EURASIP Journal on Audio, Speech, and Music Processing

EMPIRICAL RESEARCH

Open Access

Trainable windows for SincNet architecture

Prashanth H C, Madhav Rao*, Dhanya Eledath and Ramasubramanian V

Abstract

SincNet architecture has shown significant benefits over traditional Convolutional Neural Networks (CNN), especially for speaker recognition applications. SincNet comprises parameterized Sinc functions as filters in the first layer followed by convolutional layers. Although SincNet is compact in nature and offers top-level understanding of the features extracted, the effect of window function used in SincNet is not thoroughly addressed yet. Hamming and Hann are popularly used as the default time-localized windows to reduce spectral leakage. Hence, a comprehensive investigation of 28 different windowing functions on SincNet architecture towards speaker recognition task using TIMIT dataset was performed in this work. Additionally, "trainable" window functions were configured with tunable parameters to characterize the performance. The paper benchmarks the effect of the time-localized windowing function in terms of the bandwidth, side-lobe suppression, and spectral leakage for the filter banks employed in the first layer of the SincNet architecture. Trainable Gaussian and Cosine-Sum functions exhibited relative improvement of 41.46% and 82.11% in the sentence level classification error rate over Hamming window when employed on SincNet architecture.

Keywords Trainable windows, SincNet architecture, Speaker recognition

1 Introduction

SincNet architecture derived from Convolutional Neural Network (CNN) is reported to yield better results for speaker recognition tasks [1, 2] and continuous decoding of speech signals [3]. Besides an advantage of reduced parameters in SincNet structure owing to the usage of band-pass type filter bank in the first layer, the extraction of the features is much more interpretable than the mix of filters learnt in CNN. SincNet allows to keep the model size low by defining two cutoff frequencies for each of the band-pass filters employed in the first layer of the CNN. Hence, all the kernel-based filter-bank offer high-level tunable parameters, defined by pair of cutoff frequencies during the network training phase. Since its inception by Ravanelli and Bengio [2, 4], few advances in SincNet architecture have been carried out. Sinc-layer followed by depth-wise separable-convolutions (DSConv) to reduce network parameters and subsequently achieve energy efficiency was attempted on speech commands [5] and for speaker verification [6]. Curricular based loss function was applied on SincNet architecture in [7] to improve the speaker recognition accuracy. Another work on leveraging the stride and window size to extract features on SincNet was attempted [5]. All the recent works have either been an attempt towards utilizing Sinc functional layers and integrate with other existing convolutional layers to improve accuracy or utilize the existing style of operations on Sinc function to further leverage the hyper-parameters. Few other work showcased the learnability of filters in CNNs towards phone recognition [8] on raw speech. An alternative to SincNet architecture based on the complex Gabor filter learning process was proposed in [9]. To the best of our knowledge, no study has been performed on evaluating the time-localized Sinc filters using a variety of windowing functions and establish optimal windowing function for speaker recognition task. A characterization of different windowing functions on SincNet architecture in terms of trainable parameters, validation loss, and sentence level classification error rate

*Correspondence:
Madhav Rao
mr@iiitb.ac.in
International Institute of Information Technology Bangalore, Bangalore, India

(CER) is not established yet, but is highly valuable in distinguishing the speech signals by different speakers.

This paper contributes in analyzing 28 time-localized windowing functions for the SincNet architecture towards speaker recognition task. The trainability of certain groups of windowing functions is also investigated to achieve the best performance in terms of sentence level CER. A trade-off between wide band-pass spectrum with reduced stop-band ripples and less spectral leakage over the parameterized Sinc filter bank is discussed. Individual time-localized Sinc filters' temporal and frequency response for the filter bank is made available in [10] for further usage by the scientific community.

2 SincNet architecture

The first layer of SincNet architecture consists of a filter bank comprising of band-pass filters. Sinc function defined as $\frac{Sin(x)}{x}$ has a finite rectangular frequency response, $rect(\frac{f}{f_1})$ centered at 0 Hz. Hence, two Sinc functions, as represented in time-domain by Eq. 1, denotes an ideal band-pass filter in frequency spectrum as specified in Eq. 2.

$$g[n,f_1,f_2] = 2f_2 sinc(2\pi f_2 n) - 2f_1 sinc(2\pi f_1 n) \quad (1)$$

$$G[f,f_1,f_2] = rect(f/2f_2) - rect(f/2f_1) \quad (2)$$

Equation 1 is not a time-limited function; hence time-localized Sinc filters are needed to realize the filter bank for the neural network implementation. Moreover, it was reported that the dynamics of the first layer is highly influential for the overall performance of the neural network [11]. In typical applications, the window functions used are non-negative, smooth, "bell-shaped" profile. Generally, Hamming window of length L defined as $w[n] = 0.54 - 0.46 \times cos(\frac{2\pi n}{L})$ is used in narrow-band applications because of its high frequency selectivity. The time-localized Sinc filters in SincNet are expressed as stated in Eq. 3. Figure 1 shows one such Hamming window based time-localized Sinc filter.

$$g_w[n,f_1,f_2] = g[n,f_1,f_2] \times w[n] \quad (3)$$

In this work, 28 window functions including, tunable windows with trainable parameters are investigated to optimally parameterize Sinc function for the overall improvement of SincNet architecture.

3 Filter design

Windows such as Hamming, Hann, Blackman, Welch, Nuttal, Flattop, Bohman, Bartlett, and Parzen are commonly used depending on the required spectral characteristics of the filter and are fixed-shape window functions. Simple functions such as rectangular and triangular are also investigated. Gaussian, Exponential, Kaiser, Taylor, Chebwin, Tukey, Slepian, and Cosine-Sum are tunable windows that allow to modify the shape of the window. These parameters of tunable windows that allow to alter the shapes are considered as suitable candidates to optimize SincNet architecture in addition to the cutoff frequencies of the band-pass filters. In this work, the same window function is used on all Sinc filters, and hence the number of trainable parameters for the overall SincNet architecture with trainable-windows and with fixed-windows are generalized and illustrated in Table 1. To obtain results comparable to the original SincNet model, the filter length was fixed to 251. The windowing function and its trainable parameters are maintained the same for all the 80 filters implemented in the first layer of the SincNet architecture, which is targeted for speaker recognition task. Hence any increase in the number of trainable parameters remains inconsequential when

Table 1 Trainable parameters of Sinc filters, where N is the size of filter bank

Window functions	No. of trainable parameters
Gaussian, Exponential, Kaiser, Taylor, Chebwin, Tukey, Slepian,	$2N+1$
Cosine-Sum	$2N+(K+1)$; $K>0$
Fixed windows	$2N$

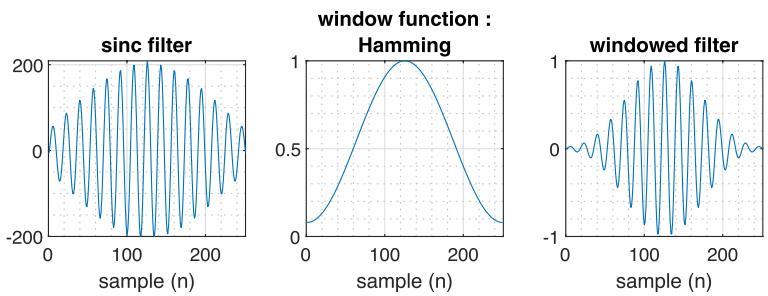

Fig. 1 Windowing effect on the ideal Sinc filter

compared to the 80 filters employed in the first layer. For 28 different windowing functions investigated, the increase in the trainable parameters ranges from 1 to 10 for the entire model, which retains the compact nature of the overall SincNet architecture. Table 1 generalizes the number of trainable parameters where $2N$ is the number of parameters defining N filters employed for the Sinc filter bank in SincNet architecture, and K is the additional trainable parameters based on the windowing function employed (as defined in Eq. 4). Trainable Cosine windows in this work are a general form of Cosine-summation windows as expressed in Eq. 4.

$$w[n] = \sum_{k=0}^{K} (-1)^k a_k cos\left(\frac{2\pi kn}{L}\right), \quad 0 \leq n < L \tag{4}$$

Hann, Hamming, Blackman, Nuttall, Flattop, and Blackman-Harris are considered as special cases of the Cosine-Sum windows, that are defined with fixed coefficients a_k. These set of window functions are referred to as ***Group1-A*** from here on. Table 2 shows the typical coefficient values used to define ***Group1-A*** window functions. ***Group1-B*** refers to a set of general Cosine-Sum window functions whose coefficients are "trainable" within the SincNet architecture for K ranging from 1 to 9. The six fixed-window functions in *Group1-A* mathematically correlate to Cosine-Sum window functions categorized in *Group1-B* with K ranging from 1 to 4. The CER performance between the two sets of functions of similar number of coefficients is further investigated in this work. Tunable windows such as Gaussian, Exponential, Kaiser, Taylor, Chebwin, Tukey, and Slepian are categorized and referred to as ***Group2***. Other fixed-windows, BartlettHann, Rectangular, Welch, Bohman, Triangular, Bartlett, and Parzen are grouped into ***Group3***.

4 Results and discussion

4.1 Dataset and training setup

The TIMIT speech corpus consisting of 4.5 h of read speech [12] as referred from the original SincNet architecture [2] was evaluated for various window functions. The dataset comprises of 6300 sentences, 10 utterances spoken by 630 speakers. SA1, SA2 utterances (being common across all the 630 speakers) were excluded in the experiments to ensure the evaluation is free of content-bias. The remaining 8 utterances were split into train and test set, each comprising of 5 and 3 sentences respectively. Non-speech intervals at the beginning and at the end of each sentence were removed. SincNet convergence is fast compared to conventional CNN. Hence individual training was run for 360 epochs; with validation error and validation loss were reported after every 8th epoch. The SincNet architecture with 80 parameterized Sinc filters of 251 sample length in the first layer, followed by two convolution layers of 60 filters with a filter length of 5 samples were setup. Layer normalization was employed for all the convolution layers including the Sinc functional layer, and for input samples. Three fully connected layer of 2048 neurons with batch normalization were applied post convolution layers. Each speech sentence is split to 200 ms segments with 10 ms overlap. The parameters of the Sinc layer were initialized to mel-scale cutoff-frequencies, whereas the rest of the network was initialized to Glorot initialization scheme. The sentence level classification error rate (CER) is calculated as the average of the frame error rate (FER) obtained over each

Table 2 The coefficients of Cosine-Sum windows, where K is the number of coefficients, as defined in Eq. 4

	Window function	a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
Coefficients of fixed Cosine-Sum windows (***Group1-A***)	Hamming	0.5400	0.4600								
	Hann	0.5000	0.5000								
	Blackman	0.4200	0.5000	0.0800							
	Nuttall	0.3636	0.4892	0.1366	0.0106						
	Blackman-Harris	0.3588	0.4883	0.1413	0.0167						
	Flattop	0.2156	0.4166	0.2773	0.0836	0.0069					
Coefficients of trained Cosine-Sum windows (***Group1-B***)	K=1	0.3102	0.6754								
	K=2	0.2161	0.4907	0.2940							
	K=3	0.2896	0.3509	0.2041	0.1572						
	K=4	0.2398	0.3127	0.1862	0.1606	0.0818					
	K=5	0.2442	0.2794	0.1794	0.1758	0.0626	0.0576				
	K=6	0.0769	0.2079	0.2706	0.1836	0.1216	0.0981	0.0423			
	K=7	0.2540	0.2563	0.1297	0.1241	0.0671	0.0819	0.0284	0.0492		
	K=8	0.2177	0.2600	0.1211	0.1283	0.0644	0.0704	0.0655	0.0370	0.0333	
	K=9	0.1821	0.2075	0.1334	0.1184	0.0633	0.1075	0.0567	0.0562	0.0524	0.0167

variable length sentence (variable number of frames per sentence).

4.2 Trainability of window functions

The parameters defining the shape of tunable window functions were configured as trainable parameters of the SincNet model during the training phase. The variation of trainable parameters during training runs were recorded. Figure 2 demonstrates the trainability of the Gaussian function, where the parameter σ (standard-deviation of the Gaussian function) was initialized to five different values. It converges to values between 0.2 and 0.1 post 360 epochs, to yield the lowest CER. The initial Gaussian window and converged Gaussian window is shown in Fig. 2, to indicate the preferred shape of the windowing function for achieving the lowest CER. Similarly, the trained parameters of ***Group1-B*** Cosine-sum windows(coefficients a_k) post 360 epochs are reported in Table 2. The trainable parameters for the functions falling in ***Group2*** are reported in Table 3. Note that all trainable parameters are extracted for the best CER achieved for the speaker recognition task.

4.3 Window analysis and discussions

The frequency response of 80 time-localized Sinc filters was investigated and analyzed for all 28 window functions primarily with respect to the three spectral parameters stated in the order of significance: (i) bandwidth of the individual filters, (ii) side-lobe suppression, and (iii) spectral leakage between the filters. Five selected temporal and frequency response of the filters from different groups are shown in Fig. 3 for easy reading. The Hamming window belonging to *Group1-A*, and Bohman window from *Group3* shows narrow band-pass spectrum, but large stop-band ripple (~ -60 db). The spectral leakage is also prominently visible in these two window functions. The trainable Gaussian categorized in *Group2*, depicts wide band-pass spectrum, but extremely low stop-band ripples, and spectral leakage. The Flattop function picked from *Group1-A* shows narrow band-pass spectrum. The stop-band ripples, and spectral leakage exists below -120 db. The trainable Cosine-Sum function belonging to *Group1-B* with $K = 4$ shows minimum stop-band ripples and very low spectral leakage with a moderate band-pass spectrum. Between the trainable Gaussian, and trainable Cosine-Sum function, the latter is preferred owing to the narrower band-pass spectrum, whereas the other two spectral parameters are comparable. Overall, the trade-off between the band-pass spectrum over stop-band ripple and spectral leakage is

Table 3 Parameters of *Group2* windows obtained after training

Window function	Trainable parameter	Description	Value
Chebwin	*attn*	Attenuation in *dB*	2.8979
Exponential	τ	Decay parameter	2.0966
Gaussian	σ	Standard deviation	0.1160
Kaiser	β	Shape parameter	0.2898
Slepian	*NW*	Half bandwidth	0.2090
Taylor	*sll* (dB)	Side lobe suppression	2.8979
Tukey	α	Shape parameter	0.0290

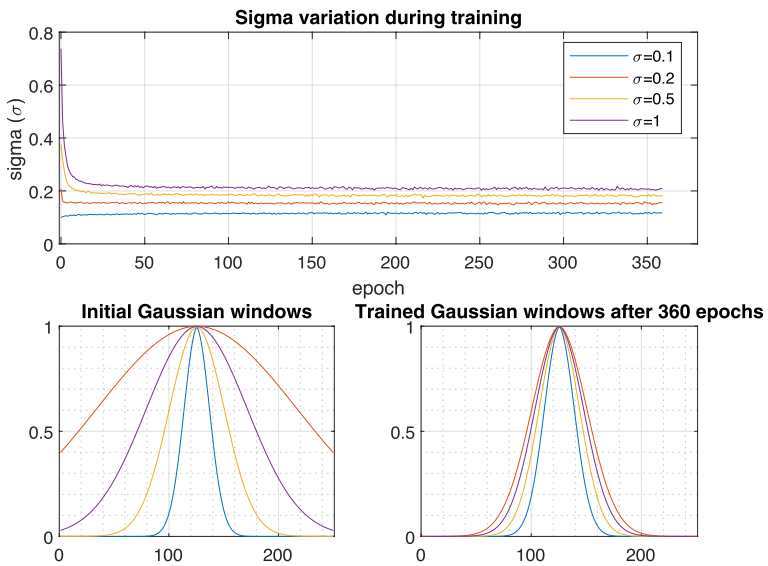

Fig. 2 Convergence of trainable parameter σ and the corresponding Gaussian window functions when initialized to different values during training runs

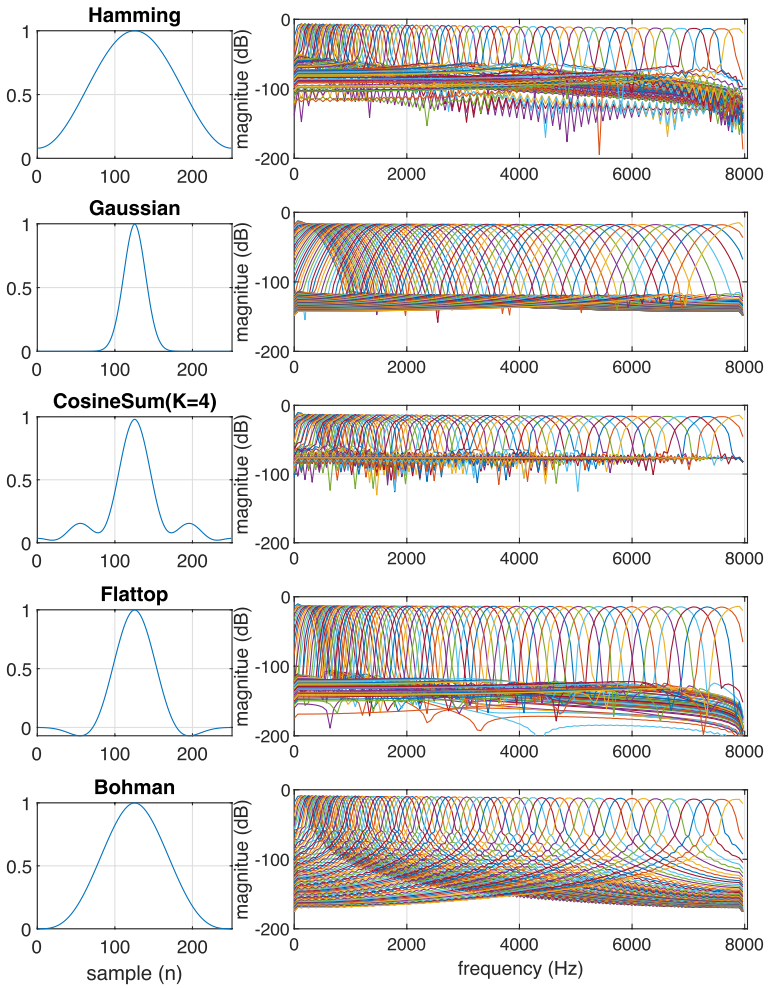

Fig. 3 Frequency response of 80 time-localized Sinc filters using five different window functions

evidently visible in the frequency-response of the filter banks. Additionally, all the windowing functions preserve the mel-scale frequency characteristics which is an incidental outcome in the SincNet for efficient speaker recognition [13]. Frequency response of the other 23 time-localized windows across the filter bank is made available in [10].

4.4 Speaker recognition performance

Figure 4a shows the test CER result for all 28 window functions when individually applied to SincNet architecture over 360 epochs during the training phase. The sentence level classification error rate (CER) is calculated as the average of the frame error rate (FER) obtained over each variable length sentence. The test loss reflects the cross-entropy loss for the multi-class speaker-identification task. This loss shows a classic behavior of having a minimum at some optimal epoch, after which the test-loss increases; corresponding to the fact that the model is overfitting on extended epochs. The Frame-Error Rate (FER) (and sentence-level CER), on the other hand, reflect the error per frame between the predicted label and the ground-truth label. Both FER and CER saturate with the minimum loss model, but continue to show a marginally decreasing trend as in the Fig. 4a, consistently for all window functions. We ascribe this to the fact that overfitted models tend to do well on the unseen test data under the FER/CER metrics. This observation from our experiments across all window functions with regard to the "test-loss and FER/CER profiles" over epochs is consistent with the observation and trends in Ravanelli and Bengio [2]. Please note that the absolute value of CER that falls between 0 to 1 is reported, and these values are not in %. A zoomed-in version of the same at minimum validation loss, i.e., around 25th epoch, and at the last epoch (360th) are also depicted in Fig. 4b and c, respectively. A poorly performing exponential window with large CER at convergence is also annotated in Fig. 4a

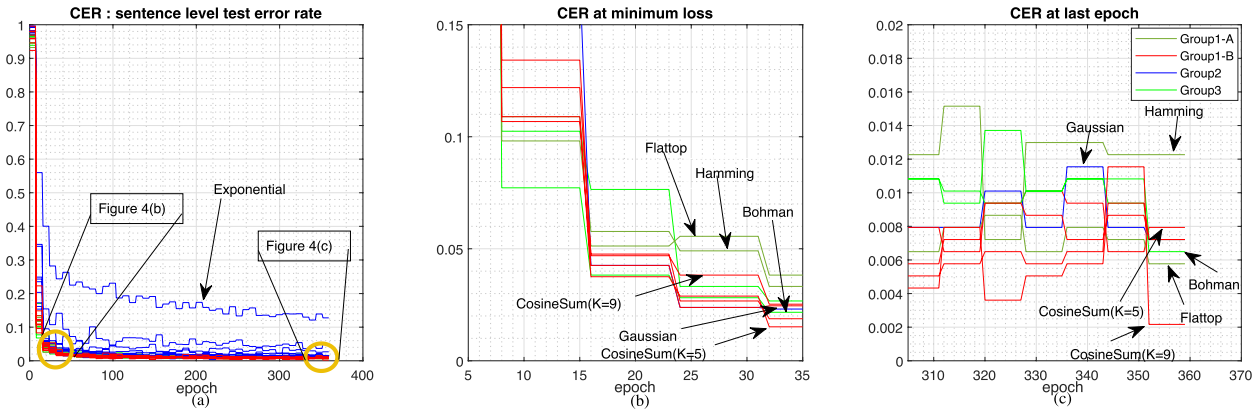

Fig. 4 **a** Sentence level CER for all 28 windows. **b** Prominent window functions at minimum loss epoch and **c** at last epoch. Best viewed in color and in enlarged form

and not included in zoomed-in versions of Fig. 4b and c. The trainable Gaussian and trainable Cosine-Sum functions clearly show a major drop in the CER to a minimum of 0.0072 and 0.0022, respectively, when compared to 0.0123 CER for Hamming window as shown in Fig. 4c. The trainable Gaussian and trainable Cosine-Sum functions converge faster to minimum CER when compared to the other fixed window functions, as shown in Fig. 4a. This further saves the training time for the SincNet architecture.

The absolute values of CER at minimum loss as referred in Fig. 4b and at last epoch as referred in Fig. 4c are reported in Table 4. The training is not stopped early at minimum validation loss, since the CER continues to reduce and the aim was to minimize the CER. Trainable Gaussian function, Bohman, Parzen, and trainable Cosine-Sum functions, especially with $K = 5$, 6, 8, and 9, showcase the best CER performance among the 28 windowing functions investigated and is highlighted in Table 4 for easy reading. The trainable Cosine-Sum function with $K = 9$ reported the best performance with a relative improvement of 82.11% when compared with the Hamming window which is conventionally used for time-localizing Sinc filter banks. Other relative improvements in the speaker recognition performance were exhibited by the trainable Cosine-Sum functions with $K = 4$, 5 (35.77%), $K = 6$, 8 (41.46%), trainable Gaussian (41.46%), and few other functions such as Parzen (41.46%), and Bohman (47.15%). This clearly confirms the need to appropriately select the windowing functions towards parameterizing Sinc filters for an effective outcome in the speaker recognition task. The improvement in the trainable Gaussian and trainable Cosine-Sum functions is attributed to reduced stop-band ripples and minimum spectral leakage which localizes the filter banks as reported in Fig. 3, and thereby extracts the necessary features to offer the best speaker recognition performances.

4.5 Stability of results

The SincNet model was trained with individual windowing functions for 5 independent runs, where the weights are randomly initialized with a different seed every time. The training was continued for 100 epochs. Figure 5 shows the CER performance in terms of mean and standard-deviation (SD) for the minimum loss, and at the 100th epochs. The CER standard deviation (SD) and mean converges to a small value as the training progresses. At 100th epoch, CER mean and SD continues to be small, when compared with the earlier minimum loss epoch for all windowing functions as shown in the Fig. 5a. Low SD suggests that the model training with windowing functions is highly stable and remains independent of the initialized weights and order of the data-points. Figure 5b is an enlarged view showcasing CER performance of *Group1-A*, *Group1-B*, and *Group3* windowing functions at 100th epoch. It is evident that among the three groups of windowing functions, Cosine-Sum (K=5) shows half the CER of Barlett-Hann function, suggesting preference to pick Cosine-Sum (K=5) over other functions. *Group2* functions associated with CER performance at 100th epoch was plotted separately in Fig. 5c. The study shows that the smaller mean value is accompanied with smaller deviations. The website [10] is further updated with the training performance for individual windowing functions for five independent runs initialized with random seeds.

5 Conclusion

Different trainable and fixed windows were thoroughly investigated for SincNet architecture by evaluating sentence level CER for speaker recognition task. The

Table 4 Absolute sentence level CER for all the investigated window functions. Prominent results are highlighted and are also referred in Fig. 4b, c

Group	Window	Minimum loss CER	Last epoch CER
Group1-A (fixed)	Hamming	0.0512	0.0123
	Hann	0.0455	0.0123
	Blackman	0.0253	0.0101
	Nuttall	**0.0209**	0.0130
	Blackman-Harris	0.0361	0.0130
	Flattop	0.0577	**0.0058**
Group1-B (trainable)	Cosine-Sum (K=1)	0.0354	0.0130
	Cosine-Sum (K=2)	0.0339	0.0087
	Cosine-Sum (K=3)	0.0310	0.0072
	Cosine-Sum (K=4)	0.0310	0.0079
	Cosine-Sum (K=5)	**0.0152**	0.0079
	Cosine-Sum (K=6)	0.0238	0.0072
	Cosine-Sum (K=7)	0.0505	0.0087
	Cosine-Sum (K=8)	0.0188	0.0072
	Cosine-Sum (K=9)	0.0253	**0.0022**
Group2 (trainable)	Chebwin	0.0440	0.0130
	Exponential	0.4004	0.1277
	Gaussian	**0.0426**	**0.0072**
	Kaiser	0.0527	0.0130
	Slepian	0.0577	0.0130
	Taylor (nbar=1)	0.0657	0.0123
	Taylor (nbar=5)	0.0599	0.0159
	Taylor (nbar=10)	0.0952	0.0267
	Taylor (nbar=20)	0.1544	0.0498
	Tukey	0.0599	0.0144
Group3 (fixed)	Bartlett-Hann	0.0375	0.0115
	Rectangular	0.0447	0.0115
	Welch	0.0455	0.0144
	Bohman	0.0332	**0.0065**
	Traingular	0.0260	0.0108
	Bartlett	**0.0202**	0.0115
	Parzen	0.0382	0.0072

trainable Cosine-Sum function with $K = 9$ and trainable Gaussian function show relative CER improvement of 82.11% and 41.46% with respect to Hamming window, which is primarily attributed to low stop-band ripple and reduced spectral leakage. The trainability of the Gaussian function was analyzed and its impact on the faster training convergence and improved classification results were noted. The main intention of the work is to study and demonstrate various spectral properties impact and influence towards the final performance in terms of multi-class speaker-recognition task. These spectral properties are essentially pass-band bandwidth (narrower the better), spectral leakage (across adjacent frequencies outside the pass-band), pass- and stop-band ripples, pass-band to stop-band attenuation (higher the better). The various visualizations of these characteristics for each of the window function, allows us to infer the CER performance. The trainable windows for SincNet architecture makes the speaker recognition task highly efficient, and similar configuration is likely to boost other tasks.

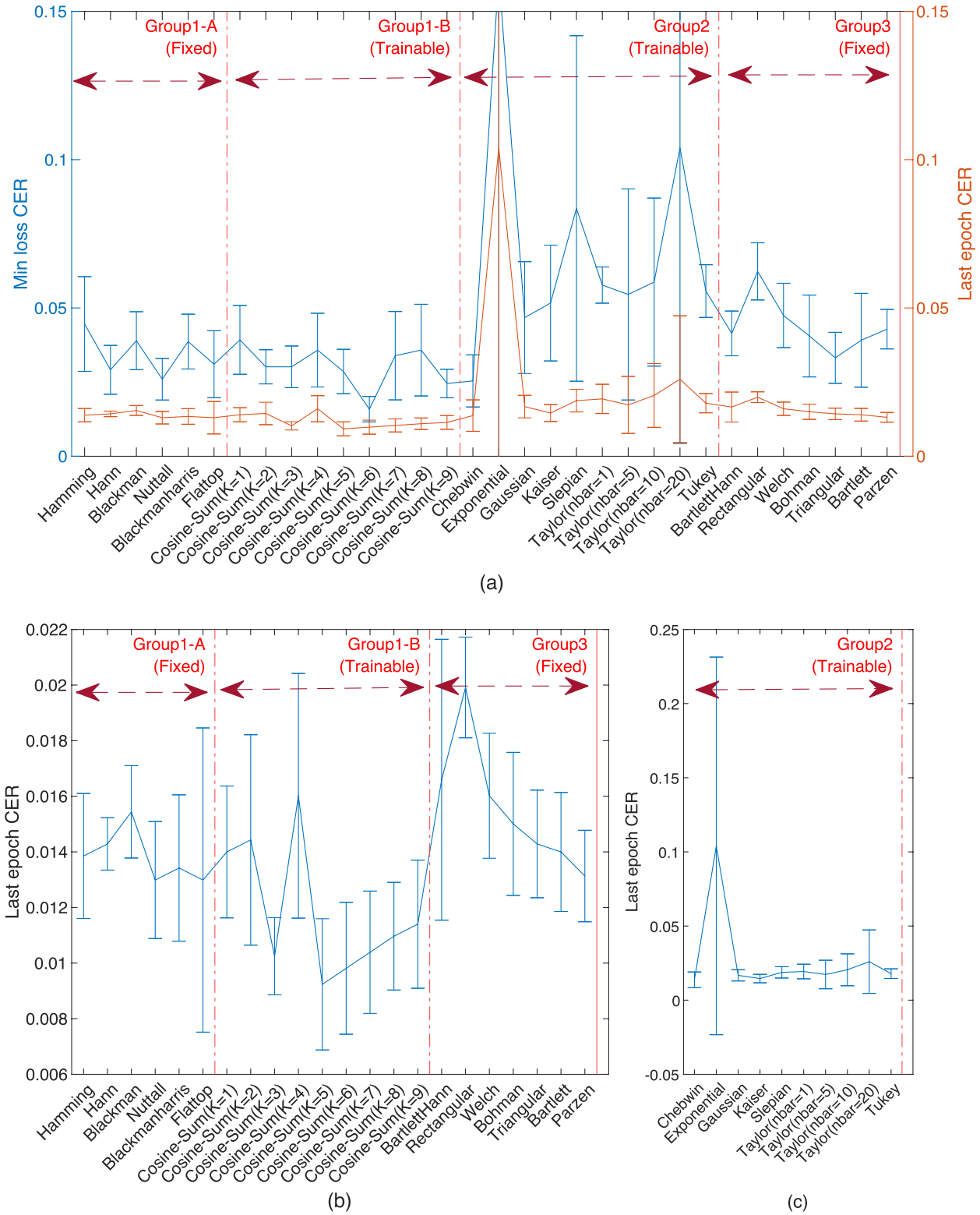

Fig. 5 Mean and standard-deviation of CER for all the investigated window functions **a** at minimum loss (left axis) and at 100th epoch (right axis). **b** Enlarged view of the CER at 100th epoch for *Group1-A*, *Group1-B*, and *Group3* functions, and **c** *Group2* functions

Authors' contributions
All authors contributed to this manuscript. The first author - Prashanth H C investigated all the mentioned filter banks on the SincNet architecture. The second author was instrumental in concept creation to improve SincNet efficiency for speaker recognition. The third and fourth author contributed in the formation of fundamentals and suggesting ways to validate the results.

Authors' information
Prashanth H C is an MS student working at IIIT Bangalore, Dhanya Eledath is a Ph.D student working at IIIT Bangalore, Madhav Rao, and V Ramasubramanian are faculty members working at IIIT Bangalore.

Funding
None

Availability of data and materials
None

Declarations

Ethics approval and consent to participate
None

Consent for publication
Yes

Competing interests
The authors declare that they have no competing interests.

Received: 1 August 2022 Accepted: 6 January 2023
Published online: 19 January 2023

References

1. M. Ravanelli. Deep learning for distant speech recognition (2017). https://arxiv.org/pdf/1712.06086.pdf
2. M. Ravanelli, Y. Bengio, in *2018 IEEE Spoken Language Technology Workshop (SLT)*. Speaker recognition from raw waveform with sincnet (2018), pp. 1021–1028. https://doi.org/10.1109/SLT.2018.8639585
3. D. Eledath, P. Inbarajan, A. Biradar, S. Mahadeva, V. Ramasubramanian, in *2021 29th European Signal Processing Conference (EUSIPCO)*, End-to-end speech recognition from raw speech: Multi time-frequency resolution cnn architecture for efficient representation learning (2021), pp. 536–540. https://doi.org/10.23919/EUSIPCO54536.2021.9616171
4. M. Ravanelli, Y. Bengio, in *Proc. 32nd Conference on Neural Information Processing Systems (NIPS 2018) IRASL workshop, Montreal, Canada*, Interpretable convolutional filters with sincnet. arXiv (2018)
5. S. Mittermaier, L. Kürzinger, B. Waschneck, G. Rigoll, in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Small-footprint keyword spotting on raw audio data with sinc-convolutions (2020), pp. 7454–7458. https://doi.org/10.1109/ICASSP40776.2020.9053395
6. D. Oneață, L. Georgescu, H. Cucu, D. Burileanu, C. Burileanu, in *2020 28th European Signal Processing Conference (EUSIPCO)*, Revisiting sincnet: An evaluation of feature and network hyperparameters for speaker recognition (2021), pp. 1–5. https://doi.org/10.23919/Eusipco47968.2020.9287794
7. L. Chowdhury, M. Kamal, N. Hasan, N. Mohammed, in *2021 International Conference of the Biometrics Special Interest Group (BIOSIG)*, Curricular sincnet: Towards robust deep speaker recognition by emphasizing hard samples in latent space (2021), pp. 1–4. https://doi.org/10.1109/BIOSIG52210.2021.9548296
8. N. Zeghidour, N. Usunier, I. Kokkinos, T. Schatz, G. Synnaeve, E. Dupoux, Learning filterbanks from raw speech for phone recognition (2018). https://doi.org/10.1109/ICASSP.2018.8462015
9. P.G. Noé, T. Parcollet, M. Morchid, Cgcnn: Complex gabor convolutional neural network on raw speech (2020). https://doi.org/10.1109/ICASSP40776.2020.9054220
10. P.H. C. Trainable windows in sincnet. https://sites.google.com/view/sincnet/home. Accessed 10 Sept 2022
11. E. Loweimi, P. Bell, S. Renals, in *Proceedings of Interspeech 2020*, On the robustness and training dynamics of raw waveform models (International Speech Communication Association, 2020), pp. 1001–1005. https://doi.org/10.21437/Interspeech.2020-0017. http://www.interspeech2020.org/. Interspeech 2020, INTERSPEECH 2020 ; Conference date: 25-10-2020 Through 29-10-2020
12. J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, V. Zue, Timit acoustic-phonetic continuous speech corpus. Linguist. Data Consortium (1992)
13. X. Liu, M. Sahidullah, T. Kinnunen, in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, Learnable mfccs for speaker verification (2021), pp. 1–5. https://doi.org/10.1109/ISCAS51556.2021.9401593

Publisher's Note