INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

On Learning Interpretable CNNs with Parametric Modulated Kernel-based Filters

Erfan Loweimi, Peter Bell and Steve Renals

Centre for Speech Technology Research (CSTR), School of Informatics, University of Edinburgh
{e.loweimi, peter.bell, s.renals}@ed.ac.uk

Abstract

We investigate the problem of direct waveform modelling using parametric kernel-based filters in a convolutional neural network (CNN) framework, building on SincNet, a CNN employing the cardinal sine (sinc) function to implement learnable bandpass filters. To this end, the general problem of learning a filterbank consisting of modulated kernel-based baseband filters is studied. Compared to standard CNNs, such models have fewer parameters, learn faster, and require less training data. They are also more amenable to human interpretation, paving the way to embedding some perceptual prior knowledge in the architecture. We have investigated the replacement of the rectangular filters of SincNet with triangular, gammatone and Gaussian filters, resulting in higher model flexibility and a reduction to the phone error rate. We also explore the properties of the learned filters learned for TIMIT phone recognition from both perceptual and statistical standpoints. We find that the filters in the first layer, which directly operate on the waveform, are in accord with the prior knowledge utilised in designing and engineering standard filters such as mel-scale triangular filters. That is, the networks learn to pay more attention to perceptually significant spectral neighbourhoods where the data centroid is located, and the variance and Shannon entropy are highest.
Index Terms: Interpretable CNN, SincNet, parametric modulated kernel-based filters, speech phone recognition

1. Introduction

Deep neural networks (DNN) are among the key breakthroughs in machine learning and have lead to remarkable performance improvement in a wide variety of tasks (e.g. [1, 2]).

However, despite remarkable progress from engineering perspective in building reliable large-scale pattern recognition systems, the understanding about their deep structure has remained shallow. This has triggered an expanding body of work aiming at deciphering the DNNs as black boxes, e.g. [3–6].

Convolutional neural networks (CNNs) are more amenable to interpretation and understanding [7, 8] due to the *convolution*[1] process, and its effect when taking the Fourier transform. This is especially the case when the filters in the first layer directly operate on the raw waveform. In this case, the learned filters could also be compared with handcrafted filters designed using prior knowledge reflecting the properties of the human's auditory system. However, CNN filters are usually not human-interpretable – in either time or frequency domains – and bear little resemblance to perceptually-motivated handcrafted filters.

SincNet is a parametric counterpart of a standard non-parametric CNN in which the filters are modulated cardinal sine (sinc) functions with only two parameters: low and high cut-off frequencies. In the frequency domain, such filters behave as ideal bandpass filters and are highly interpretable. Generally speaking, SincNet, in comparison with conventional CNNs, has the advantages of a (well-chosen) parametric model: higher interpretability and fewer parameters, requiring less training data and offering faster learning/convergence [9, 10]. Furthermore, raw waveform modelling allows the incorporation of phase spectrum information [11–18], overlooked in Fourier transform magnitude-based features such as MFCC.

In this paper, we derive a more general form of interpretable CNNs with parametric modulated kernel-based filters. SincNet is a special case of such models where the kernel function is a cardinal sine. Having derived a general formulation, we explore three alternatives to the sinc function: squared-sinc (sinc^2), gammatone [19–21] and the Gaussian kernels which lead to triangular, gammatone, and Gaussian filterbanks, respectively.

In addition, we conducted a series of analyses to further explore the characteristics of the learned filters in the aforementioned framework. It was found that the network learns to pay more attention to spectral neighbourhoods which are of higher perceptual importance (based on well-established prior knowledge reflecting the properties of the human auditory system), and where (statistically) the centroid of the data exists, and the variance and Shannon entropy (information) [22] are highest.

The rest of this paper is organised as follows. Having reviewed the SincNet in Section 2, in Section 3 we derive a general formulation for interpretable CNNs with parametric modulated kernel-based filters. In Section 4 a set of studies are carried out to explore the properties of the learned filters and their resemblance to the well-established perceptual prior knowledge. Section 5 includes experimental results on phone recognition task along with discussion and Section 6 concludes the paper.

2. SincNet: A CNN with Sinc Kernel

SincNet [9] is a parametric counterpart of the standard non-parametric CNN in which the impulse response of the filters is a subtraction of two sinc functions, resulting in an ideal bandpass filter [23]. As such in SincNet each filter is characterised by only two variables: low (f_1), and high (f_2) cut-off frequencies. The parameter set of the filterbank is given by $\Theta = \{\theta^{(i)}\} = \{f_1^{(i)}, f_2^{(i)}\}$ where i denotes the i^{th} filter in a filterbank with M filters. For a SincNet with impulse response $h(t;\theta^{(i)})$ and frequency response $H(f;\theta^{(i)})$,

$$h(t;\theta^{(i)}) = 2f_2^{(i)}\,\text{sinc}(2f_2^{(i)}t) - 2f_1^{(i)}\,\text{sinc}(2f_1^{(i)}t) \quad (1)$$

$$H(f;\theta^{(i)}) = \Pi(\frac{f}{2f_2^{(i)}}) - \Pi(\frac{f}{2f_1^{(i)}}), \quad (2)$$

where t and f denote the time and frequency independent variables, respectively, and $\Pi(\frac{f}{2B})$ is the rectangular function with unit value for $|f| < B$ and zero outside [24]. The filterbank parameters are learned during training through backpropagation.

[1] Actually, correlation is often computed rather than convolution.

http://dx.doi.org/10.21437/Interspeech.2019-1257

2.1. Practical Considerations

To implement the SincNet, there are some practical considerations which are highlighted in [9, 10]. The Fourier transform of the sinc function equals the ideal brick-wall filter only when the length of this function is infinity [23] which is impractical. Using a sinc function with finite length is equivalent to applying a rectangular window which leads to high frequency leakage and ripple in the passband and stopband [23]. To deal with this issue, the impulse response can be multiplied by a tapered window such as Hamming window. The shape of the tapered window (Hanning, Kaiser, etc.) is not a critical choice [9, 10].

When learning the filter parameters, there is no guarantee that the learned values remain positive during training. To cope with this problem, in [9, 10] the following was used: $f_1 \leftarrow |f_1|$ and $f_2 \leftarrow f_1 + |f_2 - f_1|$. In addition, since f_2 cannot be larger than Nyquist rate, it should be upper-bounded during training.

Note that learning the amplitude or gain value for each filter is not necessary because the feature maps are multiplied by the weights of the higher layer and they implicitly play the role of the filter gain. The SincNet parameters, $\{f_1^{(i)}, f_2^{(i)}\}$, may be initialised using a perceptual scale such as mel [9].

2.2. Advantages

Conventional non-parametric CNNs require L parameters for a filter length of L samples. However, for the SincNet two parameters are required for each filter, regardless of the filter length. In [9, 10], L was set to 129 which is two orders of magnitude larger than SincNet parameters. Fewer parameters can lead to requiring less data for effective training, faster learning and convergence, and better generalisation.

Fig. 1 illustrates the impulse and frequency responses of the learned filters for conventional CNN and SincNet networks trained for TIMIT phone recognition. Fig. 2 shows their performance at different epochs. As can be observed, the SincNet is more interpretable in both time and frequency domains, converges faster and results in a lower phone error rate (PER).

The characteristics of the learned filters in this framework can be compared with the handcrafted filters designed based on perceptual priors. This paves the way for exploring the agreement between what network finds important and the perceptual prior knowledge reflecting what human auditory system considers important which is interesting from both theoretical and practical standpoints. Such models also allow for incorporating some prior knowledge into the DNN when kernels with biologically plausible functionality are applied, e.g. employing triangular or gammatone filters instead of rectangular ones.

To this end, we derive a general formulation for such CNNs with kernel-based modulated filters.

3. Kernelised CNNs

Replacing the sinc with its definition in (1) and converting the sum of sinusoids to a product[2] along with some algebraic manipulation, results in the following formulation

$$h^{(i)}(t) = 2B^{(i)}\,\mathrm{sinc}(B^{(i)}t)\;\cos(2\pi f_c^{(i)}t), \qquad (3)$$

where $B^{(i)} = f_2^{(i)} - f_1^{(i)}$ and $f_c^{(i)} = \frac{f_1^{(i)}+f_2^{(i)}}{2}$ denote the bandwidth and centre frequency of the i^{th} filter, respectively.

The advantage of (3) is that the i^{th} filter of the filterbank can be expressed as a product of a baseband kernel, $K(t;\theta^{(i)})$, modulating the carrier [25], $\cos(2\pi f_c^{(i)}t)$,

$$h^{(i)}(t;\theta^{(i)}, f_c^{(i)}) = K(t;\theta^{(i)})\;\mathrm{carrier}(t; f_c^{(i)}), \qquad (4)$$

where $\theta^{(i)}$ is the kernel parameter set. The kernel and carrier parameters, i.e. $\Theta = \{\theta^{(i)}, f_c^{(i)}\}$, are learned during training.

For SincNet the kernel is the sinc function with $\theta^{(i)} = B^{(i)}$ parameter, but in general the kernel and its parameter set can be different. We examine three kernels: squared-sinc (Sinc^{2}Net), gammatone (GammaNet) and Gaussian (GaussNet).

[2] $\sin\alpha - \sin\beta = 2\sin\frac{\alpha-\beta}{2}\cos\frac{\alpha+\beta}{2}$

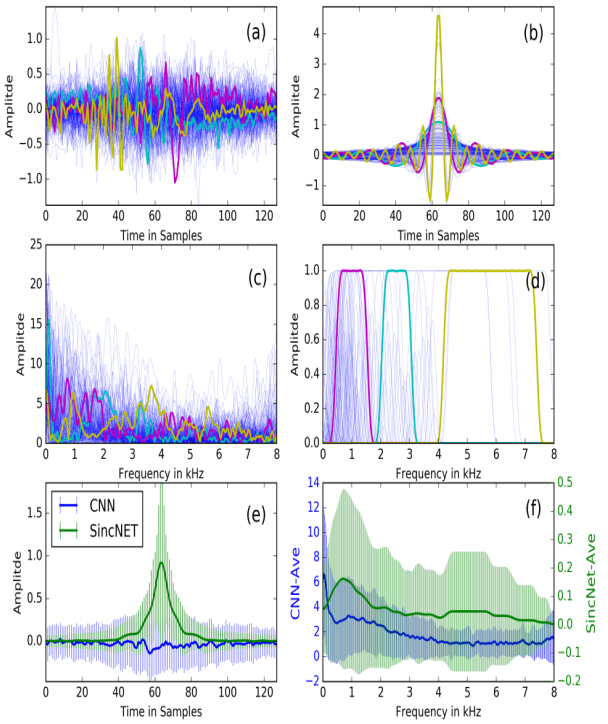

Figure 1: *CNN vs SincNet. Three filters are plotted with different colours: (a) CNN impulse responses, (b) SincNet impulse responses, (c) CNN frequency responses, (d) SincNet frequency responses, (e) average impulse response, (f) average frequency response along with standard deviation (shaded area).*

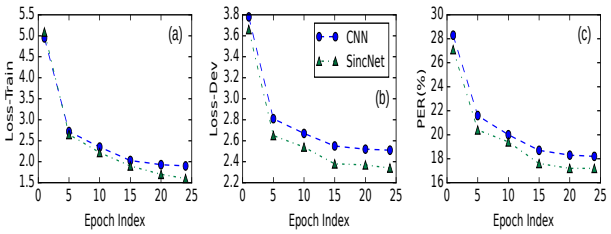

Figure 2: *CNN vs SincNet using different performance metrics: (a) loss on train data, (b) loss on devset, (c) PER.*

3.1. Sinc^{2}Net: Triangular Filters

MFCC's triangular filterbank [26] is widely-used in speech processing and is a natural choice to consider. The corresponding kernel (parametric impulse response) is squared-sinc, sinc^2,

$$K(t;\theta^{(i)}) = A^{(i)}\,\mathrm{sinc}^2(B^{(i)}t), \qquad (5)$$

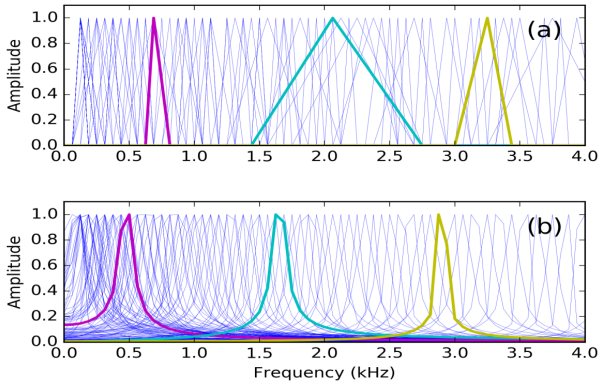

Figure 3: *(a) Sinc^{2}Net and (b) GammaNet learned filterbanks. Three filters are plotted with different colours.*

where $A^{(i)}$ and $B^{(i)}$ denote the amplitude and bandwidth (in Hz) of the i^{th} filter, respectively. To be more precise, the triangular filters in MFCC are not symmetric around the center frequency, therefore, their Fourier transform is not exactly sinc^2. To incorporate this into the model, one may add an extra degree of freedom to the kernel function in (5).

3.2. GammaNet: Gammatone Filters

Gammatone filters offer another perceptually motivated kernel:

$$K(t;\theta^{(i)}) = A^{(i)} t^{(N^{(i)}-1)} e^{-2\pi B^{(i)} t}, \qquad (6)$$

where $B^{(i)}$ and $N^{(i)}$ are the bandwidth (in ERB[3] scale [27]) and order of the i^{th} filter, respectively [20]. In this case, $\theta^{(i)} = \{A^{(i)}, B^{(i)}, N^{(i)}\}$. A typical value for order is four [21].

3.3. GaussNet: Gaussian Filters

If a Gaussian kernel, $K(t;\theta^{(i)}) = A^{(i)} \exp(-t^2/2\sigma_i^2)$, is used, then assuming B_i is the 3 dB bandwidth of the i^{th} filter in Hz, it can be shown that $\sigma_i = \sqrt{log2}/(2\pi B_i)$ [25]. The network can learn either B_i or σ_i, depending on the implementation.

4. Perceptual and Statistical Studies on Kernel-based CNNs

In this section, we explore the properties of the learned filters in the proposed kernel-based framework and compare them with handcrafted filterbanks designed based on perceptual prior knowledge. Fig. 3 depicts filterbanks learned by Sinc^{2}Net and GammaNet for TIMIT [28] phone recognition. For a better visualisation, and to avoid cluttering, the horizontal axis was limited to 4 kHz (TIMIT sampling rate is 16 kHz).

4.1. Centre Frequency Distribution

Fig. 4 shows the histogram (distribution) of the centre frequencies of the kernel-based modulated CNN filters along with uniform (uni), Mel, Bark, and ERB filterbanks using the same number of filters (128) and 50% overlap in the mentioned scale. As seen, consistent with perceptual scales inspired from the human auditory system, there are noticeably more filters operating in frequencies below 2000 Hz (histogram knee point). This implies that the network learns to be more discriminative and selective in processing those spectral components.

[3]Equivalent Rectangular Bandwidth

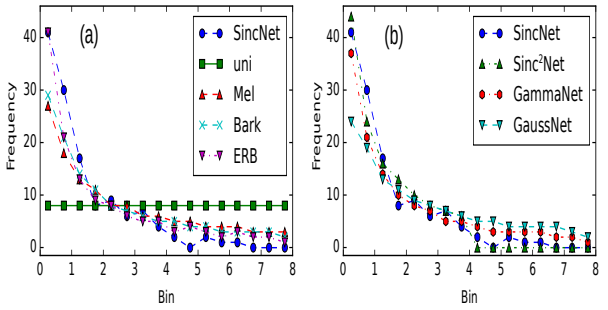

Figure 4: *Histogram of the centre frequencies (in kHz) of the kernel-based filters vs those of filterbanks designed using perceptual scales. (a) conventional filters, (b) kernel-based filters.*

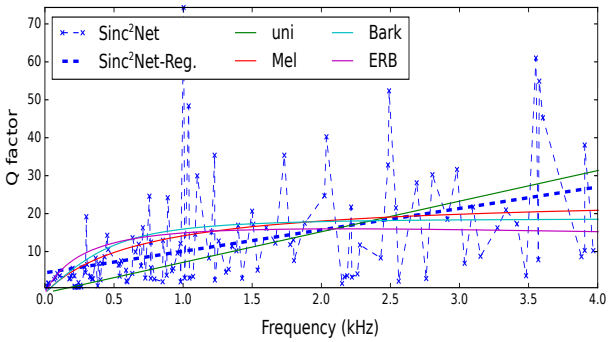

Figure 5: *Q factor for the filters belonging to different filterbanks with various scales along with that of* Sinc^2*Net.*

4.2. Filter Quality Factor

To investigate the filters' bandwidth along with the corresponding center frequency we have used the quality factor (Q) [29], the fraction of the filter centre frequency to its bandwidth [30]. Conventional filters behave almost like constant-Q filters (above 1000 Hz, Fig. 5), so that when the filter centre frequency increases, the bandwidth goes up, although the fraction remains constant. At higher frequencies the filters become wider which implies poorer spectral resolution.

For Sinc^{2}Net the Q factor of the filters is not constant, nor is the filter bandwidth variation monotonic, unlike conventional filters. However, performing a linear regression reveals that the Q factor of the filters increases as the centre frequency increases,similar to when doing linear regression for Q factor of the perceptual scales. This trend was observed for SincNet, GammaNet and GaussNet, too. To verify, that such trend is not a random effect, we performed further experiments with different initialisations: in all runs the same trend is observed (Fig. 6). Additionally, Fig. 5 indicates that monitoring the Q-factor can be useful during training to avoid outlier filters.

4.3. GammaNet Filter Order

Gammatone filters have an extra parameter, the filter order $N^{(i)}$ (Section 3.2). The typical order of four correlates well with cochlea filters [19, 21, 31]. To explore this relation we trained a GammaNet and allowed each filter to have an individual order. No particular constraint was imposed on the filter order values during training. Table 1 shows the statistics of the order of the learned filters. As may be observed, the average value for the learned order is 4.3 which is close to the typical value.

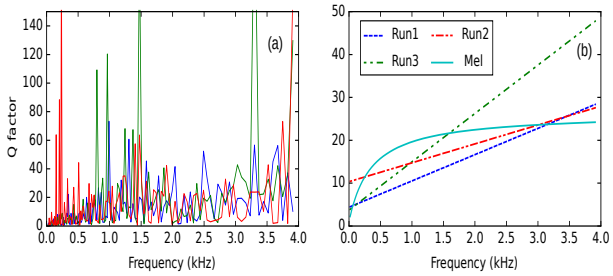

Figure 6: *Q factor of learned filters in three runs of $Sinc^2$Net.*

Table 1: *Statistics of the GammaNet learned filters order.*

	Mean	Median	Std	Min	Max
GammaNet	4.39	4.30	0.97	1.73	6.80

4.4. Learned filters and Statistical Properties of the Data

So far, we have investigated the optimality of the filters from a perceptual viewpoint by comparing them with well-established prior knowledge. It is also insightful to know the relationship (if any) between the regions where the network pays the most attention and the statistical properties of the training data. To do this, we compare the mean of the frequency responses of the learned filters with the statistical properties of the data, namely its mean, standard deviation (std) and Shannon entropy at each frequency bin. All the TIMIT training data ($\sim$ 1.4 M frames) was used for estimating the statistics.

As seen in Fig. 7, the network not only learns to be more selective at perceptually important spectral bands but also gives more attention to a part of the spectrum with highest statistical value. That is, the peak of the average frequency response (mostly attended frequency bins by the network) are in a spectral neighbourhood where the centroid of the data is located, and the variance and Shannon entropy are highest.

5. Experimental Results

5.1. Setup

Different architectures are compared on TIMIT [28] phone recognition. The initial alignments are taken from models built by the Kaldi [32] standard recipe for TIMIT. The DNN models were built using PyTorch-Kaldi [33,34] standard recipe with the same hyperparameters setting, including 200ms frame length and 10ms frame shift. For all models the same network is used; on top of the first layer which is a kernel-based CNN, an MLP consisting of five layers with 1024 nodes and ReLU [35] activation is employed. Number of epochs was set to 24 and optimisation was carried out using RMSProp [36].

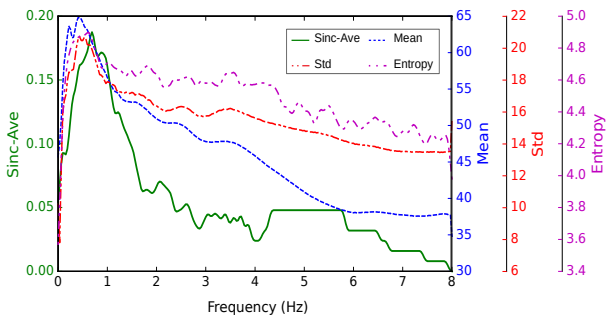

Figure 7: *TIMIT Mean/Std/Entropy for each bin vs SincNet average frequency response. All TIMIT training data is used.*

Table 2: *TIMIT PER for different kernels (200ms).*

	MLP	CNN	Sinc	$Sinc^2$	Gamma	Gauss
PER	18.5	18.2	17.6	16.9	17.2	17.0

Table 3: *TIMIT PER for different frame lengths (ms).*

	25	50	100	200	300	400
CNN	30.0	21.7	18.8	18.2	18.6	19.0
SincNet	27.7	20.6	17.6	17.4	17.6	17.7
$Sinc^2$Net	27.1	20.7	17.3	16.9	17.4	17.7

5.2. Results and Discussion

Table 2 shows the PER for an MLP trained using filterbank features (25ms) along with various CNN-based models which take the raw waveform as input (200ms). As may be observed, the CNN and kernel-based models outperform the conventional features. Compared with SincNet, the proposed kernel-based techniques lead to slightly lower PER. The difference remains low because although, for example, the triangular or gammatone filters are more biologically plausible and lead to a better performance in the shallow GMM-HMM systems [37, 38], the five MLP hidden layers, can compensate for the low-level suboptimality associated with SincNet rectangular filters.

Finally we consider the optimal frame length for direct waveform modelling using kernel-based filters. As Table 3 illustrates, the optimal frame length for all kernels is about 200ms which is considerably larger than the conventional 25ms used in Fourier-based front-ends. This allows the network to learn a short- to medium-term representation that is potentially useful in tasks where some medium-term speech properties should be recognised (e.g. speaker identification) or suppressed (e.g. speaker-independent ASR).

Why is 200ms frame length optimal for such models? In other ASR systems which take raw waveforms as input, shorter frames are typically used [39–43]. Although further exploration using other databases and tasks is warranted, possible answers include: learning some kind of temporal masking [44]; coarticulation [45]; or optimal syllable modelling, noting that the mean syllable length in English is 200ms [46].

6. Conclusions

In this paper the problem of direct waveform modelling through CNNs with parametric modulated kernel-based filters was investigated. This generalised framework was built on SincNet, a CNN with sinc kernel. In the proposed structure, the model variables are the kernel parameters and the modulator frequency. Squared-sinc, Gammatone and Gaussian kernels were studied and the properties of the learned filters was investigated from perceptual and statistical viewpoints. It was shown that the learned filterbanks, not only pay more attention to spectral bands with higher perceptual importance, but also to regions where the variance and Shannon entropy (information) are highest. Deployment of the CNNs with parametric modulated kernels as well as improving the interpretability of the DNNs, paves the way for embedding some prior knowledge in the network architecture through using perceptually-inspired kernels. This opens up a broad avenue for future research.

Acknowledgements: Supported by EPSRC Project EP/R012180/1 (SpeechWave). We benefited from discussions with Zoran Cvetkovic (KCL).

7. References

[1] W. Xiong, L. Wu, F. Alleva, J. Droppo, X. Huang, and A. Stolcke, "The Microsoft 2017 conversational speech recognition system," in *IEEE ICASSP*, 2018.

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim *et al.*, "English conversational telephone speech recognition by humans and machines," in *Interspeech*, 2017.

[3] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," in *ICLR*, 2017.

[4] R. Shwartz-Ziv and N. Tishby, "Opening the black box of deep neural networks via information," arXiv:1703.00810, 2017.

[5] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital Signal Processing*, vol. 73, pp. 1 – 15, 2018.

[6] E. Loweimi, P. Bell, and S. Renals, "On the usefulness of statistical normalisation of bottleneck features for speech recognition," in *ICASSP*, 2019.

[7] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *ECCV*, 2014, pp. 818–833.

[8] Q. Zhang, Y. Nian Wu, and S.-C. Zhu, "Interpretable convolutional neural networks," in *IEEE ICCV*, 2018, pp. 8827–8836.

[9] M. Ravanelli and Y. Bengio, "Speaker and speech recognition from raw waveform with SincNet," in *IEEE ICASSP*, 2019.

[10] ——, "Interpretable convolutional filters with SincNet," in *NIPS Workshop IRASL*, 2018.

[11] E. Loveimi and S. Ahadi, "Objective evaluation of phase and magnitude only reconstructed speech: New considerations," in *IEEE ISSPA*, May 2010, pp. 117–120.

[12] E. Loweimi, S. Ahadi, and H. Sheikhzadeh, "Phase-only speech reconstruction using very short frames." in *INTERSPEECH*. ISCA, 2011, pp. 2501–2504.

[13] E. Loweimi, S. Ahadi, and T. Drugman, "A new phase-based feature representation for robust speech recognition," in *IEEE ICASSP*, May 2013, pp. 7155–7159.

[14] E. Loweimi, J. Barker, and T. Hain, "Source-filter separation of speech signal in the phase domain." in *INTERSPEECH*. ISCA, 2015, pp. 598–602.

[15] ——, "Statistical normalisation of phase-based feature representation for robust speech recognition," in *IEEE ICASSP*, March 2017, pp. 5310–5314.

[16] E. Loweimi, J. Barker, O. Saz Torralba, and T. Hain, "Robust source-filter separation of speech signal in the phase domain," in *INTERSPEECH*, Sweden, 2017, pp. 414–418.

[17] E. Loweimi, J. Barker, and T. Hain, "On the usefulness of the speech phase spectrum for pitch extraction," in *Proc. INTERSPEECH 2018*. ISCA, 2018, pp. 696–700.

[18] E. Loweimi, "Robust phase-based speech signal processing; from source-filter separation to model-based robust asr," Ph.D. dissertation, University of Sheffield, Sheffield, UK, Feb 2018. [Online]. Available: http://etheses.whiterose.ac.uk/19409/

[19] R. D. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice, "An efficient auditory filterbank based on the gammatone function," MRC Applied Psychology Unit, Cambridge, Tech. Rep., 1987.

[20] M. Slaney, "An efficient implementation of the Patterson-Holdsworth auditory filter bank," Apple Computer Technical Report 35, 1993.

[21] M. Cooke, *Modelling Auditory Processing and Organisation*. Cambridge University Press, 1993.

[22] C. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, 1948.

[23] A. Oppenheim and R. Schafer, *Discrete-Time Signal Processing*, 3rd ed. Prentice Hall, 2009.

[24] A. Oppenheim, A. Willsky, and S. Nawab, *Signals and Systems*, 2nd ed. Prentice-Hall, 1996.

[25] S. Haykin and M. Moher, *Communication Systems*. Wiley, 2010.

[26] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, pp. 357–366, 1980.

[27] B. C. J. Moore and B. R. Glasberg, "Suggested formulae for calculating auditory-filter bandwidths and excitation patterns." *Journal of the Acoustical Society of America*, vol. 74, no. 3, pp. 750–3, 1983.

[28] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT acoustic phonetic continuous speech corpus CDROM," 1993. [Online]. Available: http://www.ldc.upenn.edu/Catalog/LDC93S1.html

[29] C. A. Desoer and S. K. Ernest, *Basic Circuit Theory*. McGraw Hill, 1969.

[30] B. Gold, N. Morgan, and D. Ellis, *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*, 2nd ed. New York, NY, USA: Wiley-Interscience, 2011.

[31] D. Schofield, "Visualisations of speech based on a model of the peripheral auditory system," *NASA STI/Recon Technical Report N*, vol. 86, Jul. 1985.

[32] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *IEEE ASRU*, 2011.

[33] M. Ravanelli, T. Parcollet, and Y. Bengio, "The PyTorch-Kaldi speech recognition toolkit," in *IEEE ICASSP*, 2019.

[34] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS Workshop on Autodiff*, 2017.

[35] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *ICML*, 2010, pp. 807–814.

[36] T. Tieleman and G. Hinton, "Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude," COURSERA: Neural Networks for Machine Learning, 2012.

[37] B. Shannon and K. K. Paliwal, "A comparative study of filter bank spacing for speech recognition," in *Microelectronic Engineering Research Conference*, 2003.

[38] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (pncc) for robust speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1315–1329, July 2016.

[39] Z. Tuske, P. Golik, R. Schlüter, and H. Ney, "Acoustic modeling with deep neural networks using raw time signal for LVCSR," in *Interspeech*, 2014.

[40] D. Palaz, R. Collobert, and M. Magimai-Doss, "Analysis of CNN-based speech recognition system using raw speech as input," in *Interspeech*, 2015.

[41] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNs," in *Interspeech*, 2015.

[42] Y. Hoshen, R. J. Weiss, and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *IEEE ICASSP*, 2015.

[43] Z. Tuske, R. Schlüter, and H. Ney, "Acoustic modeling of speech waveform based on multi-resolution, neural network signal processing," in *IEEE ICASSP*, 2018.

[44] B. C. J. Moore, *An introduction to the psychology of hearing*. Brill, 2012.

[45] Y. Chow, R. Schwartz, S. Roucos, O. Kimball, P. Price, F. Kubala, M. Dunham, M. Krasner, and J. Makhoul, "The role of word-dependent coarticulatory effects in a phoneme-based speech recognition system," in *IEEE ICASSP*, 1986, pp. 1593–1596.

[46] S. Greenberg, "Speaking in shorthand – a syllable-centric perspective for understanding pronunciation variation," *Speech Communication*, vol. 29, no. 2, pp. 159 – 176, 1999.