathbb{L} = -\frac{1}{M} \sum_{m=1}^M [t_m \cdot \log p_m + (1 - t_m) \cdot \log (1 - p_m)], \quad (12)$$

where  $M$  denotes the number of labels of an instance.  $t_m$  and  $p_m$  represent the target probabilities and the predicted probabilities of  $m$ -th label, respectively.

- After training, the output values of the 1D CNN are converted to hard weights through rounding, resulting in binary values of 0 or 1.

#### B. Soft Weights

Soft weights differ from hard weights in that the weights used to combine sub control filters are numeric values ranging from 0 to 1. Hence, producing soft weights is a regression task. The most commonly used loss function for regression, the mean squared error (MSE) loss function [35], is utilized to train the 1D CNN. The MSE loss is computed as

$$\frac{\sum_{i=1}^n (p_i - t_i)^2}{n}, \quad (13)$$

where  $p_i$  and  $t_i$  denote the predicted value and its target value, and  $n$  represents the total number of samples that are used to compute MSE.

Following training, the output of the 1D CNN is directly used as the combination weights of sub control filters without further processing. In summary, the proposed 1D CNN can be trained to create hard or soft weights, depending on the labels of the dataset and the type of loss function during training. Notably, the generation of hard weights involves a rounding operation, whereas the generation of soft weights does not.

#### C. Adaptive Labelling Mechanism

To automatically label the noises used for training network with its corresponding combination weight vector, we have proposed an adaptive labelling mechanism illustrated in Fig. 5. The adaptive labelling mechanism simplifies the labelling process, making it more efficient and practical in actual ANC applications. If we assume that the input training noise is  $\mathbf{x}(n)$ , the output signal of the  $m$ -th fixed sub control filter is computed as

$$y_m(n) = \mathbf{x}^T(n) \mathbf{c}_m, m \in [1, M]. \quad (14)$$

The control signal  $y_g(n)$  is subsequently derived from the weighted sum of all sub control filter outputs:

$$\begin{aligned} y_g(n) &= \mathbf{y}^T(n)\mathbf{g}(n), \\ \mathbf{y}(n) &= [y_1(n), \dots, y_m(n), \dots, y_M(n)]^T, \\ \mathbf{g}(n) &= [g_1(n), \dots, g_m(n), \dots, g_M(n)]^T. \end{aligned} \quad (15)$$

where  $\mathbf{g}(n)$  refers to the combination weight vector.

Subsequently, the control signal passes through the secondary path to cancel the disturbance  $d(n)$ . The error signal is obtained from

$$e(n) = d(n) - y_g(n) * s(n), \quad (16)$$

where  $d(n)$  and  $s(n)$  represent the disturbance collected by the error microphone and the impulse response of secondary path;  $*$  represents the linear convolution.

To minimize the square of (16) through least mean square (LMS) algorithm [36], the updating formula of the weights is derived as

$$\begin{aligned} \mathbf{g}(n+1) &= \mathbf{g}(n) + \mu \mathbf{y}'(n)e(n), \\ \mathbf{y}'(n) &= \mathbf{y}(n) * s(n), \end{aligned} \quad (17)$$

where  $\mu$  stands for the step size.

Once (17) converges, the obtained combination weight vector can be directly treated as the soft weights vector  $\mathbf{S}$ :

$$\mathbf{S} = [g_1^o, \dots, g_m^o, \dots, g_M^o]^T, \quad (18)$$

where  $g_m^o$  represents the optimal combination weight of  $m$ -th sub control filter. Additionally, the soft weights vector can be rounded to obtain the hard weights vector  $\mathbf{H}$ :

$$\begin{aligned} \mathbf{H} &= [h_1, \dots, h_m, \dots, h_M]^T, \\ h_m &= \begin{cases} 1 & g_m^o \geq 0.5, \\ 0 & g_m^o < 0.5, \end{cases} \end{aligned} \quad (19)$$

The produced weight vectors  $\mathbf{S}$  and  $\mathbf{H}$  serve as the soft label and hard label of the training noise, respectively. With the soft-labeled and hard-labeled noise datasets, the 1D CNN model can be trained to predict soft weights and hard weights as illustrated in Fig. 4.

#### IV. BAYESIAN FILTERING MODULE

The pre-trained 1D CNN model can automatically output the combination weights of sub control filters for each frame noise. Each output value represents the predicted probability of employing the corresponding sub control filter. Moreover, its prior probability is provided from previous frames. With the prior probability and predicted probability, we can obtain the posterior probability of each sub control filter based on Bayesian theory [26]. This posterior probability is used as the decided combination weight of the sub control filter. Below is an explanation of the Bayesian filtering module.

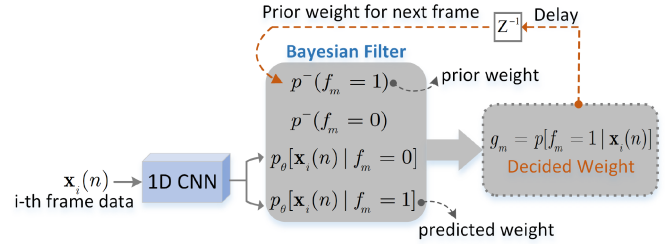


Fig. 6. Block diagram of Bayesian filtering module, where the combination weight of  $m$ -th sub control filter is decided by the prior weight and predicted weight based on Bayesian filter. The  $Z^{-1}$  represents a one-frame delay unit to retain the decided weight as the prior weight for the next frame.

#### A. Processing Within Bayesian Filtering

The Bayesian filtering module is depicted in Fig. 6. For  $i$ -th frame of noise  $\mathbf{x}_i(n)$ , it is assumed that the combination weight of  $m$ -th sub control filter is denoted as  $g_m$ . In the following analysis, we use  $f_m = 1$  to indicate the use of the  $m$ -th sub control filter, otherwise  $f_m = 0$ . For the  $m$ -th sub control filter, the predicted combination weight can be denoted as  $p_\theta[\mathbf{x}_i(n) | f_m = 1]$ , which is automatically obtained by the 1D CNN. Meanwhile, the prior combination weight  $p^-[f_m = 1]$  has been provided by the previous frame data. Initially, the prior combination weight is set to 0.5 for the first frame data. With the predicted weight and prior weight, the combination weight of the  $m$ -th sub control filter is decided based on the Bayesian filter:

$$\begin{aligned} g_m &= p[f_m = 1 | \mathbf{x}_i(n)] = \\ &= \frac{p_\theta[\mathbf{x}_i(n) | f_m = 1] p^-(f_m = 1)}{p_\theta[\mathbf{x}_i(n) | f_m = 1] p^-(f_m = 1) + p_\theta[\mathbf{x}_i(n) | f_m = 0] p^-(f_m = 0)}, \end{aligned} \quad (20)$$

where  $p_\theta[\mathbf{x}_i(n) | f_m = 0]$  and  $p^-(f_m = 0)$  are obtained from

$$\begin{aligned} p_\theta[\mathbf{x}_i(n) | f_m = 0] &= 1 - p_\theta[\mathbf{x}_i(n) | f_m = 1], \\ p^-(f_m = 0) &= 1 - p^-(f_m = 1). \end{aligned} \quad (21)$$

The pseudo-code for determining the combination weights of sub control filters is detailed in Table I. As shown in Table I, the prior weight vector is iteratively updated by the decided combination of sub control filters for each frame noise. In the Bayesian filtering module, previous information is also utilized to decide the combination weights in addition to the current information obtained by the current frame [24], [37]. Due to exploiting both the prior and predicted information, the prediction accuracy of the combination weights can be improved by the Bayesian filtering module [38]. Furthermore, the following section analyzes the effect of the Bayesian filtering module from a theoretical standpoint.

#### B. Theoretical Analysis of Bayesian Filtering

The accuracy of predicting the combination weights of sub control filters in GFANC-Bayes can be improved by using Bayesian filtering technology. Formally, the accuracy improvement can be theoretically analyzed as follows. For simplicity, we assume that the input signal of the 1D CNN is a scalar  $x$ .

TABLE I  
PSEUDO-CODE OF DECIDING THE COMBINATION WEIGHTS OF SUB CONTROL FILTERS

**Algorithm Description:** Deciding the combination weights of  $M$  sub control filters based on Bayesian theory in the GFANC-Bayes approach. CNN( $\cdot$ ) denotes the operation in the pre-trained 1D CNN model.  $\odot$  denotes the element-wise production.

**Input:** Assume that the reference signal has  $L$  seconds; the  $i$ -th frame data of reference signal is  $\mathbf{x}_i(n)$ ; the prior weight vector of length  $M$  is  $\mathbf{g}_o = [g_o(1), \dots, g_o(m), \dots, g_o(M)]$ , and its initial values are all 0.5.  
**Output:** The decided combination weights of sub control filters for  $i$ -th frame data is denoted as  $\mathbf{g}$ , with length  $M$ .  
**for**  $i$  in  $\{1, \dots, L\}$  **do**  
      $\hat{\mathbf{x}}_i(n) \leftarrow \text{Min-Max}(\mathbf{x}_i(n))$      $\triangleright$  Min-Max( $\cdot$ ) means the normalization (10).  
      $\hat{\mathbf{g}} \leftarrow \text{CNN}(\hat{\mathbf{x}}_i(n))$      $\triangleright$  Get the predicted weight vector by 1D CNN.  
      $\mathbf{g} \leftarrow (\hat{\mathbf{g}} \odot \mathbf{g}_o) / [\hat{\mathbf{g}} \odot \mathbf{g}_o + (1 - \hat{\mathbf{g}}) \odot (1 - \mathbf{g}_o)]$      $\triangleright$  Decide the combination weights of sub control filters based on Bayesian theory.  
      $\mathbf{g}_o \leftarrow \mathbf{g}$      $\triangleright$  Update the prior weight vector for the next frame.  
     **return**  $\mathbf{g}$      $\triangleright$  Return the decided combination weights of  $M$  sub control filters for  $i$ -th frame data.  
**end for**

For the analysis of the Bayesian filtering module, we have first introduced a prior probability distribution  $p^-(f)$ , which is a Gaussian distribution as

$$p^-(f) = \frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{(f-\mu_0)^2}{2\sigma_0^2}}. \quad (22)$$

where  $\mu_0$  and  $\sigma_0^2$  represent its mean and variance, respectively.  $f_m$  denotes the discrete state of the random variable value  $f$  and can be represented as

$$\begin{cases} f_m = 1 & f \geq \mu_0, \\ f_m = 0 & f < \mu_0. \end{cases} \quad (23)$$

The predicted combination weight by the 1D CNN is also assumed to be a Gaussian distribution as

$$p_\theta(x|f) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-f)^2}{2\sigma_1^2}}, \quad (24)$$

where  $f$  and  $\sigma_1^2$  denote its mean and variance, respectively.

From (22), (23), and (24), we can derive the multiplication of  $p_\theta(x|f_m = 1)$  and  $p^-(f_m = 1)$  as

$$p_\theta(x|f_m = 1)p^-(f_m = 1) = \int_{\mu_0}^{\infty} p_\theta(x|f)p^-(f)df. \quad (25)$$

In this case,  $f$  has a range of  $[\mu_0, \infty)$ . Then, we divide the range into  $K$  intervals and apply the mean value theorem [39] to (25). It can be rewritten as

$$p_\theta(x|f_m = 1)p^-(f_m = 1) = \sum_{k=1}^K p_\theta(x|\xi_k)p^-(\xi_k)\varepsilon, \quad (26)$$

where  $\xi_k \in (a_k, a_k + \varepsilon)$  and  $a_1 = \mu_0$ .  $a_k$  and  $\varepsilon$  refer to the left endpoint and length of the  $k$ -th interval, respectively. Meanwhile, we can get

$$p_\theta(x|f_m = 1)p^-(f_m = 1) + p_\theta(x|f_m = 0)p^-(f_m = 0)$$

$$\begin{aligned} &= \int_{\mu_0}^{\infty} p_\theta(x|f)p^-(f)df + \int_{-\infty}^{\mu_0} p_\theta(x|f)p^-(f)df \\ &= \int p_\theta(x|f)p^-(f)df. \end{aligned} \quad (27)$$

Substituting (26) and (27) into the Bayesian (20) yields the decided combination weight  $g_m$ :

$$\begin{aligned} g_m &= p(f_m = 1|x) \\ &= \frac{p_\theta(x|f_m = 1)p^-(f_m = 1)}{p_\theta(x|f_m = 1)p^-(f_m = 1) + p_\theta(x|f_m = 0)p^-(f_m = 0)} \\ &= \sum_{k=1}^K \varepsilon \left[ \frac{p_\theta(x|\xi_k)p^-(\xi_k)}{\int p_\theta(x|f)p^-(f)df} \right] \\ &= \varepsilon \sum_{k=1}^K p(\xi_k|x). \end{aligned} \quad (28)$$

Due to  $p(\xi_k|x) \sim N\left(\frac{\mu_0\sigma_1^2 + x\sigma_0^2}{\sigma_0^2 + \sigma_1^2}, \frac{\sigma_0^2\sigma_1^2}{\sigma_0^2 + \sigma_1^2}\right)$  derived in Appendix B, the distribution of  $g_m$  can be represented as

$$g_m \sim N\left(\frac{\mu_0\sigma_1^2 + x\sigma_0^2}{\sigma_0^2 + \sigma_1^2}K\varepsilon, \frac{\sigma_0^2\sigma_1^2}{\sigma_0^2 + \sigma_1^2}K\varepsilon^2\right). \quad (29)$$

If there is no Bayesian filtering module in the GFANC-Bayes system, the decided combination weight  $g_m$  will be the same as those output by the 1D CNN. In this case,  $g_m$  can be derived as

$$g_m = p_\theta(x|f_m = 1) = \varepsilon \sum_{k=1}^K p_\theta(x|\xi_k). \quad (30)$$

Since the variance of  $p_\theta(x|\xi_k)$  is  $\sigma_1^2$  given in (24), the variance of  $g_m$  can be derived as

$$\mathbb{D}(g_m) = \sigma_1^2 K \varepsilon^2. \quad (31)$$

Comparing the variances of the two normal distributions (29) and (31), we can find as

$$\frac{\sigma_0^2\sigma_1^2}{\sigma_0^2 + \sigma_1^2}K\varepsilon^2 = \frac{1}{1 + \frac{\sigma_1^2}{\sigma_0^2}}\sigma_1^2 K \varepsilon^2 < \sigma_1^2 K \varepsilon^2. \quad (32)$$

The comparison demonstrates that using the Bayesian filtering module to decide the combination weight brings about a lower variance compared to not using it. On the basis of this finding, it is suggested that the integration of the Bayesian filtering module can improve the accuracy of the prediction of the combination weights of sub control filters, and can generate more appropriate control filters. Additionally, using Bayesian filtering technology to decide the combination weights can improve the robustness of the system by reducing the impact of random variations from incoming noises.

## V. DELAYLESS NOISE CANCELLATION

For each frame noise,  $M$  sub control filters are summed based on the obtained combination weights to produce a corresponding

control filter  $\mathbf{w}(n)$ :

$$\mathbf{w}(n) = \sum_{m=1}^M g_m \mathbf{c}_m(n), \quad (33)$$

where  $n$  denotes the time index, and  $g_m$  represents the combination weight of  $m$ -th sub control filter  $\mathbf{c}_m(n)$ .

The generated control filter is subsequently provided to the real-time controller for noise control. The reference signal is processed by the generated control filter to obtain the control signal:

$$y(n) = \mathbf{x}^T(n) \mathbf{w}(n). \quad (34)$$

The control signal filtered by the secondary path is used to obtain the anti-noise wave for suppressing the disturbance  $d(n)$  in the listening environment. After noise reduction, the error signal picked up by the error sensor is given by

$$e(n) = d(n) - y(n) * s(n), \quad (35)$$

where  $s(n)$  is the impulse response of the secondary path,  $*$  represents linear convolution operation. In particular, unlike traditional adaptive ANC algorithms, GFANC-Bayes does not require the error signal to update its control filter, which minimizes the risk of divergence and increases its robustness.

Overall, the proposed GFANC-Bayes method is a novel approach to generate different control filters and achieve delayless noise control. The delayless noise reduction process of the GFANC-Bayes method can be summarized as follows:

- 1) The co-processor utilizes the 1D CNN to predict the combination weights, which are filtered by the Bayesian filtering module. Following this, a new control filter is generated by combining sub control filters using the combination weights.
- 2) To achieve delayless noise control, the co-processor operates at the frame rate while the real-time controller performs at the sample rate in parallel.
- 3) When the input noise frame changes, the generated control filter is adjusted accordingly, and the updated version is transmitted to the real-time controller.

This approach allows for the utilization of a high-performance co-processor (e.g., a laptop), operating at the frame rate to execute the 1D CNN. Concurrently, online noise control is carried out at the sampling rate on a less powerful processor. The GFANC-Bayes approach can generate suitable control filters given the input noises and achieve delayless noise control, making it a highly promising solution for practical ANC applications.

## VI. NUMERICAL SIMULATION RESULTS

In this section, a series of numerical simulations are conducted to assess the efficacy of the GFANC-Bayes approach in comparison to the SFANC and FxLMS algorithms.

### A. Experimental Setup

In the simulations, the pre-trained broadband control filter is decomposed into 15 sub control filters. The control filter has 1,024 taps, and the sampling rate is set to 16 kHz. Notably,

TABLE II  
PERFORMANCE COMPARISON OF DIFFERENT NETWORKS

Network	Testing Accuracy	#Parameters
1D Convolutional Neural Networks		
Proposed 1D CNN	<b>97.20%</b>	<b>0.21M</b>
M3 Network [30]	96.64%	0.22M
M5 Network [30]	96.70%	0.56M
M11 Network [30]	96.65%	1.79M
M18 Network [30]	94.35%	3.69M
M34-res Network [30]	96.74%	3.99M
2D Convolutional Neural Networks		
ShuffleNet v2 [41]	96.40%	0.25M
Mobilenet v2 [42]	96.05%	2.89M
Attention Network [43]	95.80%	4.95M

the primary and secondary paths utilized in simulation *B-F* are synthetic bandpass filters with a frequency range of 20-7,980 Hz, whereas the final experiment is conducted on real acoustic paths. The performance evaluations of GFANC-Bayes are conducted using real-recorded dynamic noises: an aircraft noise with a frequency range of 50 to 8,000 Hz and a traffic noise with a frequency range of 40 to 1,200 Hz. Notably, these noises from the real world are not included in the training dataset. In addition, two additional primary noises were generated by cascading and mixing the aircraft and traffic noises in the time domain.

### B. Effectiveness of the Proposed 1D CNN

1) *Training of the Network*: For training the network, a synthetic noise dataset is divided into 3 subsets: 80,000 noise tracks for training, 2,000 noise tracks for validation, and the remaining 2,000 noise tracks for testing. The synthetic noises are generated by filtering white noise through various band-pass filters with randomly chosen centre frequencies and bandwidths. Each noise track in the dataset has a 1-second duration. These noise tracks are seen as the reference signal and labelled with soft weights and hard weights by the adaptive labelling mechanism illustrated in Fig. 5. Additional noise (as described in Appendix A) is added to the reference signal to change its SNR as 30 dB.

During network training, the Adam algorithm was used for optimization. The number of training epochs was set as 50. The starting learning rate was 0.01, and it decreased by a factor of 5 after every five epochs. To prevent gradients from exploding or vanishing, Glorot initialization [43] was employed to initialize the CNN's parameters. Moreover, all network parameters were constrained to  $L_2$  regularization with a coefficient of 0.0001.

2) *Testing of the Network*: The generation of hard weights used to combine 15 sub control filters is a multi-label classification task. In the multi-label classification task, the proposed 1D CNN is compared against the M3, M5, M11, M18, and M34-res 1D networks [29]. As baselines, some lightweight 2D networks such as ShuffleNet v2 [40], Mobilenet v2 [41], and Attention Network [42] are also employed. The performance of these networks in the GFANC-Bayes approach is shown in Table II, where testing accuracy refers to the accuracy of predicting the hard weights for the testing noises. A noise instance is considered

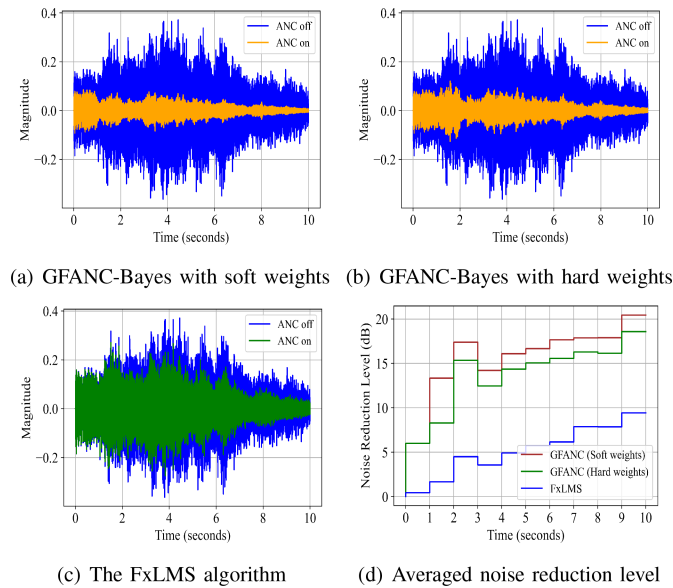


Fig. 7. (a)–(c): Error signals of different ANC algorithms, (d): Comparison of the averaged noise reduction level in each second, on the aircraft noise.

correctly classified only if all of its hard weights are correctly predicted.

On the test dataset, the proposed 1D CNN achieves the greatest prediction accuracy of 97.20%, indicating that it can effectively extract noise features to provide suitable hard weights for the input noises. A higher accuracy suggests that a more suitable control filter can be generated by combining sub control filters using the predicted hard weights. Therefore, the GFANC-Bayes system can generate more suitable control filters with the proposed 1D CNN compared to using other networks. Meanwhile, the proposed 1D network is lightweight, with only 0.21 M parameters, making it possible to be implemented on co-processors such as low-cost mobile devices. Furthermore, compared to using frequency-domain data in 2D networks, using time-domain waveform directly in 1D networks is more convenient and practical [30].

### C. Comparison of Soft Weights and Hard Weights

As discussed in Section III, the proposed 1D CNN trained with different methods can provide hard weights or soft weights. Also, the noise dataset with its labels of hard weights and soft weights is used to train the 1D CNN model for providing hard weights and soft weights, respectively. In this section, we conduct experiments to compare the noise reduction performance of using hard weights and soft weights to combine sub control filters in GFANC-Bayes.

Fig. 7 depicts the noise reduction effects of GFANC-Bayes with hard and soft weights on the real-recorded aircraft noise. The averaged noise reduction level is calculated every second. According to the results, the proposed GFANC-Bayes method with hard weights and soft weights can respond rapidly and achieve good noise reduction performance on the aircraft noise. Especially, using soft weights instead of hard weights to combine sub control filters yields superior results. Taking 1s–2 s

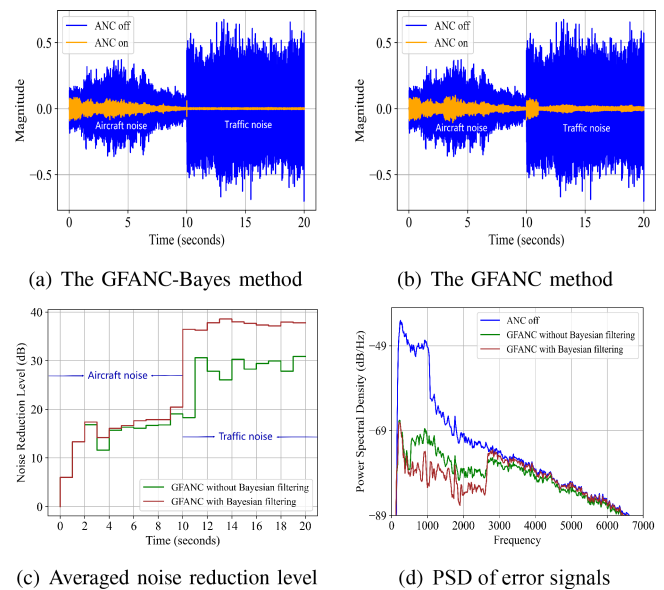


Fig. 8. Noise reduction performance of the GFANC method using soft weights with and without Bayesian filtering module, on the cascaded noise of aircraft and traffic noises.

as an example, the GFANC-Bayes algorithm with soft weights outperforms that with hard weights by around 5 dB in terms of averaged noise reduction level.

The choice between hard and soft weights depends on the particular objectives for noise reduction. Hard weights excel in computational efficiency due to their reliance on binary operations, but their noise reduction performance may be limited in certain situations. In contrast, soft weights offer a more nuanced degree of noise control at the cost of increased computational demands. Therefore, if the main goal is maximum noise reduction, opting for soft weights is advisable. Conversely, hard weights may be the preferred choice if computational efficiency is prioritized. In the following simulations, we opt for soft weights to combine sub control filters to achieve a higher level of noise reduction.

### D. Effectiveness of Bayesian Filtering Module

The Bayesian filtering module is utilized to determine the combination weights of sub control filters based on both prior weights and predicted weights. To ascertain the effectiveness of the Bayesian filtering module, we compared the noise attenuation results obtained by the GFANC method with and without the Bayesian filtering module. The cascaded noise of aircraft and traffic noises is used as the primary noise.

In Fig. 8, it is observed that the GFANC method with Bayesian filtering module can effectively track and attenuate different parts of the cascaded noise. Especially, during the transition period from one noise to the other (i.e. 10s–11 s), the GFANC method with Bayesian filtering significantly outperforms that without Bayesian filtering. This may be due to inaccurate 1D CNN predictions in the sudden transition from aircraft noise to traffic noise. The Bayesian filtering module can mitigate false predictions stemming from the 1D CNN, which plays an important role in improving robustness during noise control.



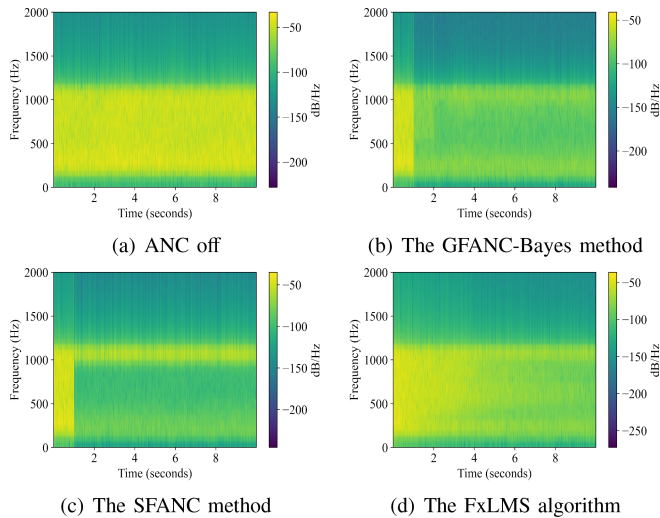


Fig. 9. Spectrogram of error signal obtained (a) without ANC, (b) by the GFANC-Bayes algorithm using soft weights, (c) by the SFANC algorithm, (d) by the FxLMS algorithm, on the traffic noise.

In terms of power spectral density (PSD), the GFANC method with Bayesian filtering shows superiority in reducing low-frequency noise components (20–2,700 Hz) compared to that without Bayesian filtering. The improvement is primarily attributed to the Bayesian filtering module’s utilization of both previous and predicted information. This allows for more accurate and robust prediction of the combination weights of sub control filters. As a result, the GFANC method with Bayesian filtering can generate more suitable control filters, thereby achieving better noise reduction performance.

### E. Comparison With the SFANC Method

The number of pre-trained control filters is limited in SFANC [44], which may result in unsatisfied noise control performance for some noises. In comparison, the GFANC-Bayes algorithm is able to generate different control filters for different input noises. To validate the superiority of GFANC-Bayes over SFANC, we conducted some comparison experiments on the traffic noise. The spectrograms of the noise signal attenuated by different ANC algorithms are presented in Fig. 9.

According to Fig. 9, it is found that the traffic noise is a broadband low-frequency noise, and most components of the traffic noise can be effectively attenuated when the GFANC-Bayes algorithm is used. However, SFANC is less effective at removing traffic noise components above 1,000 Hz. The reason is that SFANC can only choose control filters from a limited number of candidates, which may lead to unsuitable filters for some noise signals. In contrast, the GFANC-Bayes method is capable of generating more appropriate control filters to deal with different parts of the input noise. Also, we can observe that the FxLMS algorithm responds much slower in cancelling the traffic noise than GFANC-Bayes and SFANC.

Additionally, on the traffic noise, we examined the robustness of GFANC-Bayes and SFANC when dealing with a varying primary path that changes every 10 seconds. The primary path is mixed with additional white noises to adjust its power ratios

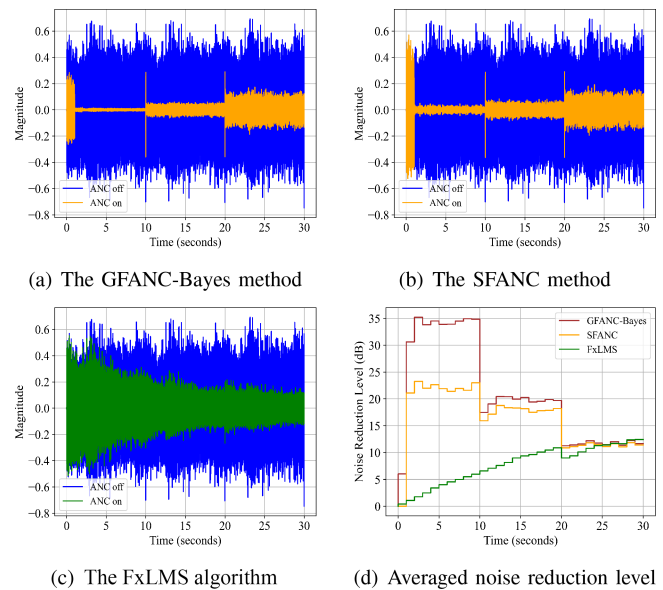


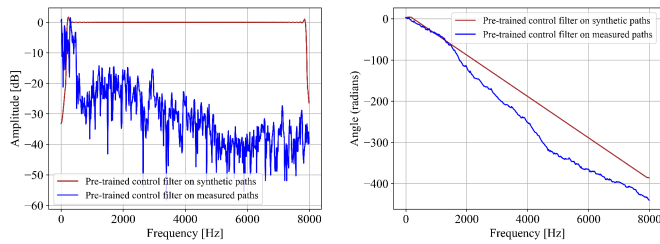
Fig. 10. Error signals (a)–(c) and Averaged noise reduction level in each second (d), when using different ANC algorithms to attenuate the traffic noise on the varying primary path.

(power of the primary path vector to that of noise) to be 50 dB in the second 10 seconds and 30 dB in the last 10 seconds. The performance of GFANC-Bayes and SFANC on the varying primary path are shown in Fig. 10. Although the variations of the primary path degrade the performance of GFANC-Bayes and SFANC, they still perform noise cancellation to some extent. The result shows the robustness of GFANC-Bayes and SFANC on the slight variations of the primary path.

### F. Comparison With the FxLMS Algorithm

In Fig. 7, the GFANC-Bayes approaches with hard weights and soft weights are compared to the FxLMS algorithm with a step size of 0.0001. It is found that GFANC-Bayes with hard weights or soft weights consistently outperforms the FxLMS algorithm. Noticeably, the GFANC-Bayes approach can respond to the noise much faster and achieve lower errors than the FxLMS algorithm. In the initial stage, GFANC-Bayes with soft weights achieves an averaged noise reduction level of 13 dB in 1s–2 s, whereas the FxLMS algorithm only obtains 2 dB. The averaged noise reduction level during 9s–10 s achieved by GFANC-Bayes with soft weights and GFANC-Bayes with hard weights are approximately 12 dB and 9 dB greater than that of FxLMS, respectively.

Additionally, Fig. 10 illustrates the performances of GFANC-Bayes and FxLMS algorithms in response to the varying primary path. Despite its adaptive capabilities, the FxLMS algorithm exhibits a slow convergence speed on the varying primary path. Conversely, the control filters generated by the GFANC-Bayes method can rapidly attenuate the noise. Furthermore, the FxLMS algorithm relies on error signals to update control filters, resulting in a risk of divergence. Differently, the absence of feedback error signals in the GFANC-Bayes method helps prevent instability. Our prior work [37] showed that an effective combination



(a) Magnitude Response Comparison (b) Phase Response Comparison

Fig. 11. Frequency spectrum comparison of the pre-trained broadband control filters on different acoustic paths.

 TABLE III  
 NMSE (IN DB) RESULTS OF THE GFANC-BAYES AND SFANC METHODS

Algorithm	Synthetic acoustic paths		Measured acoustic paths	
	Aircraft	Traffic	Aircraft	Traffic
GFANC-Bayes	-14.30	-16.15	-12.97	-15.03
SFANC [45]	-6.98	-9.94	-11.75	-9.69

\* NMSE is usually negative, and a lower value indicates superior performance.

\*\* Aircraft and traffic represent aircraft noise and traffic noise, respectively.

of fixed-filter algorithms and adaptive algorithms contributes to improving overall noise reduction performance.

### G. Noise Cancellation on Measured Acoustic Paths

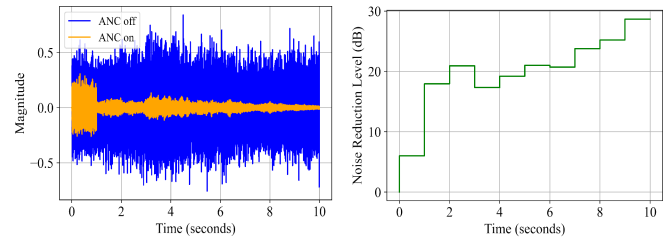
In the above simulations, the primary and secondary paths are synthetic bandpass filters due to the ease of adjustment. The acoustic paths used during evaluation are the same as those used during training. However, the acoustic paths in realistic applications are likely to be different from those used in training. In this simulation, to assess the transferability of GFANC-Bayes, we used the real acoustic paths measured from the vent of a noise chamber, as illustrated in Appendix C.

Firstly, the GFANC-Bayes and SFANC systems trained on synthetic acoustic paths are transferred into the measured acoustic paths. In GFANC-Bayes, we need to obtain the corresponding broadband control filter on the measured acoustic paths and decompose it into 15 sub control filters. The broadband control filters pre-trained on different acoustic paths are quite different, as illustrated in Fig. 11. In SFANC [44], we need to obtain 15 pre-trained control filters on the measured acoustic paths. Noticeably, the trained CNNs in the GFANC-Bayes and SFANC systems remain unchanged.

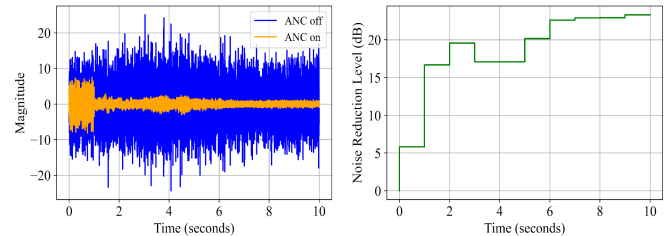
The performances of GFANC-Bayes and SFANC in terms of normalized mean squared error (NMSE) are summarized in Table III. NMSE in dB is defined as

$$\text{NMSE} = 10 \log_{10} \frac{\sum_{n=1}^L e^2(n)}{\sum_{n=1}^L d^2(n)}, \quad (36)$$

where  $L$  denotes the length of the signal vector. Table III shows that GFANC-Bayes and SFANC can attenuate real-world noises on different acoustic paths, but GFANC-Bayes consistently outperforms SFANC by a large margin.



(a) GFANC-Bayes on synthetic acoustic paths



(b) GFANC-Bayes on measured acoustic paths

Fig. 12. Noise reduction results of the GFANC-Bayes method used on (a) synthetic acoustic paths and (b) measured acoustic paths, where the primary noise is the mixed noise of aircraft and traffic noises.

Additionally, the error signals of GFANC-Bayes on different acoustic paths are presented in Fig. 12. The primary noise is formed by mixing aircraft and traffic noises. According to Fig. 12, GFANC-Bayes used on real acoustic paths has a slight performance degradation compared to that used on synthetic acoustic paths. Therefore, the simulations validate the effectiveness and transferability of the GFANC-Bayes method on real acoustic paths. Furthermore, using the trained GFANC-Bayes system in a new environment only requires the corresponding broadband control filter, making the method easy to implement in a variety of practical scenes.

## VII. CONCLUSION

In this paper, we propose the GFANC-Bayes method to generate appropriate control filters for different primary noises. The co-processor utilizes the 1D CNN to predict the combination weights of sub control filters for each noise frame. The Bayesian filtering module is then employed to decide the combination weights based on prior and predicted weights. After that, a new control filter is generated by combining sub control filters and sent to the real-time controller operating at the sampling rate. Due to the efficient collaboration between the co-processor and real-time controller, the GFANC-Bayes method can achieve delayless noise reduction.

In the simulations of real-world noises, the efficacy of the proposed 1D CNN and Bayesian filtering module is demonstrated. Due to the flexible generation of control filters, the GFANC-Bayes method reduces the traffic noise by approximately 10 dB more than the SFANC method. Also, it has a faster response time than the FxLMS algorithm because adaptive updating is avoided. Additionally, the GFANC-Bayes method shows good robustness and transferability when dealing with different acoustic paths.

These results imply that the GFANC-Bayes approach has the potential to be a practical noise reduction method.

#### APPENDIX A IMPROVEMENT OF GFANC OVER SFANC

Based on the work of SFANC [19], it is suggested that a pre-trained control filter can be chosen by considering the similarity of spectral content between the training noise and the actual primary noise. The spectral content, which encompasses the central frequency and bandwidth in the frequency domain, plays a crucial role. In a single-channel feedforward ANC system, as depicted in Fig. 1,  $P(\omega)$ ,  $S(\omega)$ , and  $W(\omega)$  correspond to the transfer functions of the primary path, the secondary path, and the control filter, respectively. For simplicity, it is assumed that the reference signal is directly obtained from the noise source.

In the offline pre-training stage of SFANC, a series of primary noises are utilized to obtain corresponding optimal control filters. Formally, a broadband training signal  $X_0(\omega)$  is input as a reference signal

$$X_0(\omega) = T_0(\omega) \cdot \text{Rect}\left(\frac{\omega - \omega_0}{2B_0}\right), \quad (\text{A.1})$$

where  $T_0(\omega)$  refers to a conjugate symmetric function with respect to frequency  $\omega$  [45].  $\omega_0$  and  $B_0$  represent the central frequency and bandwidth in the frequency domain, respectively. The rectangular function  $\text{Rect}(\cdot)$  is given by

$$\text{Rect}\left(\frac{\omega - \omega_0}{2B_0}\right) = \begin{cases} 1 & |\omega| \in [\omega_0 - B_0, \omega_0 + B_0] \\ 0 & \text{otherwise.} \end{cases} \quad (\text{A.2})$$

According to [19], the optimal control filter for the training signal can be expressed by

$$W_{\text{opt}}^0(\omega) = \frac{P(\omega)}{S(\omega)} \cdot \text{Rect}\left(\frac{\omega - \omega_0}{2B_0}\right). \quad (\text{A.3})$$

At the online control stage, the incoming primary noise assumed to be a broadband noise with the central frequency  $\omega_1$  and bandwidth  $B_1$  can be formulated as

$$X_1(\omega) = T_1(\omega) \cdot \text{Rect}\left(\frac{\omega - \omega_1}{2B_1}\right), \quad (\text{A.4})$$

where  $T_1(\omega)$  is a conjugate symmetric function with respect to frequency  $\omega$ . The rectangular function is given by

$$\text{Rect}\left(\frac{\omega - \omega_1}{2B_1}\right) = \begin{cases} 1 & |\omega| \in [\omega_1 - B_1, \omega_1 + B_1] \\ 0 & \text{otherwise.} \end{cases} \quad (\text{A.5})$$

Using the FxLMS algorithm, the optimal control filter for the primary noise can be derived as

$$W_{\text{opt}}^1(\omega) = \frac{P(\omega)}{S(\omega)} \cdot \text{Rect}\left(\frac{\omega - \omega_1}{2B_1}\right). \quad (\text{A.6})$$

In practice, apart from the primary noise  $x_1(n)$ , quantization noises, measurement noises, or electronic component interference may also exist in the ANC system. Considering the additional noise, the reference signal can be expressed as

$$x(n) = x_1(n) + v(n) \quad (\text{A.7})$$

where  $v(n)$  means a white Gaussian noise  $v(n) \sim \mathcal{N}(0, N_0/2)$ . If the pre-trained control filter  $W_{\text{opt}}^0$  is used to attenuate the primary noise, the error signal can be derived as

$$e(n) = d_1(n) - [x_1(n) + v(n)] * w_{\text{opt}}^0(n) * s(n), \quad (\text{A.8})$$

where  $d_1(k)$  is the disturbance. To simplify the analysis, we assume that  $\omega_0 = \omega_1$  and  $B_1 \leq B_0$ . Under this situation, the mean square error (MSE) of the error signal is expressed by

$$\mathbb{E}[e^2(k)] = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_e(\omega)_{\min} d\omega + \frac{N_0}{2\pi} (B_0 - B_1), \quad (\text{A.9})$$

where  $S_e(\omega)_{\min}$  is the power spectral density (PSD) of the error signal when using the optimal control filter  $W_{\text{opt}}^1(\omega)$ .

Therefore, it is concluded that if the input primary noise has the same central frequency and bandwidth as the training noise, using SFANC algorithm [19] can achieve the same noise reduction level as FxLMS algorithm. That is to say  $\omega_0 = \omega_1$  and  $B_0 = B_1$ . However, as described in [19], the number of  $B_0$  is limited in SFANC algorithm:

$$B_0 \in \{B_1^s, \dots, B_i^s, \dots, B_{15}^s\} i \in [1, 15], \quad (\text{A.10})$$

where  $B_i^s$  represents the available bandwidth of the training noises in SFANC. Although pre-training more control filters in SFANC can potentially increase the number of  $B_i^s$ , the number of  $B_i^s$  is still limited. Hence, there is a possibility that there might not be a matching  $B_0$  for a given  $B_1$  in SFANC.

To overcome the limitation of SFANC, we use the weighted sum of  $M$  sub-bandwidths  $B_m^g$  to generate arbitrary bandwidth  $B_0$  in the proposed GFANC method:

$$B_0 = \sum_{m=1}^M g_m B_m^g, \quad (\text{A.11})$$

where  $g_m$  means the corresponding combination weight of  $B_m^g$ . In summary, the number of  $B_0$  is limited in SFANC, but it is unlimited in the proposed GFANC method. A more suitable  $B_0$  can be automatically generated in the GFANC method to be matched with  $B_1$  so that better noise control performance can be obtained.

#### APPENDIX B DERIVATION OF THE POSTERIOR PROBABILITY $p(\xi_k|x)$

Given the (28), the posterior probability  $p(\xi_k|x)$  is computed as

$$p(\xi_k|x) = \frac{p_\theta(x|\xi_k)p^-(\xi_k)}{\int p_\theta(x|f)p^-(f)df}. \quad (\text{B.1})$$

Also, according to (22) and (24),  $p_\theta(x|\xi_k)$  and  $p^-(\xi_k)$  are computed as

$$p_\theta(x|\xi_k) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left[-\frac{(x - \xi_k)^2}{2\sigma_1^2}\right],$$

$$p^-(\xi_k) = \frac{1}{\sqrt{2\pi}\sigma_0} \exp\left[-\frac{(\xi_k - \mu_0)^2}{2\sigma_0^2}\right]. \quad (\text{B.2})$$

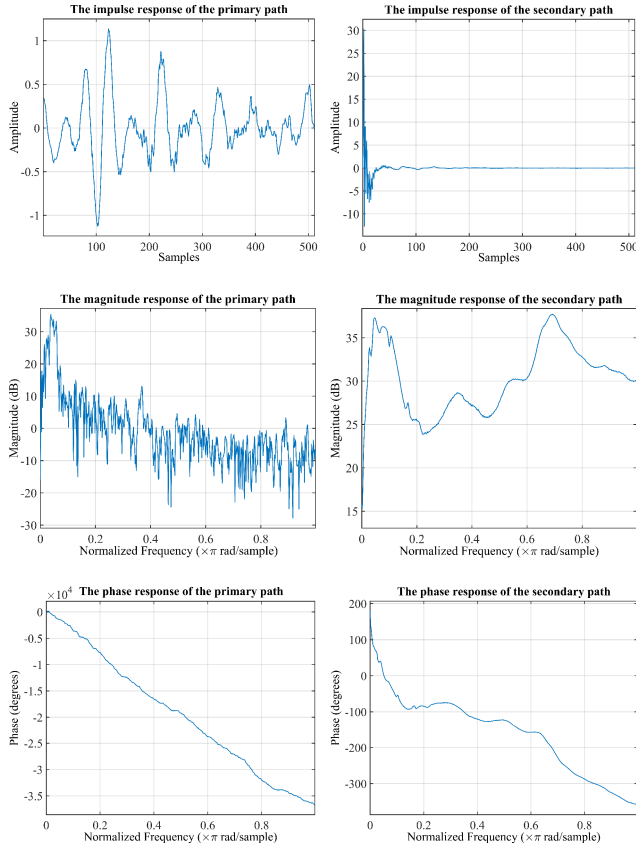


Fig. 13. The impulse responses, magnitude responses, and phase responses of the primary and secondary paths measured from the vent of a noise chamber.

Hence, the product of  $p_\theta(x|\xi_k)$  and  $p^-(\xi_k)$  can be expressed as

$$\begin{aligned} p_\theta(x|\xi_k)p^-(\xi_k) &= \frac{1}{2\pi\sigma_0\sigma_1} \exp\left[-\frac{(x-\xi_k)^2}{2\sigma_1^2} - \frac{(\xi_k-\mu_0)^2}{2\sigma_0^2}\right] \\ &= \frac{1}{2\pi\sigma_0\sigma_1} \exp\left[-\frac{(\xi_k-x)^2}{2\sigma_1^2} - \frac{(\xi_k-\mu_0)^2}{2\sigma_0^2}\right] \\ &= S \cdot \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(\xi_k-u)^2}{2\sigma^2}\right], \end{aligned} \quad (\text{B.3})$$

where

$$u = \frac{\mu_0\sigma_1^2 + x\sigma_0^2}{\sigma_0^2 + \sigma_1^2}, \sigma^2 = \frac{\sigma_0^2\sigma_1^2}{\sigma_0^2 + \sigma_1^2}, \quad (\text{B.4})$$

and

$$S = \frac{1}{\sqrt{2\pi}(\sigma_0^2 + \sigma_1^2)} \exp\left[-\frac{(x-\mu_0)^2}{2(\sigma_0^2 + \sigma_1^2)}\right]. \quad (\text{B.5})$$

Similarly, the product of  $p_\theta(x|f)$  and  $p^-(f)$  is given by

$$p_\theta(x|f)p^-(f) = S \cdot \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(f-u)^2}{2\sigma^2}\right]. \quad (\text{B.6})$$

Thus, the integration in (B.1) can be derived as

$$\int p_\theta(x|f)p^-(f)df = \int S \cdot \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(f-u)^2}{2\sigma^2}\right]df = S. \quad (\text{B.7})$$

Therefore, the posterior probability  $p(\xi_k|x)$  is computed as

$$\begin{aligned} p(\xi_k|x) &= \frac{p_\theta(x|\xi_k)p^-(\xi_k)}{\int p_\theta(x|f)p^-(f)df} \\ &= \frac{S \cdot \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(\xi_k-u)^2}{2\sigma^2}\right]}{S} \\ &= \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(\xi_k-u)^2}{2\sigma^2}\right], \end{aligned} \quad (\text{B.8})$$

it is demonstrated that  $p(\xi_k|x) \sim N(u, \sigma^2)$ , namely  $p(\xi_k|x) \sim N\left(\frac{\mu_0\sigma_1^2 + x\sigma_0^2}{\sigma_0^2 + \sigma_1^2}, \frac{\sigma_0^2\sigma_1^2}{\sigma_0^2 + \sigma_1^2}\right)$  [46].

## APPENDIX C MEASURED ACOUSTIC PATHS

See Fig. 13.

## REFERENCES

- [1] S. M. Kuo and D. R. Morgan, "Active noise control: A tutorial review," *Proc. IEEE*, vol. 87, no. 6, pp. 943–973, Jun. 1999.
- [2] C. R. Fuller and A. H. V. Flotow, "Active control of sound and vibration," *IEEE Control Syst. Mag.*, vol. 15, no. 6, pp. 9–19, Dec. 1995.
- [3] S. J. Elliott and P. A. Nelson, "Active noise control," *IEEE Signal Process. Mag.*, vol. 10, no. 4, pp. 12–35, Oct. 1993.
- [4] C. N. Hansen, *Understanding Active Noise Cancellation*. Boca Raton, FL, USA: CRC Press, 2002.
- [5] T. Schumacher, H. Krüger, M. Jeub, P. Vary, and C. Beaugeant, "Active noise control in headsets: A new approach for broadband feedback ANC," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2011, pp. 417–420.
- [6] Y. Kajikawa, W.-S. Gan, and S. M. Kuo, "Recent advances on active noise control: Open issues and innovative applications," *APSIPA Trans. Signal Inf. Process.*, vol. 1, 2012, Art. no. e3.
- [7] J. Cheer and S. J. Elliott, "Multichannel control systems for the attenuation of interior road noise in vehicles," *Mech. Syst. Signal Process.*, vol. 60, pp. 753–769, 2015.
- [8] S. Liebich, C. Anemüller, P. Vary, P. Jax, D. Rüschen, and S. Leonhardt, "Active noise cancellation in headphones by digital robust feedback control," in *Proc. IEEE 24th Eur. Signal Process. Conf.*, 2016, pp. 1843–1847.
- [9] M. Pawelczyk, "Analogue active noise control," *Appl. Acoust.*, vol. 63, no. 11, pp. 1193–1213, 2002.
- [10] L. Yin et al., "Adaptive parallel filter method for active cancellation of road noise inside vehicles," *Mech. Syst. Signal Process.*, vol. 193, 2023, Art. no. 110274.
- [11] X. Shen, D. Shi, and W.-S. Gan, "A wireless reference active noise control headphone using coherence based selection technique," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 7983–7987.
- [12] C.-Y. Ho, K.-K. Shyu, C.-Y. Chang, and S. M. Kuo, "Integrated active noise control for open-fit hearing aids with customized filter," *Appl. Acoust.*, vol. 137, pp. 1–8, 2018.
- [13] F. Yang, J. Guo, and J. Yang, "Stochastic analysis of the filtered-x LMS algorithm for active noise control," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2252–2266, 2020.
- [14] R. Ranjan and W.-S. Gan, "Natural listening over headphones in augmented reality using adaptive filtering techniques," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 11, pp. 1988–2002, Nov. 2015.
- [15] D. Shi, W.-S. Gan, B. Lam, Z. Luo, and X. Shen, "Transferable latent of CNN-based selective fixed-filter active noise control," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 2910–2921, 2023.
- [16] X. Shen, D. Shi, Z. Luo, J. Ji, and W.-S. Gan, "A momentum two-gradient direction algorithm with variable step size applied to solve practical output constraint issue for active noise control," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1–5.
- [17] J. Ji, D. Shi, Z. Luo, X. Shen, and W.-S. Gan, "A practical distributed active noise control algorithm overcoming communication restrictions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1–5.

- [18] C. Shi, R. Xie, N. Jiang, H. Li, and Y. Kajikawa, "Selective virtual sensing technique for multi-channel feedforward active noise control systems," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 8489–8493.
- [19] D. Shi, W.-S. Gan, B. Lam, and S. Wen, "Feedforward selective fixed-filter active noise control: Algorithm and implementation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 1479–1492, 2020.
- [20] H. Zhang and D. Wang, "Deep MCANC: A deep learning approach to multi-channel active noise control," *Neural Netw.*, vol. 158, pp. 318–327, 2023.
- [21] Z. Luo, D. Shi, J. Ji, and W.-S. Gan, "Implementation of multi-channel active noise control based on back-propagation mechanism," 2022, *arXiv:2208.08086*.
- [22] H. Zhang and D. Wang, "Deep ANC: A deep learning approach to active noise control," *Neural Netw.*, vol. 141, pp. 1–10, 2021.
- [23] H. Zhang, A. Pandey, and D. L. Wang, "Low-latency active noise control using attentive recurrent network," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 1114–1123, 2023.
- [24] D. Shi, B. Lam, K. Ooi, X. Shen, and W.-S. Gan, "Selective fixed-filter active noise control based on convolutional neural network," *Signal Process.*, vol. 190, 2022, Art. no. 108317.
- [25] Z. Luo, D. Shi, X. Shen, J. Ji, and W.-S. Gan, "Deep generative fixed-filter active noise control," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1–5.
- [26] J. M. Bernardo and A. F. Smith, *Bayesian Theory*, vol. 405. Hoboken, NJ, USA: Wiley, 2009.
- [27] H. H. Dam, S. Nordholm, A. Cantoni, and J. M. de Haan, "Iterative method for the design of DFT filter bank," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 51, no. 11, pp. 581–586, Nov. 2004.
- [28] D. Shi, W.-S. Gan, B. Lam, and C. Shi, "Two-gradient direction FXLMS: An adaptive active noise control algorithm with output constraint," *Mech. Syst. Signal Process.*, vol. 116, pp. 651–667, 2019.
- [29] W. Dai, C. Dai, S. Qu, J. Li, and S. Das, "Very deep convolutional neural networks for raw waveforms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 421–425.
- [30] E. Tsalera, A. Papadakis, and M. Samarakou, "Comparison of pre-trained CNNs for audio classification using transfer learning," *J. Sensor Actuator Netw.*, vol. 10, no. 4, 2021, Art. no. 72.
- [31] S. M. Kuo and D. R. Morgan, *Active Noise Control Systems*, vol. 4. New York, NY, USA, Wiley, 1996.
- [32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [33] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Mach. Learn.*, vol. 85, no. 3, pp. 333–359, 2011.
- [34] C.-K. Yeh, W.-C. Wu, W.-J. Ko, and Y.-C. F. Wang, "Learning deep latent space for multi-label classification," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 2838–2844.
- [35] X. Qiu, L. Zhang, Y. Ren, P. N. Suganthan, and G. Amaratunga, "Ensemble deep learning for regression and time series forecasting," in *Proc. IEEE Symp. Comput. Intell. Ensemble Learn.*, 2014, pp. 1–6.
- [36] X. Shen, D. Shi, W.-S. Gan, and S. Peksi, "Adaptive-gain algorithm on the fixed filters applied for active noise control headphone," *Mech. Syst. Signal Process.*, vol. 169, 2022, Art. no. 108641.
- [37] Z. Luo, D. Shi, and W.-S. Gan, "A hybrid SFANC-FxNLMS algorithm for active noise control based on deep learning," *IEEE Signal Process. Lett.*, vol. 29, pp. 1102–1106, 2022.
- [38] S.-K. Lai and Y.-t. Zhang, "Real-time prediction of noise signals for active control based on Bayesian forecasting and time series analysis," in *Proc. Inter-Noise Noise-Con Congr. Conf.*, 2019, vol. 259, pp. 728–738.
- [39] W. Rudin et al. *Principles of Mathematical Analysis*. New York, NY, USA, McGraw-Hill 1976, vol. 3.
- [40] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 116–131.
- [41] S. Adapa, "Urban sound tagging using convolutional neural networks," in *Proc. Detection Classification Acoust. Scenes Events Workshop*, 2019, pp. 5–9.
- [42] Z. Luo, J. Li, and Y. Zhu, "A deep feature fusion network based on multiple attention mechanisms for joint iris-periocular biometric recognition," *IEEE Signal Process. Lett.*, vol. 28, pp. 1060–1064, 2021.
- [43] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist. Workshop Conf.*, 2010, pp. 249–256.
- [44] Z. Luo, D. Shi, W.-S. Gan, Q. Huang, and L. Zhang, "Performance evaluation of selective fixed-filter active noise control based on different convolutional neural networks," in *Proc. Inter-Noise Noise-Con Congr. Conf.*, 2023, vol. 265, pp. 1615–1622.
- [45] S. M. Kuo, "Adaptive active noise control systems: Algorithms and digital signal processing (DSP) implementations," in *Proc. SPIE*, vol. 10279, pp. 26–52, 1995.
- [46] S. Thrun, "Probabilistic robotics," *Commun. ACM*, vol. 45, no. 3, pp. 52–57, 2002.