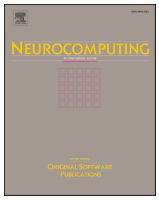
Neurocomputing 558 (2023) 126767

Contents lists available at ScienceDirect

Neurocomputing

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



IIRI-Net: An interpretable convolutional front-end inspired by IIR filters for speaker identification

Hossein Fayyazi, Yasser Shekofteh *

Faculty of Computer Science and Engineering, Shahid Beheshti University, Tehran, Iran

ARTICLE INFO

Communicated by Rodrigo Capobianco Guido

Keywords:
eXplainable AI
Auditory Filter Models
IIR filter
SincNet
Speaker Identification
Deep Learning

ABSTRACT

Learning interpretable filters in Convolutional Neural Networks (CNNs) is an approach that helps to build models with better generalization ability. Interpretable filters can reveal some hidden aspects of the task and help to improve the model. One of the most successful approaches in the field of the speech processing is *SincNet*, where the model learns some band-pass filters in the first layer of a CNN with a raw waveform as its input. In this paper, similar to *SincNet*, some meaningful filters are proposed, which here are inspired by Infinite Impulse Response (IIR) filters. The proposed model uses a phase correction process to ensure that phase linearity is satisfied. The effective length of the truncated IIR filter is calculated based on the accumulated energy, and the effect of changing the filter size on the final results has been investigated. The proposed model is evaluated in the speaker identification task on the TIMIT and Librispeech datasets and compared with traditional CNNs and four interpretable kernel-based models. The experimental results show the superiority of the proposed model both in performance and convergence speed. Moreover, some patterns of the speech signal, which lead to uniquely identifying a speaker, are analyzed by examining the spectrum of the learned filters.

1. Introduction

With the advent and widespread use of Deep Neural Networks (DNNs), most research efforts focused on developing deep learning (DL) techniques with the best performance in different tasks, and the justification of the model's decision-making process were neglected. Recently, this black-box overlooked aspect of DL models has been addressed in the revived field of research referred to as eXplainable artificial intelligence (XAI). XAI can help to discover the strengths and weaknesses of the models at hand. Moreover, XAI can reveal paths for future research and help develop more reliable and efficient machine learning (ML) solutions [1].

There are some worthy efforts to expose the black-box nature of deep models in the speech processing field of research. Some of such attempts are as follows. Learning interpretable representations by using relevance weighting scheme [2] or Transformer Variational AutoEncoders [3], exploration of SHapley Additive exPlanations (SHAP) to explain the behavior of DNNs in spoofing and speech deep-fake detection tasks [1], and to help Convolutional Neural Networks (CNNs) to discover more meaningful filters [4].

Learning specific filter types can be derived from the functional level description of the operation of the cochlea in the inner ear which acts as a frequency analyzer. Different auditory filter models have been proposed to simulate the operation that occurred in cochlea. Simple resonance, rounded exponential, rectangular, gammatone, Gaussian, and filter cascade are the main models used for this purpose [5]. This reference enumerates some good properties for the filters, as auditory filter modeling, which are derived from physiological and psychological experiments. Some of the most important ones are as follows: (1) The filters should have a simple description with bandwidth control ability. (2) It is better that the filter shape be asymmetric because experimental data show that the filter skirt on the high frequency side is usually steeper than the other side. (3) Easy implementation of filters is another property essential for developing machine hearing applications. (4) A good digital filter can be made by describing it in terms of poles and zeros and (5) the filter must maintain phase characteristics [5].

SincNet [4] is one of the well-known first efforts in this context, where some rectangular band-pass filters were learned in the first convolutional layer. Intuition behind this was that the first layer is the critical part of a CNN architecture because it interacts with high-dimensional inputs and is susceptible to the vanishing gradient problem in DL models [4]. The *SincNet* achieves better results compared with

* Corresponding author.
E-mail address: y_shekofteh@sbu.ac.ir (Y. Shekofteh).

https://doi.org/10.1016/j.neucom.2023.126767
Received 2 December 2022; Received in revised form 9 July 2023; Accepted 11 September 2023
Available online 13 September 2023

conventional methods in Speaker Identification (SID) and verification tasks. In addition to improving the performance, it learns such an efficient representation, which is readable by humans and reveals some known important speaker characteristics, such as pitch and formants.

Inspired by the *SincNet* and auditory filter models, this paper proposes another model for learning interpretable filterbanks in the first convolutional layer. This technique suggests learning a truncated version of a second-order Infinite Impulse Response (IIR) filter, which leads to faster training and better performance. Similar to *SincNet*, such a filter can be determined by only two parameters, center frequency and bandwidth. In this way, the model parameters are reduced compared with traditional CNNs. Since the IIR filters can have non-linear phases, a Forward-Backward Filtering (FBF) method is used to ensure that the designed filter has a linear phase and hence no information is lost because of the phase distortions.

The main properties of the proposed model named *IIRI-Net* are as follows: (1) the convergence speed of the model is increased by learning a filter with only two parameters. Fewer parameters impose a bias on the model that increases model generalization, too. (2) Since the original filter is derived from an IIR filter and recurrent neural networks can be viewed as a non-linear version of IIR filters [6], the learned filter can be regarded as a recurrent unit with a linear activation function and zero bias. This gives the learned filter more learning ability. (3) The IIR filter almost has all essential characteristics of a good auditory filter model mentioned above. (4) The last and most important property of the model is its interpretability. Learned filters clearly show the most critical frequency ranges and can reveal some interesting characteristics of the model and data.

The remainder of this paper is organized as follows. In Section 2, some main related works are introduced. The proposed method is explained in Section 3. Experimental results and conclusions are described in Sections 4 and 5, respectively.

2. Related works

As mentioned in the previous section, the main idea of learning meaningful filters in the first layer of a CNN originates from *SincNet* [4]. This architecture utilizes direct learning from raw speech signals, which completely avoids the need for any feature extraction step and allows for more customized low-level representations. According to the findings in reference [7], DNNs can learn a set of band-pass filters directly from the raw signal in the time domain. The study presented in [8] demonstrated that models with raw signal inputs can achieve performance comparable to log-*Mel* trained neural networks by employing a sophisticated acoustic model. This technique was also analyzed in [9] to understand how speech information is modeled between the first two convolution layers. In this section, *SincNet* is introduced in more details and then some extensions of this method are briefly described. A standard convolutional layer can be regarded as a set of time domain convolution operations. Suppose that $x[n]$ is the raw input signal and $h[n]$ is a filter with length L, which is learned in a convolutional layer. The filtered signal, $y[n]$, which is the convolution of $h[n]$ and $x[n]$, is obtained by the Equation (1), which is written based on the commutative property of convolution operator.

$$y[n] = x[n]*h[n] = \sum_{l=0}^{L-1} x[l] \,.\, h[n-l] = \sum_{l=0}^{L-1} x[n-l] \,.\, h[l] \tag{1}$$

The commutative property of the convolution operator is useful for mathematical proofs, but is not typically important for software implementations. Therefore, most neural networks libraries, such as pyTorch, which we used in our implementations, use the cross-correlation operator. This operator is similar to convolution, but the kernel is not flipped. As a result, the output $y[n]$ is calculated based on the equation $\sum_{l=0}^{L-1} x[n+l].h[l]$, and the algorithm learns a flipped kernel compared to an algorithm with flipping. In the context of ML, the obtained mirrored filters do not affect training [10]. However, because pyTorch does not use a full-correlation operator, several columns of the input may be lost, and the output size will be smaller than the input.

Table 1
Impulse response formulas of the baseline kernel-based models.

Model	Impulse response formula	
SincNet	$h[n] = 2B\text{sinc}(Bn)\cos(2\pi f_c n)$	(4)
Sinc^{2}Net	$h[n] = A\text{sinc}^2(Bn)\cos(2\pi f_c n)$	(5)
GammaNet	$h[n] = An^{N-1}\exp(-2\pi Bn)\cos(2\pi f_c n)$	(6)
GaussNet	$h[n] = A\exp(-n^2/2\sigma^2)\cos(2\pi f_c n), \sigma = \sqrt{(\log 2)/2\pi B}$	(7)

The idea of the *SincNet* is that the standard filter, $h[n]$, can be replaced with a parameterized *sinc* function representing a rectangular band-pass filter. By applying this restriction, the learned parameters of each filter are reduced from L to only two low and high cutoff frequencies. Moreover, the learned filters will be more meaningful. Equations (2) and (3) show the time and frequency domain formula of the considered filter.

$$h[n, f_1, f_2] = 2f_2 sinc(2\pi f_2 n) - 2f_1 sinc(2\pi f_1 n) \tag{2}$$

$$H(f, f_1, f_2) = \Pi\left(\frac{f}{2f_2}\right) - \Pi\left(\frac{f}{2f_1}\right) \tag{3}$$

where $sinc(x) = \sin(x)/x$, f_1 and f_2 are the learned low and high cutoff frequencies, respectively, and Π represents a rectangular function in the magnitude frequency domain.

In [11], three new filter types are introduced by converting the sum of sinusoids into a product. In this way, the impulse response can be interpreted as the product of a baseband kernel, modulating the carrier. The kernel function for *SincNet* is a *sinc* function. The paper examines three other kernels, including *squared sinc*, *gammatone*, and *Gaussian*, which give rise to the structures called *Sinc^{2}Net*, *GammaNet*, and *GaussNet*. The performance of the proposed kernels was examined on phoneme recognition task, and its superiority has been shown compared with the *SincNet*. Since we use these structures as our baseline models for comparison, the impulse response formulas are shown in Table 1. Based on the studies, these models have not been evaluated for SID application. In these equations, $B = f_2 - f_1$ and $f_c = (f_1 + f_2)/2$ are the bandwidth and center frequency of the filter, respectively. Parameter N in *GammaNet* is the order of the filter. Since the typical order of four correlates well with cochlea filters, $N = 4$ is used in the experiments.

PF-Net [12] introduced another type of filter with more learnable parameters. The method forced the model to learn some deformation points in the frequency domain at different low and high frequencies and hence constructs a filter made of consecutive line segments. This method achieved a lower classification error rate compared to *SincNet* in the SID task. Reference [13] extends the parameterized filters of the *SincNet* to complex-valued analytic filters to enable perfect synthesis in the task of the end-to-end speech separation. The authors show that the resulting filterbanks are more interpretable and perform equally well or better than their real-valued counterparts for speech separation in clean or noisy conditions.

In [14] a fully learnable architecture that can be used as an alternative for *Mel*-filterbanks is introduced. The model parameterizes a complex-valued filtering layer with Gabor filters and learns all operations of audio feature extraction, including filtering, pooling and, compression or normalization. The model is evaluated using different classification problems and either outperforms or matches the accuracy of other front-ends. CGCNN [15] combines the complex Gabor filter with complex-valued DNNs to replace standard CNN kernels and, in this way, entirely takes advantage of its optimal time–frequency resolution and complex domain. The experiments in the phoneme recognition task show relatively similar results compared to the *SincNet*. Parzen convolutional block [16] was another use of a specific filter type that decomposed the raw signal into some frequency sub-bands and then made a high-dimensional structured space. The method was evaluated

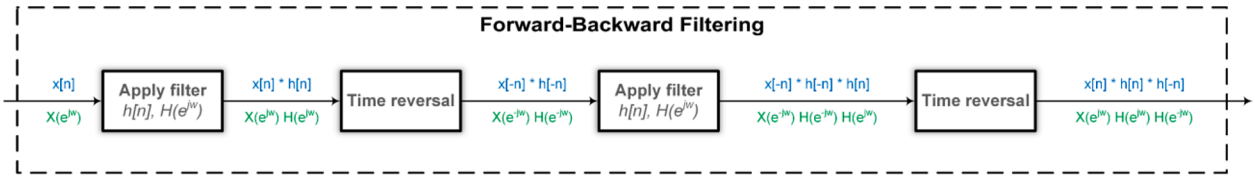


Fig. 1. The operations involved in obtaining a phase-corrected signal after applying a filter using the FBF technique. The time-domain operations are depicted in blue and the frequency-domain operations are shown in green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

in the speech recognition task and showed that the model outperforms other feed-forward models.

The use of specialized audio front-ends has been widely used in different speech processing tasks. The performance of the previously mentioned methods is evaluated in speaker identification and verification [4,12,14], phoneme recognition [11,15], speech separation [13], and different classification problems [14]. Some other applications that have used these models are as follows. References [17,18] use the *SincNet* in speaker diarization task. To reduce power and memory consumption, reference [19] eliminates preprocessing and domain transfer steps and reduces model parameters by using the raw audio as the input of the first *SincNet* layer of its proposed architecture for keyword spotting task. The medical domain has also used these models in various applications such as neurodegenerative-related disorder classification [20] and emotion recognition [21–24].

3. Materials and methods

In this section, the proposed procedure for learning an *IIR Inspired (IIRI) filter* in the first layer of a CNN-based model, *IIRI-Net*, is described. A general second-order all-pole filter can be represented by the following rational system function.

$$H(Z) = \frac{1}{(1 - z_1 Z^{-1})(1 - z_2 Z^{-1})} \tag{8}$$

where z_1 and z_2 determine its poles and two zeros are located in $Z = 0$. Given the causality assumption of the system, the impulse response, $h[n]$, must be a right-sided sequence, so the Region Of Convergence (ROC) of $H(Z)$ must be outside the outermost pole. By performing partial fraction expansion and applying inverse Z-transform, its impulse response corresponds to

$$h[n] = \left(\frac{z_1}{z_1 - z_2} z_1^n + \frac{z_2}{z_2 - z_1} z_2^n\right) u[n] \tag{9}$$

where $u[n]$ is a discrete-time unit step signal that is 1 for $n \geq 0$ and 0 for $n < 0$. If the poles are considered complex conjugates of each other, then a real-valued multiplier coefficients second-order IIR filter will be obtained. In this setting, the poles can be regarded as $z_1 = e^{(-\sigma + j\omega_0)}$ and $z_2 = z_1^*$. The necessary and sufficient condition of filter stability is satisfied by constraining σ to be positive, which forces the poles to remain inside the unit circle. In this situation, it can be shown that equation $z_1/(z_1 - z_2) = e^{j\omega_0}/(2j\sin\omega_0)$ holds true. By replacing this equivalence and its conjugate in Equation (9), and after some simplifications, the impulse response can be expressed as Equation (10). The frequency response of such an impulse response is a stable resonator, where ω_0 is the center frequency and 2σ is the bandwidth.

$$h[n] = \frac{sin((n+1)\omega_0)}{sin\omega_0} e^{-n\sigma} u[n] \tag{10}$$

Preserving the input signal's phase relationships is crucial in almost all speech processing applications. Therefore, it is essential to ensure that the resulting filter has a linear phase, especially in the band-pass region. Forward-Backward Filtering (FBF) [25] is a technique that forces this constraint on the resulting filter. The overall procedure of this technique is as follows: Apply the causal filter to the input signal, $x[n]$, forward in time, and then apply a second anti-causal filter backward on the filtered signal. In this way, the FBF squares the frequency response magnitude and zeros the phase in the frequency domain. Equations (11) and (12) show the final output of the FBF method in the time and frequency domains, respectively.

$$y[n] = h[n]*h[-n]*x[n] \tag{11}$$

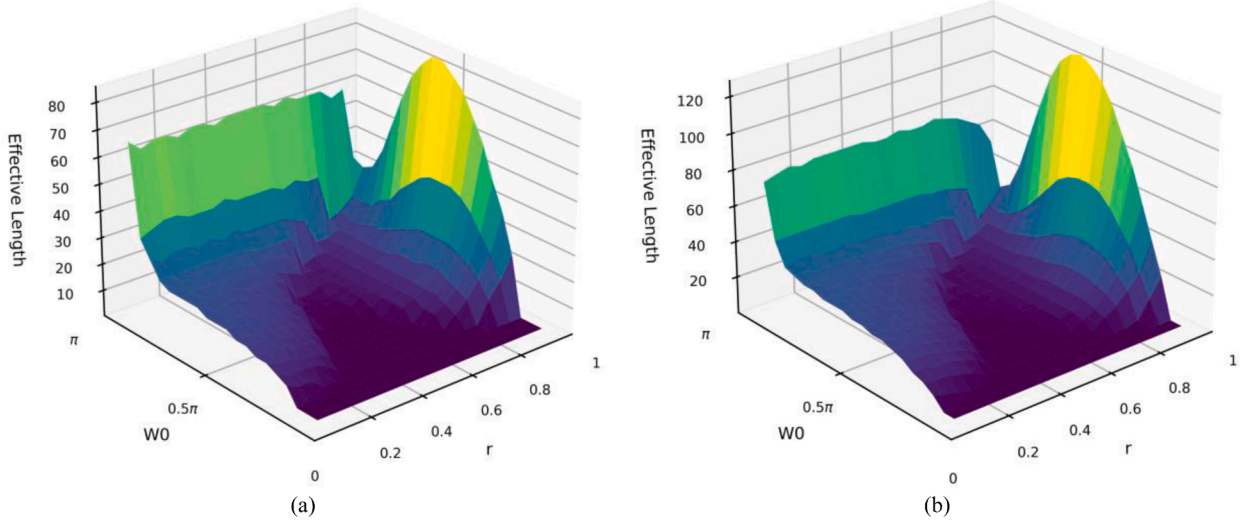


Fig. 2. Effective length (EL) for different pole locations for the proposed second-order all-pole filter for percentage (a) $P = 70\%$, (b) $P = 80\%$ of the impulse response energy.

Table 2
Maximum effective length (EL) of the proposed impulse response for different energy percentages.

Percentage of the impulse response energy	70%	80%	90%	95%	99%
Maximum Effective Length	84	126	250	488	2044

$$Y\left(e^{j\omega}\right) = X(e^{j\omega})\left|H(e^{j\omega})\right|^2 \tag{12}$$

The resulting new impulse response, $h'[n] = h[n]*h[-n]$, has infinite length, which is damped by the term of $e^{-n\sigma}$. Fig. 1 illustrates the operations performed on the input speech signal, $x[n]$, during the FBF procedure for phase correction while applying a filter. The figure shows both the time-domain and frequency-domain operations involved in the process.

The Effective Length (EL) of the impulse response of such a stable recursive filter can be calculated based on the accumulated energy [26]. Therefore, without losing extra information, $h'[n]$ can be truncated to a pre-defined length, and hence, it can be used as a FIR filter, which can be learned in the first convolutional layer. The algorithm for computing the EL of a general recursive filter is explained in reference [26]. In this manner, the EL of the impulse response is defined as the smallest non-negative integer time-index, L, for which at least $P\%$ of the total energy of the impulse response is attained (see Equation (13)).

$$\sum_{n=0}^{L} h'^2[n] \geq \frac{P}{100}\sum_{n=0}^{N_{max}} h'^2[n] \tag{13}$$

By using this procedure and estimating the total energy of $h'[n]$ as the sum of squared values of the first $N_{max} = 10^4$ samples, the EL for different values of magnitude ($0 < r < 1$) and phase ($0 < w_0 < \pi$) of the filter is calculated. In this case, the considered value for N_{max} is sufficiently large to get a reasonable estimate of the total energy. Fig. 2 depicts the surface of the EL for different pole locations for percentages $P = 70\%$ and 80% of the impulse response energy.

As can be seen, as the poles approach the unit circle ($r \rightarrow 1$) and locate in higher frequencies ($\omega_0 \rightarrow \pi$), the EL increases, which means more samples are needed to achieve the desired energy value. Table 2 shows the maximum EL to have different percentages of energy for the proposed impulse response.

It is obvious that by truncating the impulse response to the EL, we will not have an ideal IIR filter, and some undesirable ripples and overshoots will appear in the frequency response. The experimental results show that considering a reasonable filter length (i.e. near the maximum EL computed for percentage $P = 70\%$ or 80%), these effects can be ignored. Moreover, windowing has been used to smooth some discontinuities at the ends of $h'[n]$. Here, the popular *Hamming* window is used.

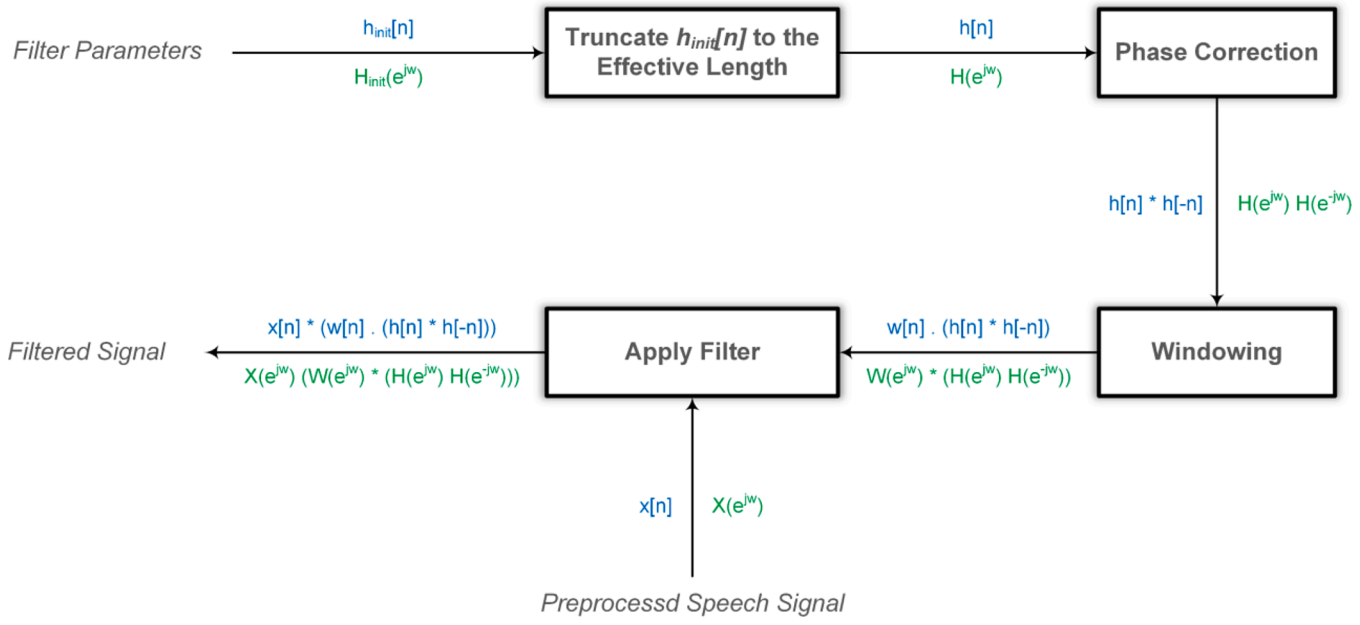


Fig. 3. The overall procedure for applying a filter to the input speech signal in the first convolutional layer.

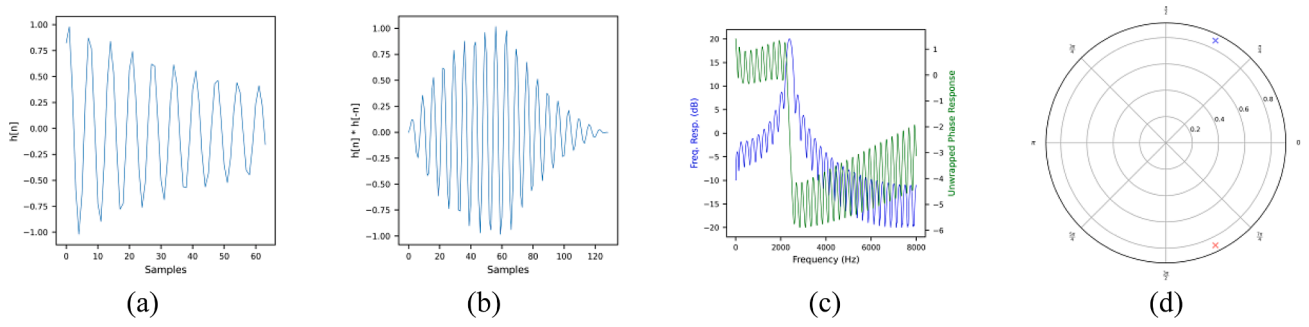


Fig. 4. (a) Impulse response, (b) linear phase impulse response, (c) frequency and phase response and (d) pole-zero plot of a learned filter with length of 129.

Table 3
Detailed configuration of the model architecture.

Model Structure		Layer Type	#Neurons	#Filters	Filter Length	Activation Type	#Parameters For *standard* model	#Parameters For *IIRI-Net* model
Input		Speech waveform	3200	–	–	–	0	0
CNN	CNN layer 1	Interpretable Conv. Filters	–	80	L	Leaky ReLU	278,360	257,880
		Max Pooling	–	–	3			
		Layer Normalization	–	–	–			
	CNN layer 2	Standard Conv. Filters	–	60	5	Leaky ReLU		
		Max Pooling	–	–	3			
		Layer Normalization	–	–	–			
	CNN layer 3	Standard Conv. Filters	–	60	5	Leaky ReLU		
		Max Pooling	–	–	3			
		Layer Normalization	–	–	–			
DNN	DNN layer 1	Fully Connected	2048	–	–	Leaky ReLU	21,580,328	21,580,328
		Batch Normalization	–	–	–			
	DNN layer 2	Fully Connected	2048	–	–	Leaky ReLU		
		Batch Normalization	–	–	–			
	DNN layer 3	Fully Connected	2048	–	–	Leaky ReLU		
		Batch Normalization	–	–	–			
Output		SoftMax	#Speakers	–	–	–	1,293,390 (For TIMIT)	1,293,390 (For TIMIT)

$$w[n] = 0.54 - 0.46 \,.\, cos\left(\frac{2\pi n}{L}\right) \tag{14}$$

where L is the filter length. By multiplying $h'[n]$ by the *Hamming* window, the desired final filter, $h''[n]$, can be represented by Equation (15).

$$h''[n] = w[n].\ (h[n]*h[-n]) \tag{15}$$

Fig. 3 illustrates the complete operations performed on the initial filter, $h_{init}[n]$, obtained from Equation (10), to obtain a windowed phase-corrected filter and apply it to the input speech signal. The figure presents both the time-domain and frequency-domain formulas involved in the process.

Fig. 4 shows $h[n], h''[n]$, frequency and unwrapped phase response, and the pole-zero plot of a filter of length 129 that is learned by the proposed *IIRI-Net*. The impact of applying FBF can be seen in the phase response, where it is linear in the band-pass regions. The pole-zero plot clearly shows the relationship between the frequency domain and Z-domain. If the gain factor of frequency response is ignored, then the pole-zero plot will contain enough information to define the frequency response. As can be seen, the presence of the pole near the unit circle causes a peak in the corresponding frequency in 2.8 kHz. This property offers the use of this representation in a more efficient way for analyzing the corresponding filter.

In this manner, the only parameters required for learning are ω_0 and σ, which somehow represent location of the poles of the reference IIR filter. The learned filter is a band-pass filter that amplifies the frequencies around the center frequency, ω_0, and attenuates other frequencies.

Similar to the *SincNet*, the center frequencies and bandwidths can be initialized with the center frequencies and bandwidths of the *Mel*-filterbanks, respectively. This causes more filters to be assigned to the lower part of the spectrum. In addition, like the *SincNet*, the gain of the filter is not a learning parameter because the weights learned in subsequent layers can increase or decrease the importance of each filter's output. All of the operations described above for obtaining the final desired filter, including convolution, forward–backward filtering, truncation, and windowing, are differentiable. Hence, gradient-based optimization methods can be used to optimize the new layer along with the standard layers of a CNN [4].

4. Experimental Setup, Results, and discussion

The proposed method is evaluated on the TIMIT [27] and Librispeech [28] for SID task. In this section, the experimental setup, including the details of the datasets used, model architecture and various experiments are explained.

4.1. Datasets

TIMIT contains recordings of 630 speakers of eight major dialects of American English, each reading ten phonetically rich sentences in which a 16-bit, 16 kHz speech waveform file is included for each utterance. Non-speech intervals at the beginning and end of each sentence are removed. There are two calibration sentences designed to allow cross-speaker comparisons, which were removed in this experiment [27]. Five sentences for each speaker are used for training, two for validation, and one for testing, which leads to a 630-class classification problem.

For the second dataset, we use LibriSpeech, a corpus of read English speech recordings. We specifically utilize the train-clean-360 subset of the corpus, which comprises 363.6 h of 16 kHz speech data from 921 distinct speakers which leads to a 921-class classification problem. To simulate challenging but realistic conditions, we use minimal training data of 12–15 s for each speaker and short test and validation segments ranging from 2 to 6 s. The training, validation and test segments are randomly selected.

4.2. Model architecture

Since the goal is to compare the proposed front-end layer with *SincNet*, the architecture used here is similar to the model used in [4]. The detailed configuration of the *SincNet* architecture is presented in Table 3.

Each speech sentence is split into chunks of 200 ms with 10 ms overlap, which makes the 3200 samples input length. In the first layer, 80 interpretable filters of length L samples are learned. Two standard convolutional layers with 60 filters of length 5 are placed after this layer. Max pooling, layer normalization [29], and Leaky-ReLU activation [30] are applied in order to all 3 convolutional layers without any dropout. Then 3 fully-connected layers with 2048 neurons and batch normalization [31] are used. Leaky-ReLU is the activation function used in all hidden layers, and the final classification result is obtained by applying a SoftMax classifier. The sentence-level classification is obtained by averaging the frame predictions and voting for the speaker, which maximizes the average posterior. The Negative Log Likelihood (NLL) loss, RMSprop optimizer with a learning rate of 0.001, smoothing constant 0.95, $\varepsilon = 10^{-7}$, and mini-batches of size 128 are used for training. Training process is done for 1000 and 500 epochs for the TIMIT and Librispeech datasets, respectively.

The last two columns of Table 3 report the number of trainable

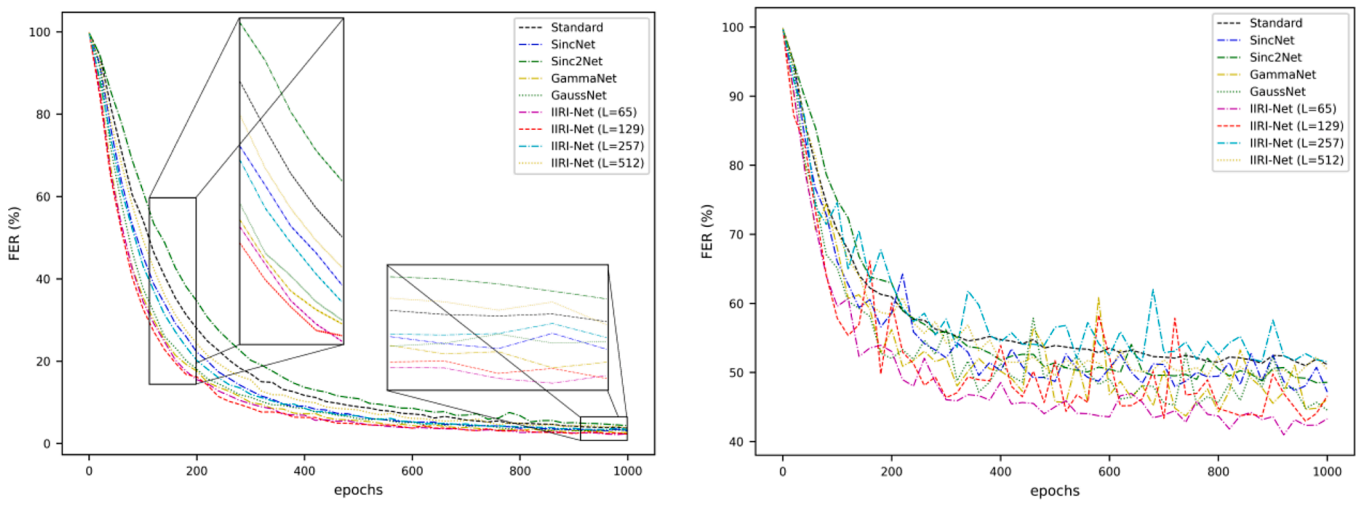


Fig. 5. Frame Error Rate for the (a) training and (b) validation data of the TIMIT dataset for *standard* and *kernel-based* CNN and *IIRI-Net* with different lengths over training epochs.

Table 4
Frame Error Rate (FER%) and Classification Error Rate (CER%) for TIMIT speaker identification task.

	Validation		Test	
	FER (%)	CER (%)	FER (%)	CER (%)
Standard	51.5 ± 0.8	1.5 ± 0.2	50.6 ± 0.6	1.9 ± 0.5
SincNet	47.9 ± 0.8	0.9 ± 0.2	48.4 ± 0.8	1.4 ± 0.4
Sinc^{2}Net	49.8 ± 1.5	0.8 ± 0.1	48.6 ± 1.1	1.1 ± 0.1
GammaNet	44.7 ± 1.8	1.1 ± 0.4	44.8 ± 1.3	1.1 ± 0.2
GaussNet	45.8 ± 1.8	0.7 ± 0.4	46.1 ± 1.0	1.4 ± 0.8
IIRI-Net (L = 65)	**43.1 ± 1.2**	0.6 ± 0.1	**43.5 ± 0.8**	1.1 ± 0.7
IIRI-Net (L = 129)	44.1 ± 1.3	0.7 ± 0.1	43.9 ± 1.3	**0.8 ± 0.2**
IIRI-Net (L = 257)	51.0 ± 1.0	0.8 ± 0.1	50.8 ± 1.0	1.3 ± 0.4
IIRI-Net (L = 513)	49.7 ± 0.4	**0.5 ± 0.1**	49.8 ± 0.4	0.8 ± 0.4

parameters in each part of the model structure for comparison purposes. Using meaningful filters in the first convolutional layer leads to a reduction of 20,480 model parameters. Although this value is relatively small compared to the number of parameters in the subsequent layers, it still plays a crucial role in reducing the search space and facilitating the optimization process. Moreover, in some low-resource applications [32], a filter selection mechanism is employed to identify the effective filters that are truly beneficial to support the classification in a CNN architecture. The new representation of learned filters in terms of center frequency and bandwidth helps such approaches to prune learned filterbanks more effectively.

4.3. Experiments

In the following, the experiments conducted to evaluate the convergence speed and the impact of the filter lengths on the performance metrics is presented. To evaluate the effect of the filter size in the first layer, different sizes $L = 65, 129, 257$ and 513 are considered for IIRI filters and compared with four *kernel-based* CNN models, *SincNet*, *Sinc^{2}Net*, *GammaNet*, *GaussNet*, and *standard* CNN with $L = 257$ as the baseline models. To speed up convergence, we initialize the center frequencies and bandwidths of all kernel-based and *IIRI-Net* models using the center frequencies and bandwidths of the *Mel*-filterbanks.

Two criteria, Frame Error Rate (FER) and Classification Error Rate (CER), are utilized to evaluate the models. CER is calculated as the number of incorrectly classified speech signals divided by the total number of speech signals, while FER is the corresponding proportion calculated at the frame level. Fig. 5 illustrates the changes in FER of the models over various epochs during a single run of the training process, in both the training and validation datasets. It should be noted that the experiments were conducted five times for each model, and the execution process shown in this figure corresponds to the run for which the CER on the test data was the median. It can be observed that in general, *IIRI-Net (L = 65, L = 129)* converges slightly faster than *standard* CNN and *kernel-based* models. This may be attributed to the implicit recursive nature inherent in this model, which provides it with a higher learning capacity.

Table 4 compares the average and standard deviation of FER and CER obtained from different models across five runs for the TIMIT dataset. For each run, the final model is selected based on the best CER

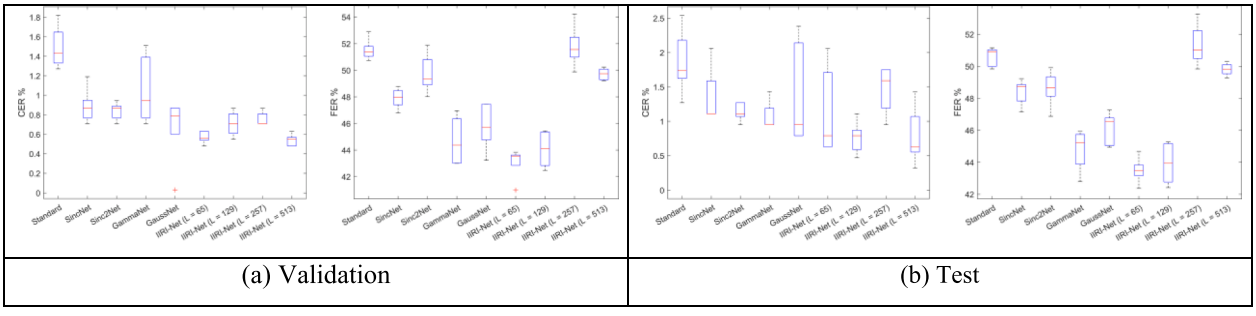


Fig. 6. Distribution statistics on the FER % for TIMIT speaker identification task on the (a) validation and (b) test data.

Table 5
Frame Error Rate (FER%) and Classification Error Rate (CER%) for Librispeech speaker identification task.

	Validation		Test	
	FER (%)	CER (%)	FER (%)	CER (%)
Standard	47.9 ± 1.8	1.4 ± 0.2	47.8 ± 1.8	2.3 ± 0.3
SincNet	48.4 ± 0.7	1.3 ± 0.1	48.4 ± 0.7	1.8 ± 0.4
GammaNet	47.8 ± 1.0	1.2 ± 0.2	47.9 ± 0.6	1.5 ± 0.3
IIRI-Net (L = 129)	**45.4 ± 0.7**	**1.1 ± 0.2**	**45.0 ± 0.6**	**1.3 ± 0.3**

on the validation data during the training process. The first- and second-best evaluation values are bolded and underlined, respectively. The proposed *IIRI-Net* in all cases outperforms *standard* and *kernel-based* models based on the aforementioned evaluation criteria. *IIRI-Net* model with filter length $L = 65$ achieves better FER while *IIRI-Net* with $L = 129$ has the best CER over the test data. Although we expect better performance by increasing the filter length, which corresponds to increasing the impulse response energy and losing less information due to truncating the respective IIR filter, it is observed that the best results are obtained using filters with smaller lengths. This suggests that holding all information is not necessary to achieve optimal performance. Instead, the use of smaller filters to construct appropriate band-pass filters can be sufficient and improve the generalization of the model.

The corresponding spread of the results presented in Table 4 is depicted in Fig. 6. Among all cases, the *GaussNet* has relatively higher variability compared to the other models. The poor performance of the *IIRI-Net* models with longer lengths in terms of FER in the validation and test data is evident in these plots. However, contrary to the poor performance of these models based on this evaluation criterion, they perform well in terms of CER. *IIRI-Net (L = 129)* shows a stable behavior with a symmetric appearance. Among the 45 evaluated models, the minimum CER and FER results are obtained with *IIRI-Net for L = 513* and *IIRI-Net for L = 65*, respectively.

The weights of the models trained on the TIMIT dataset were used as the initial point for training a model for the Librispeech SID task. The *Standard*, *SincNet* and *GammaNet* models are exploited here as baseline models and *IIRI-Net (L = 129)* is chosen as a representative of the family of *IIRI-Net* filters. Table 5 presents the FER and CER results for each model over five runs. In all cases, *IIRI-Net (L = 129)* achieved the best results. While *GammaNet* has the second-best overall performance, the *standard* CNN architecture achieved a better FER over the test data compared to *SincNet* and *GammaNet* models. Since DNN-based models are sensitive to the size of training set [33], this may be due the ability of the model to learn more complex filters with sufficient training data.

The histogram of the center frequencies of the filters of the examined models is depicted in Fig. 7 for the TIMIT dataset. As expected, since most of the speech frequency information is spread at low frequencies, more filters operate in this region. In other words, the learned models, consistent with perceptual scales inspired by the human auditory system, are more sensitive to these spectral components. It can be seen that all filters reduce the importance of the frequency range around *2 kHz* compared with *Mel*-filterbank, which means that this frequency interval has less impact on the performance of the SID. This frequency region corresponds to the first three formants which are encoded in *200 Hz ∼ 3 kHz* that has less importance in the SID. The models also give more importance to higher frequencies compared with *Mel*-Filterbanks. This finding is reported in [34] which analyses the importance of frequency

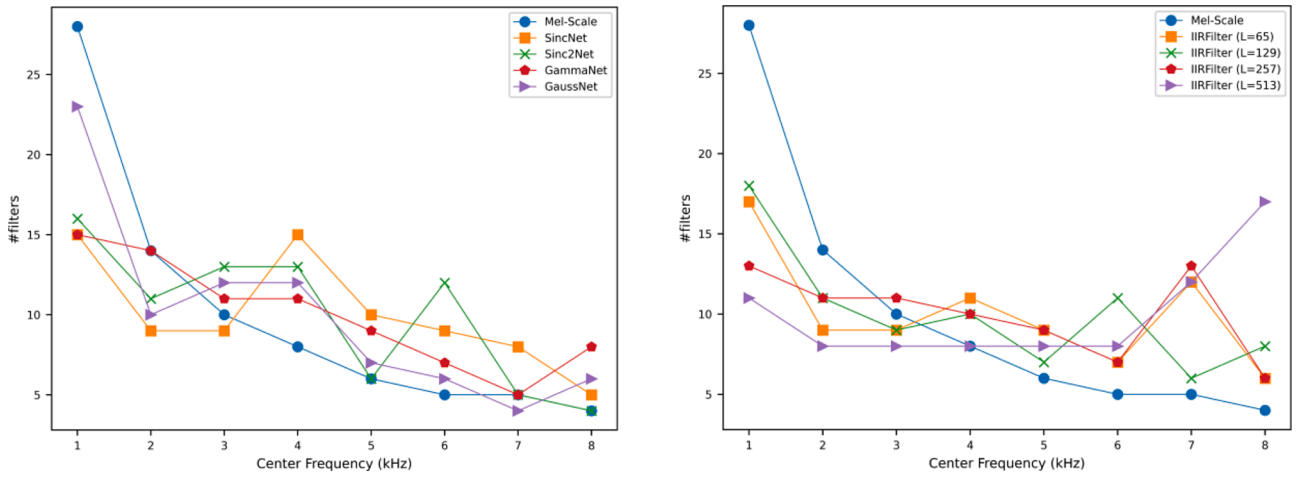


Fig. 7. Comparison of the distribution of center frequencies of learned filters by different models for the TIMIT dataset.

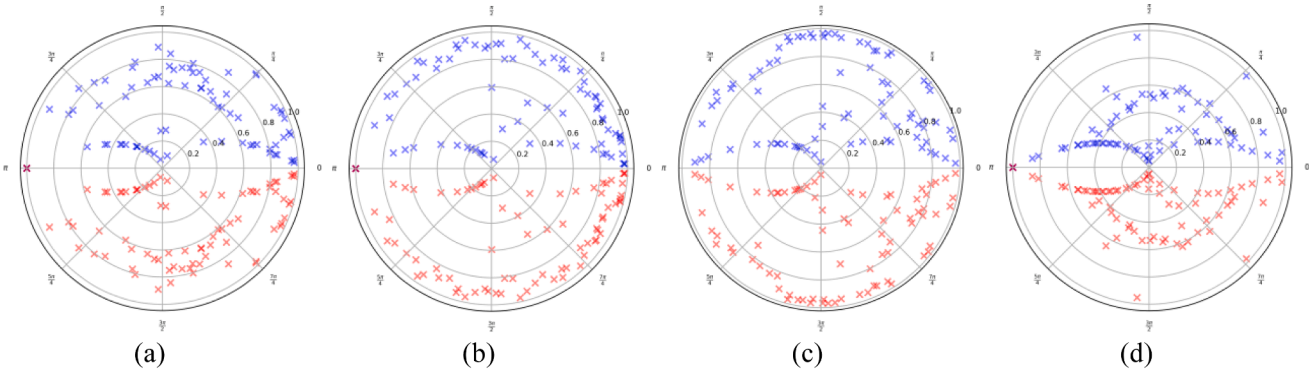


Fig. 8. The location of poles learned in *IIRI-Net* with different lengths, (a) $L = 65$, (b) $L = 129$, (c) $L = 257$, and (d) $L = 513$ for the TIMIT dataset.

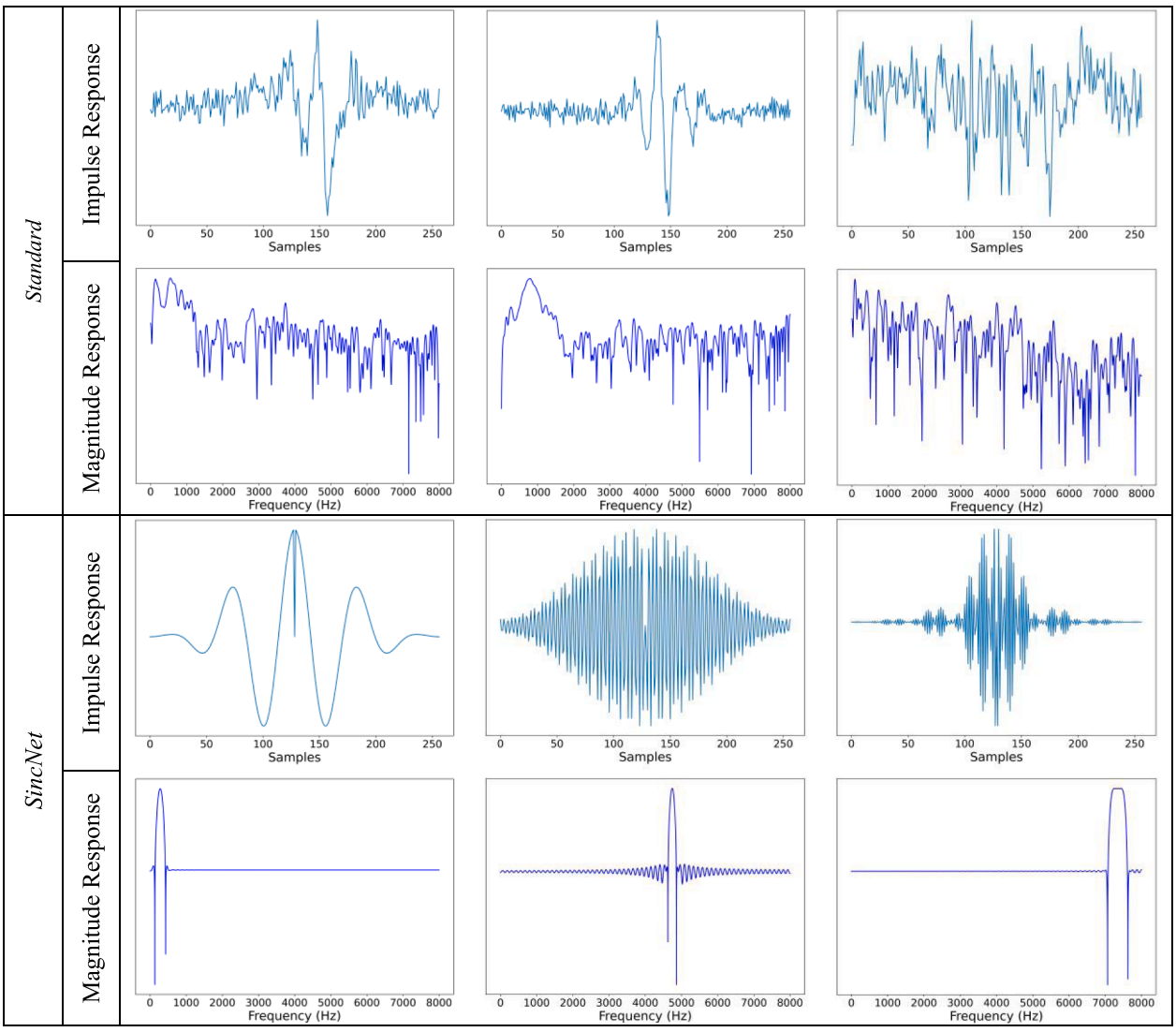


Fig. 9. Impulse response and magnitude response of three learned filters by different models (The magnitude response is plotted between 0 and 8 kHz).

bands from the view point of the speech production.

The above behavior of the IIRI filters can also be analyzed based on the location of the poles corresponding to different learned filters in the complex plane. Fig. 8 shows the poles locations of different learned filters with $L = 65, 129, 257$ and 513 for the TIMIT dataset. As can be seen, the spatial distribution of the poles is denser at lower frequencies. Moreover, as we approach high frequencies, the poles move further away from the unit circle especially in the filters with lower lengths. This implies the compatibility of the learned filters with the human auditory system, which is more sensitive to lower frequencies. On the other hand, the longer the filter length, the more regular the distribution of the poles, which indicates the effect of the total energy preserved by the longer filter length. But as can be seen in Table 4, the best results of FER are acquired using a filter with $L = 65$, and as the length of the filters increased, the results get worse. This property can be related to the model's effort to achieve higher accuracy against the loss of interpretability. Therefore, making a model interpretable can lead to a decrease in its accuracy.

Fig. 9 depicts three sample filters learned by different models examined in the experiments. This representation provides a good insight into what the network learns. As can be observed, the magnitude response of the filters learned from *standard* CNN looks noisy and can be considered to have a multi-band shape. In contrast, *SincNet*, *Sinc^{2}Net*, *GammaNet*, and *GaussNet* learned filters are all well-known *rectangular*, *triangular*, *gammatone* and *Gaussian* band-pass filters, respectively, in which their bandwidth and center frequency reveal the sensitivity of the network to some specific frequency region. This property is also valid for the filters learned by *IIRI-Net*. As the filter length of the *IIRI-Net* increases, the learned filters become more like the ideal IIR filters with fewer stop-band ripples.

5. Conclusions and future works

Directly processing the audio waveforms with DL models has shown notable success in different speech processing tasks. Even though these models have demonstrated high efficiency, similar to other DL methods, they suffer from not being interpretable. Based on the principles proposed for designing efficient filters derived from human auditory system, this paper introduced *IIRI-Net*, a CNN network with IIR-inspired filterbanks that are meaningful, implementable and controllable. Although the proposed filter type is not purely IIR, it has, to a certain extent, part of the benefits of the IIR filters. The CNN equipped with this filter type can converge faster than *standard* and *kernel-based* CNNs. The filters are determined by only two parameters, which offer an efficient

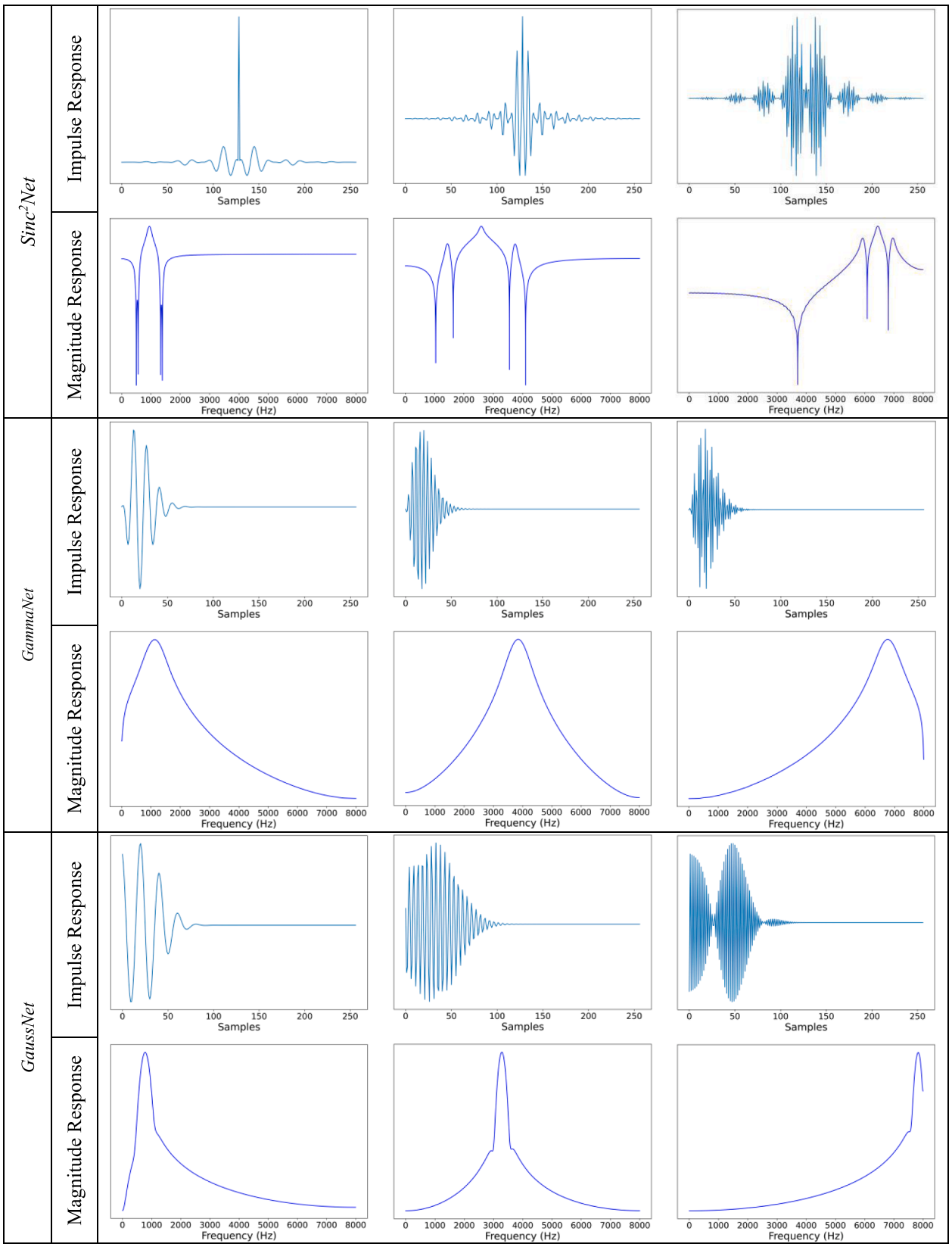


Fig. 9. (*continued*).

way of training CNN models while achieving better generalization. It can be believed that because of the general concept of the IIR filters that are used in some way in the proposed model, this model can achieve better performance in other speech or even signal processing tasks.

Three research trends can be considered to develop the presented method. (1) Taking advantage of the interpretable nature of the learned filters, the power, and memory consumption of the model can be reduced by taking into account the pole locations. The distribution of the

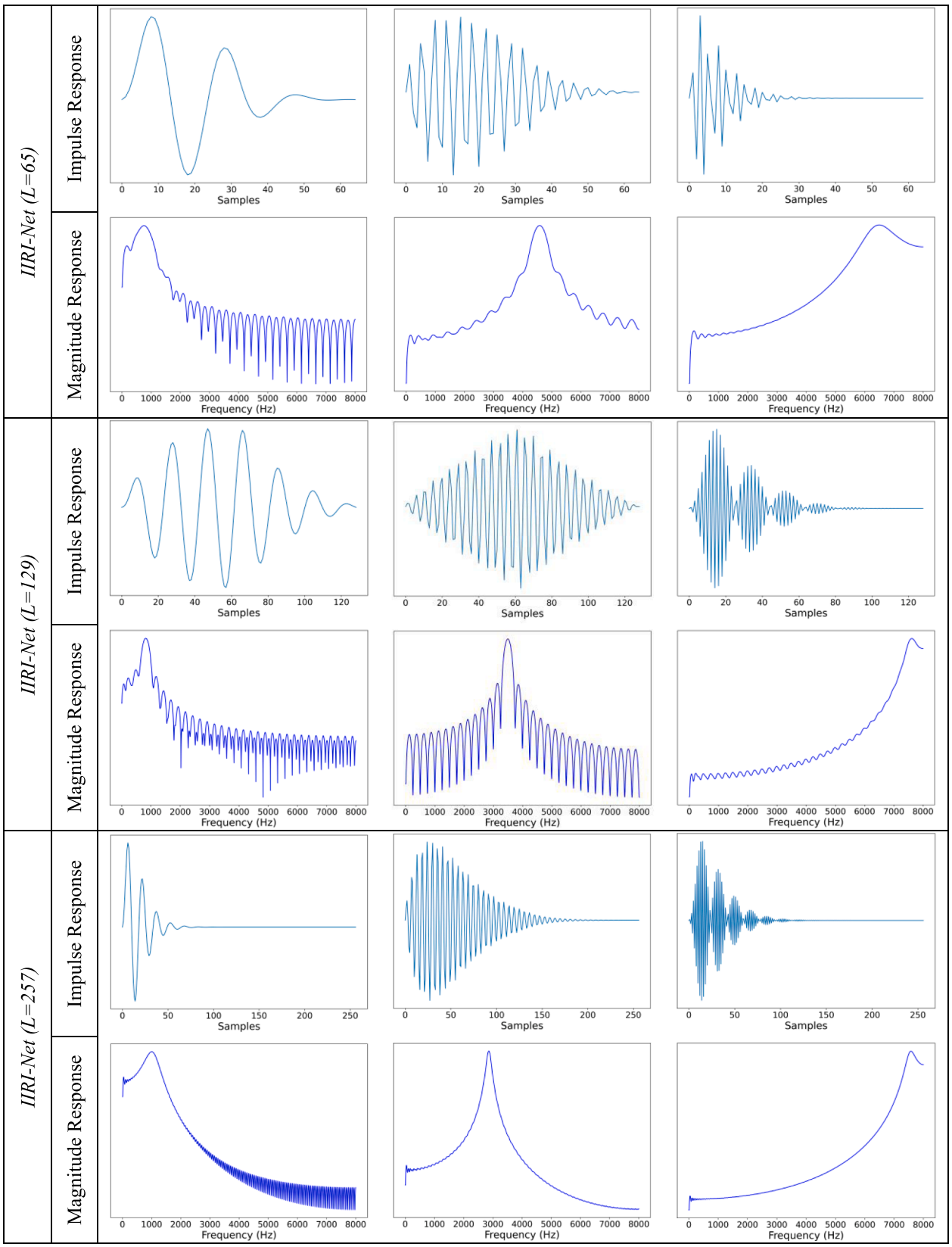


Fig. 9. (*continued*).

poles in the complex plane gives a good insight into the optimal number of filters required for the task at hand. Therefore, with the aim of power saving, which has a great impact in battery-powered speech-based applications, the future works can be concentrated on using this representation of learned filters, e.g., advanced clustering of the poles, for making lighter models. (2) The order of the learned filters can be increased by cascading two or more layers equipped with the proposed filter type. Cascaded filter banks reduce the total computational

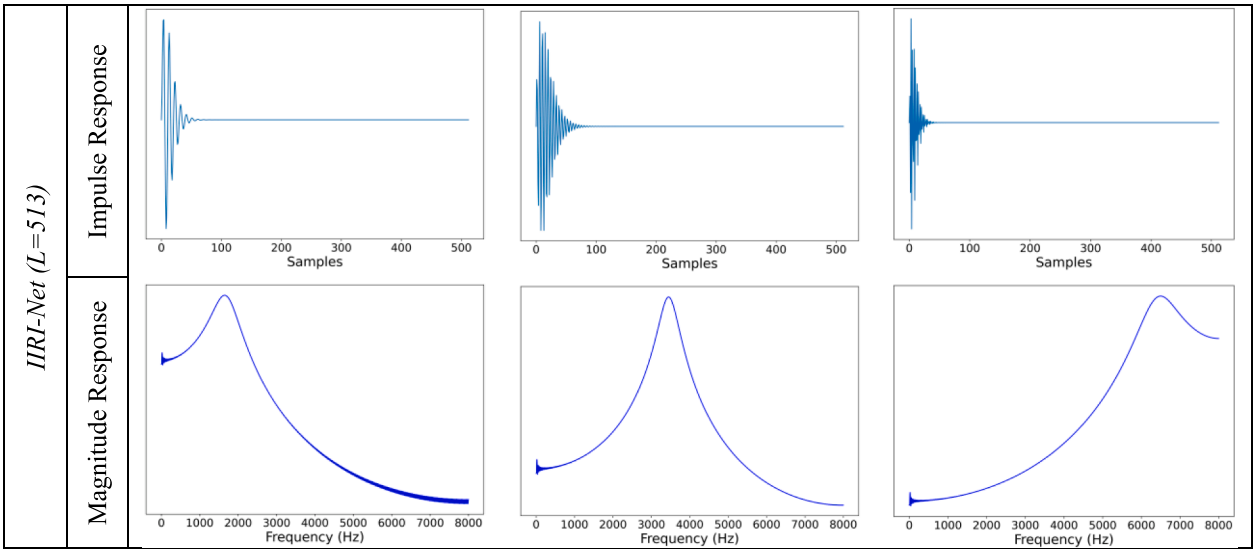


Fig. 9. (*continued*).

complexity of the model and make the model more interpretable. (3) The relation between an explainable model and biological processes occurring in the human auditory system can also be explored. Proper understanding the constraints of the human auditory system and applying them to the proposed model, the performance of the model can be improved. On the other hand, bio-inspired interpretations can be extracted from the learned model. Interpretable filters can have the capacity to reveal some bio-mechanisms. Based on the observations from resulted learned filters, the sensitivity of the auditory system filters can be changed according to the application. This property can be revealed by examining the learned filters by other speech processing applications like phone recognition and speech enhancement.

CRediT authorship contribution statement

Hossein Fayyazi: Conceptualization, Methodology, Software, Investigation, Formal analysis, Visualization, Writing – original draft. **Yasser Shekofteh:** Supervision, Conceptualization, Investigation, Validation, Writing – review & editing.

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.

Data availability

Data will be made available on request.

References

[1] W. Ge, et al., Explaining Deep Learning Models for Spoofing and Deepfake Detection With SHapley Additive ExPlanations. ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022.
[2] P. Agrawal, S. Ganapathy, Interpretable representation learning for speech and audio signals based on relevance weighting, IEEE/ACM Trans. Audio, Speech, Language Process. 28 (2020) 2823–2836.
[3] J. Jiang, et al., TrAnsformer VAe: A HierArchicAl Model for Structure-AwAre and InterpretAble Music RepresentAtion LeArning. ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020.
[4] M. Ravanelli, Y. Bengio, Speaker Recognition From Raw Waveform With Sincnet. 2018 IEEE Spoken Language Technology Workshop (SLT), IEEE, 2018.
[5] Lyon, R.F., *Human and machine hearing: extracting meaning from sound*. 2017: Cambridge University Press.
[6] Kuznetsov, B., J.D. Parker, and F. Esqueda. *Differentiable IIR filters for machine learning applications*. in *Proc. Int. Conf. Digital Audio Effects (eDAFx-20)*. 2020.
[7] Z. Tüske, et al. Acoustic modeling with deep neural networks using raw time signal for LVCSR. in Fifteenth annual conference of the international speech communication association. 2014. Citeseer.
[8] T. Sainath, et al., *Learning the speech front-end with raw waveform CLDNNs*. 2015.
[9] D. Palaz, R. Collobert, *Analysis of CNN-based speech recognition system using raw speech as input*. 2015, Idiap.
[10] I. Goodfellow, Y. Bengio, A. Courville, *Deep learning*. 2016: MIT press.
[11] E. Loweimi, P. Bell, S. Renals, On Learning Interpretable CNNs with Parametric Modulated Kernel-Based Filters, Interspeech. (2019).
[12] W. Li, et al., PF-Net: Personalized Filter for Speaker Recognition from Raw Waveform. Mobile Multimedia Communications: 15th EAI International Conference, MobiMedia 2022, Virtual Event, July 22–24, 2022, Proceedings, Springer, 2023.
[13] M. Pariente, et al., Filterbank Design for End-to-end Speech Separation, IEEE, 2020.
[14] N. Zeghidour, et al., *LEAF: A learnable frontend for audio classification*. arXiv preprint arXiv:2101.08596, 2021.
[15] P.-G. Noé, T. Parcollet, M. Morchid, Cgcnn: Complex Gabor Convolutional Neural Network on Raw Speech. ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020.
[16] D. Oglic, et al. *A Deep 2D Convolutional Network for Waveform-Based Speech Recognition*. in *INTERSPEECH*. 2020.
[17] H. Dubey, A. Sangwan, J.H. Hansen, Transfer Learning Using Raw Waveform Sincnet for Robust Speaker Diarization. ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019.
[18] A. Larcher, et al., Speaker Embeddings for Diarization of Broadcast Data in the Allies Challenge. ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2021.
[19] S. Mittermaier, et al., Small-footprint Keyword Spotting on Raw Audio Data With Sinc-convolutions. ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020.
[20] Y. Pan, et al., Acoustic Feature Extraction With Interpretable Deep Neural Network for Neurodegenerative Related Disorder Classification. Proceedings of Interspeech 2020, International Speech Communication Association (ISCA), 2020.
[21] J.M. Mayor-Torres, et al., Interpretable Sincnet-based Deep Learning for Emotion Recognition From Eeg Brain Activity. 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, 2021.
[22] C. Liu, J. Jin, I. Daly, S. Li, H. Sun, Y. Huang, X. Wang, A. Cichocki, SincNet-based hybrid neural network for motor imagery EEG decoding, IEEE Trans. Neural Syst. Rehabil. Eng. 30 (2022) 540–549.
[23] H. Zeng, Z. Wu, J. Zhang, C. Yang, H. Zhang, G. Dai, W. Kong, EEG emotion classification using an improved SincNet-based deep learning model, Brain Sci. 9 (11) (2019) 326.
[24] A. Anand, S. Negi, N. Narendra, Filters Know How You Feel: Explaining Intermediate Speech Emotion Classification Representations. 2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), IEEE, 2021.
[25] F. Gustafsson, Determining the initial states in forward-backward filtering, IEEE Trans. Signal Process. 44 (4) (1996) 988–992.
[26] T.I. Laakso, V. Valimaki, Energy-based effective length of the impulse response of a recursive filter, IEEE Trans. Instrum. Meas. 48 (1) (1999) 7–17.
[27] J.S. Garofolo, et al., *Darpa timit acoustic-phonetic continuous speech corpus CD-ROM {TIMIT}*. 1993.

[28] V. Panayotov, et al., Librispeech: an Asr Corpus Based on Public Domain Audio Books. 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, 2015.
[29] J. Lei Ba, J.R. Kiros, G.E. Hinton, *Layer normalization.* ArXiv e-prints, 2016: p. arXiv: 1607.06450.
[30] A.L. Maas, A.Y. Hannun, A.Y. Ng, Rectifier nonlinearities improve neural network acoustic models. Proc. icml. 2013. Atlanta, Georgia, USA, 2013.
[31] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift. International conference on machIne learnIng, PMLR, 2015.
[32] C. Alippi, S. Disabato, M. Roveri, Moving convolutional neural networks to embedded systems: the alexnet and VGG-16 case. 2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), IEEE, 2018.
[33] Yang, C.-H.H., et al., *A quantum kernel learning approach to acoustic modeling for spoken command recognition.* arXiv preprint arXiv:2211.01263, 2022.
[34] X. Lu, J. Dang, An investigation of dependencies between frequency components and speaker characteristics for text-independent speaker identification, Speech Comm. 50 (4) (2008) 312–322.

Hossein Fayyazi is a Ph.D. student in the Faculty of Computer Science and Engineering at Shahid Beheshti University (SBU), Tehran, Iran, where his research is primarily focused on developing interpretable models for speech processing applications such as speaker identification and spoofing detection techniques.

Yasser Shekofteh received his MS and Ph.D. in biomedical engineering from Amirkabir University of Technology (AUT), Tehran, Iran, in 2008 and 2013, respectively. He is currently an assistant professor in the faculty of Computer Science and Engineering at Shahid Beheshti University (SBU), Tehran, Iran. His research interests include speech processing, digital signal processing, pattern recognition, and artificial intelligence.