On the Robustness and Training Dynamics of Raw Waveform Models

INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

On the Robustness and Training Dynamics of Raw Waveform Models

Erfan Loweimi, Peter Bell and Steve Renals

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

Abstract

We investigate the robustness and training dynamics of raw waveform acoustic models for automatic speech recognition (ASR). It is known that the first layer of such models learn a set of filters, performing a form of time-frequency analysis. This layer is liable to be under-trained owing to gradient vanishing, which can negatively affect the network performance. Through a set of experiments on TIMIT, Aurora-4 and WSJ datasets, we investigate the training dynamics of the first layer by measuring the evolution of its average frequency response over different epochs. We demonstrate that the network efficiently learns an optimal set of filters with a high spectral resolution and the dynamics of the first layer highly correlates with the dynamics of the cross entropy (CE) loss and word error rate (WER). In addition, we study the robustness of raw waveform models in both matched and mismatched conditions. The accuracy of these models is found to be comparable to, or better than, their MFCC-based counterparts in matched conditions and notably improved by using a better alignment. The role of raw waveform normalisation was also examined and up to 4.3% absolute WER reduction in mismatched conditions was achieved.
Index Terms: ASR, acoustic modelling, raw waveform, training dynamics, average frequency response

1. Introduction

Feature engineering has been an active research area in speech processing. Hand-crafted features such as MFCC [1] and PLP [2] have been designed to utilise knowledge of the human auditory perception and speech production mechanisms, with consideration of desirable modelling properties. In spite of their general effectiveness, they are application-blind and discard information (such as the phase spectrum [3–7]) along a fixed pipeline, without considering the final task objective.

Deep neural networks (DNNs) successfully tackle the parameterisation problem through implicitly and jointly learning the front-end and back-end with the guidance of the objective function optimiser. However, even if they manage to perfectly process the input information, they cannot compensate for the information lost during the extraction of input features.

A possible solution to this issue is to supply the network with minimally processed or, ideally, raw data, minimising the potential information loss. However, this is a challenging route to take as the input space expands substantially, making training highly demanding. Nevertheless, in light of the current powerful computing infrastructures and training techniques, there is a growing body of work in which DNNs taking the raw waveform as input, have been found to yield promising results in acoustic modeling for ASR [8–19], which is our focus here.

In raw waveform models, as explained in Section 3, the first layer is tasked with learning a set of optimal filters that perform a quasi time-frequency analysis. However, due to its location in a deep structure, this layer is most susceptible to the vanishing gradient phenomenon. The first issue which we investigate here is to what extent gradient vanishing is problematic for the first layer. What are the training dynamics of this layer, i.e. how fast and how accurately does it evolve towards an optimal set of filters? We investigate this through a set of oracle experiments in controlled noisy conditions and show that the first layer of a raw waveform model can effectively learn an optimal representation, and efficiently filter out noisy subbands with a high spectral resolution and at a reasonable speed.

Having shown that such models are discriminant enough to accurately filter clean/noisy subbands, we investigate the robustness of raw waveform models in matched and mismatched noisy conditions, and explore techniques to enhance their performance. Experimental results on Aurora-4 [20] show that such systems return comparable to better WER in matched conditions w.r.t. their MFCC-based counterpart and employing better alignment could notably improve their performance. We also show that in mismatched conditions a proper normalisation can significantly improve the performance.

2. Raw waveform modelling

Raw waveform acoustic modelling could be traced back to a decade ago [21–24] and has attracted much attention over the last five years. Palaz et al [8] investigated the usefulness of raw waveform models on the TIMIT phone recognition task and showed CNNs to have superior performance over fully-connected networks. Later a similar CNN was tested on the WSJ task [12], achieving comparable results to an MFCC-DNN system. Moreover, transfer learning of the first layer between TIMIT and WSJ tasks was explored. The robustness of this system on Aurora-2 [25] connected-digit task in matched and mismatched conditions has also been investigated [13].

Tuske et al [9] compared raw waveform features with traditional features in an LVCSR task (50 hours) with a DNN-based acoustic model. MFCCs outperformed other features, with a 10% absolute WER reduction compared to raw waveform features, but use of ReLU non-linearity and 5-fold more data (250 hours) reduced the gap to 2.4%. They also interpreted the first DNN layer weights as impulse responses and demonstrated their resemblance to auditory filters. In later work the first two layers were replaced with a CNN [14], and the max-pooling layer was substituted by a time-convolutional layer with low-pass filters that extract envelopes at various sampling rates, returning a multi-resolutional representation [17].

Sainath et al [10, 26] deployed a CLDNN architecture for raw waveform modelling: a cascade of CNN, LSTM and fully-connected DNN architectures to leverage their complementary modeling capacities. It was trained with about 2000 hours speech and was the first system to outperform its log-filterbank based counterpart. They employed this structure for beamforming in a multi-channel scenario, similar to [11], which utilised raw-waveform models for joint acoustic modelling and beam-

Supported by EPSRC Project EP/R012180/1 (SpeechWave).

http://dx.doi.org/10.21437/Interspeech.2020-17

forming. They also showed that stacking raw-waveform and log-filterbank could provide further WER reduction [10].

Ghahremani et al [15] used a TDNN along with a network-in-network [27] architecture for raw waveform modeling and investigated the usefulness of i-vector for speaker adaptation.

Zhu et al [16] performed multi-scale acoustic modeling by inserting the raw waveform into three parallel branches of CNNs with 1ms, 4ms and 40ms filter lengths, with the aim of feeding higher layers with a representation with both high temporal and spectral resolutions. Von Platen et al [28] gained a similar advantage through the use of multiple (three) streams of information created through decomposing the speech signal into different frame lengths (multi-span) around each frame centre.

The above cited work uses conventional CNNs, which are a set of non-parametric FIR filters with L free parameters, where L is the filter length in samples (taps). In another line of research, CNN filters are assumed to be parametric, resulting in filters characterised by many fewer parameters that learn faster and are trainable with less data, at the cost of a lower modelling capacity. Assuming the first layer learns a quasi filterbank, not necessarily an abstract representation, any collection of bandpass filters seems to be sufficient.

Examples of parametric CNNs include TD-Filterbanks [18, 29] and SincNet [19,30] where each (bandpass) filter is modeled by a Gabor waveletor Sinc function, respectively, and is characterised by only centre frequency and bandwidth. We recently extended this idea to a more generalised modulated Kernel-based CNNs and investigated Sinc^{2}Net, GammaNet and Gauss-Net in which the filters take triangular, Gammatone and Gaussian shapes, respectively [31]. Fainberg et al [32] studied the possibility of speaker adaptation via retraining the Sinc layer parameters and promising results were achieved.

3. Analysis the Dynamics of the First Layer

In raw waveform models, the first layer aims to learn an optimal set of filters which perform a quasi time-frequency analysis. It should ideally propagate through the task-relevant information and filter out nuisance aspects of the input. However, due to the vanishing gradient phenomenon, the error-correcting signal that the first layer receives is weaker than higher layers. This may lead to poor and/or slow training and suboptimal filters.

To scrutinise this issue, we begin by studying the learning dynamics of the first layer through a set of oracle experiments in a controlled noisy variant of the TIMIT [33] phone recognition task. Signals are contaminated by an additive white Gaussian noise with 0 dB SNR passed through a cascade of two bandpass filters with the following passbands: 1.2 kHz to 1.6 kHz and 1.8 kHz to 2.1 kHz. Fig. 1(g) shows the spectrogram of a typical noisy signal. We placed the two noisy subbands close to each other, leaving a narrow clean subband in between, to evaluate the spectral resolution of the first layer.

We measure the sensitivity of the first layer to different frequency bins through computing the *average frequency response* (AFR) of the learned filters. The evolution of the AFR over different epochs reflects the first layer's learning dynamics during training. Such an oracle experiment paves the way for measuring the optimality of the learned filters, speed of learning and the degree to which the vanishing gradient is problematic. For a better understanding and visualisation of the first layer's training dynamics, we have studied both conventional non-parametric CNNs and parametric SincNet raw waveform models. DNNs were trained using PyTorch-Kaldi [34–36] with a default configuration (more details in Section 4).

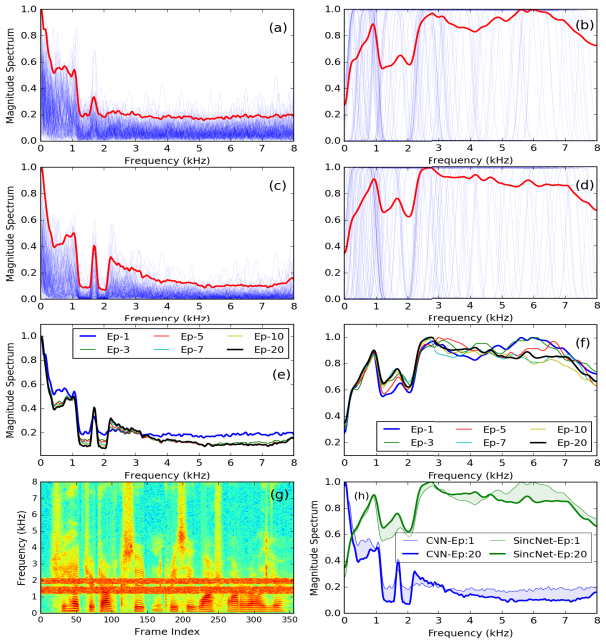

Figure 1: *Training dynamics of the first layer of CNN and SincNet. Frequency response of the learned filters (blue) along with average (red) after epoch 1: (a) CNN, (b) SincNet; after epoch 20 (c) CNN, (d) SincNet. AFR at different epochs (e) CNN, (f) SincNet. (g) spectrogram of a training data. (h) AFR dynamics for CNN and SincNet, from epoch 1 to 20.*

3.1. Simulation results

Fig. 1 illustrates the AFR of the first layer at different epochs. At the end of epoch 1, the CNN has not yet distinguished the clean and noisy subbands (Fig. 1(a)) whereas the SincNet has approximately found not only the noisy subbands but also the clean narrow subband in between (Fig. 1(b)). This is not surprising given the number of parameters of each model. As seen in Fig. 1(c) and (d), after 20 epochs both models have an optimal frequency response for the given task. The CNN shows a better spectral resolution (sharper transitions) due to its higher number of parameters and consequently modelling capacity.

Fig. 1(e) and (f) depict the training dynamics between epoch 1 and epoch 20. As seen, after about 10 epochs both raw waveform models reach a reliable estimation of clean/noisy subbands, and more epochs further fine tune the frequency response. This illustrates that gradient vanishing – at least as far as learning the filterbank is concerned – does not limit the learning capabilities of the first layer and can be safely neglected.

3.2. Effect of activation function on training dynamics

Fig. 2 (a) and (b) show the training dynamics (shaded area between epoch 1 to 20) in both clean (normal TIMIT) and the aforementioned noisy scenario when applying ReLU, Tanh and Sigmoid activation functions. As seen, the shaded area for ReLU is the smallest and for Sigmoid it is noticeably larger. This implies that ReLU contributes towards a faster learning, which could be attributed to sparsity [37,38]. Also note that the shaded area for the clean training is smaller than the noisy one.

How correlated are the convergence at the first layer and the overall performance of the system? To investigate this, the phone error rate (PER) for each system vs epoch was plotted in Fig. 2(c) and (d). As seen, the convergence trends for learn-

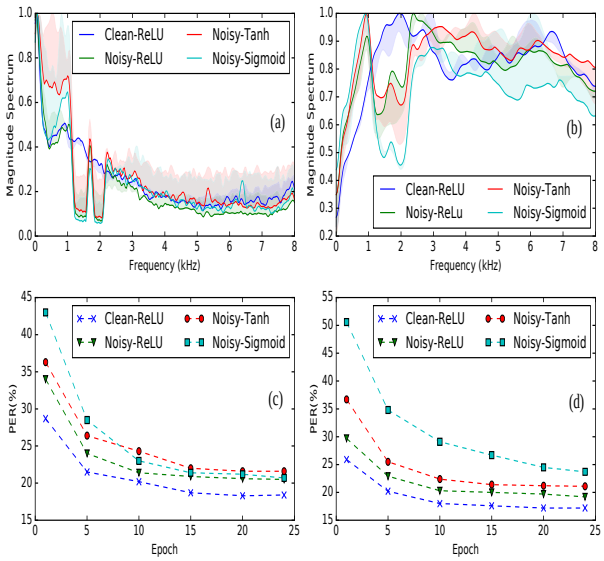

Figure 2: *Training dynamics (epoch 1 to 20 (solid line)) of the first layer in clean and noisy conditions for different activation functions (ReLU, Tanh and Sigmoid): (a) CNN, (b) SincNet. Corresponding PER vs epoch: (c) CNN, (d) SincNet.*

ing the filters at the first layer and the performance curve (the knee point) are similar. For example, when using ReLU, PER convergence occurs around the tenth epoch (Fig. 2 (c) and (d)), coinciding with the epoch at which the filter learning process gets very close to the optimal AFR (Fig. 1(e) and (f)).

3.3. Effect of database on training dynamics

To explore the data effect on training dynamics, we first compute the AFR evolution (epochs 1 to 20) for TIMIT as well as the clean and multi-style training data of Aurora-4 [20]. As seen in Fig. 3(a) and (b), the AFR for both TIMIT and Aurora-4 (clean) is similar, but the training dynamics (shaded area) for Aurora-4 is narrower. This is owing to the fact that at the end of epoch one, the network has seen more data (almost 5.4 hours for TIMIT versus 14 hours for Aurora-4) and therefore is in a more stable position, evolving less in later epochs.

The shaded area for Aurora-4 is proportional to the complexity of the training data. As Fig. 3 (c) and (d) show, for the CNN the shaded area for multi-style training (noisy speech) is wider than for clean training. Recall that similar trend was observed for TIMIT in Fig. 2 (a) and (b). Such a trend reaffirms the fact that for learning from more complex data, more epochs are needed. For SincNet, the shaded area is narrower as each filter has only two parameters to learn which in turn, constrains the hypothesis space and consequently the training dynamics.

To further investigate the dynamics of the first layer, we trained a similar system using WSJ [39] with 81 hours of speech. Furthermore, we computed the mean square error between the AFR at each epoch and the AFR of the last epoch[1], calling it *AFR-Error*. Now, we wish to measure the correlation between the dynamics of the AFR-Error with other performance metrics such as cross-entropy (CE) loss and WER. As illustrated in Fig. 4, the knee point and dynamics of the AFR-Error (epoch 5) is very similar to those of the CE loss and WER (epoch 7). Furthermore, Table 1 quantitatively shows that the correlation of the AFR-Error dynamics with the CE and WER is very high.

[1]Assuming AFR of the last epoch is the optimal frequency response.

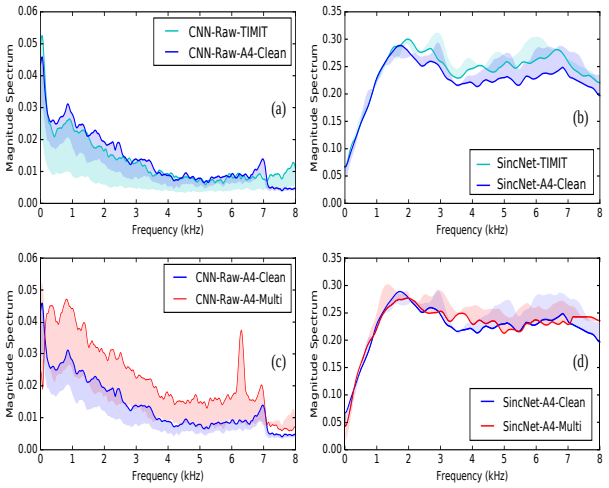

Figure 3: *Training dynamics of the first layer in TIMIT and Aurora-4 tasks. Training dynamics in clean condition: (a) CNN, (b) SincNet. Training dynamics for Aurora-4 in clean and multi-style training modes: (c) CNN, (d) SincNet.*

Table 1: *Correlation of the AFR-Error with other measures.*

	CE-Train	CE-Dev	WER-Dev	WER-Eval
Corr	0.99	0.94	0.88	0.95

4. ASR Experiments

4.1. Experimental Setup

In all experiments we have used CNNs, trained using PyTorch-Kaldi [34, 35] with default settings (without monophone regularisation), including layer normalisation [40], batch normalisation [41] and dropout [42]. Experiments were carried out on WSJ and Aurora-4 in both clean and multi-style training modes. Alignments were taken from the respective Kaldi recipes [36]. The Aurora-4 test set consists of four subsets: A (clean speech), B (additive noise), C (channel mismatch) and D (additive noise and channel mismatch). Ave in Fig. 5 and Table 3 is computed as follows: $(A + 6B + C + 6D)/14$. Symbols $*$ and † indicate the features are dimension-wise mean-variance normalised (MVN) at the utterance (*) and speaker (†) levels, respectively.

4.2. Results and Discussion

Fig. 5 (a) shows the results for clean training on Aurora-4 with a high degree of mismatch. In this case MFCC features clearly outperform the raw waveform models and log-mel filterbank (FBank) with average (Ave) WER of 13.8% (after MVN at utterance level). Superiority of MFCC is owing to the information loss along its pipeline which is the highest and could minimise mismatch. In this scenario, a proper normalisation is highly beneficial for all features. Also, SincNet clearly surpasses CNN-Raw, in contrast to the multi-style training (Fig. 5 (b)) where the gap is marginal. Recall that in CNN-Raw each filter is characterised with many more free parameters, making it vulnerable to overfitting. This shortcoming increases the system's vulnerability when the mismatch level is high.

Fig. 5 (b) illustrates the recognition results for multi-style training. As seen, the gap between the MFCC system and the raw-waveform model is substantially lower and, in fact, raw waveform models outperform their MFCC-based counterparts, although still lagging behind FBank feature with an average

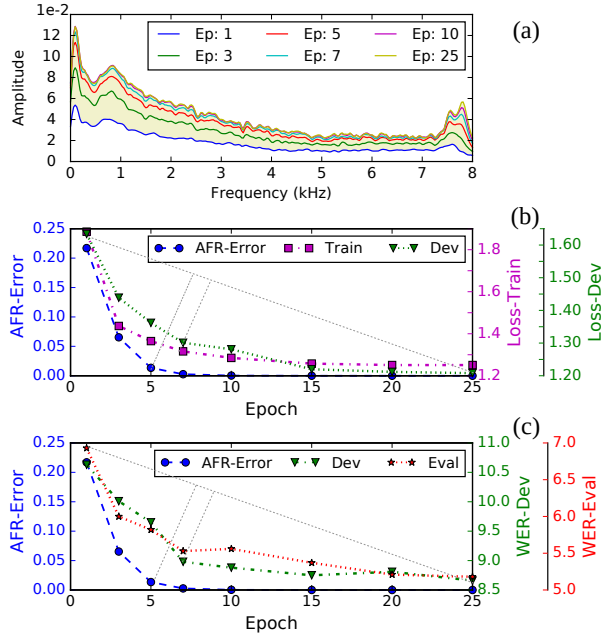

Figure 4: *Dynamics of the first layer of a CNN-Raw model trained on WSJ. (a) AFR at different epochs; (b) Dynamics of AFR-Error, CE-Loss-Train (SI284) and CE-Loss-Dev (Dev93); (c) Dynamics of AFR-Error, WER-Dev and WER-Eval (Eval92).*

Table 2: *WSJ WER for different front-ends.*

	MFCC†	FBank†	CNN-Raw	Sinc-Raw
Dev93	10.4	9.1	8.6	8.5
Eval92	6.8	5.9	5.1	5.0

WER of 9.2% (after MVN at utterance level). Note that while for clean training mode the MVN* was beneficial, in the multi-style case, its influence is marginal for raw waveform models, contrary to classic features.

Fig. 5 also depicts the training dynamics (WER vs epoch) of the MFCC system along with raw waveform models. As seen, for clean-training mode, the network converges quickly while for multi-style training it requires more iterations. This is in harmony with Fig. 3(c) where the shaded area was wider for multi-style data. It could be justified considering the fact that distilling information from a mixture of clean speech and noise (additive and channel) in the multi-style mode is more challenging than learning the subword units from clean signals.

Table 2, shows the WER of the classic features vs raw waveform for WSJ. As seen, the raw waveform noticeably outperforms the FBank and MFCC. Why does raw waveform outperform FBank here, while in Aurora-4 FBank results in a lower WER? One explanation could be the fact that raw waveform models demand more training data: 14 hours for Aurora-4 vs 81 hours for WSJ. However, if the amount of training data was the *only* decisive factor, then why does raw waveform (with similar architectures) outperform the classic features for smaller task such as TIMIT, as shown in Table 2 in [31]?

4.3. Importance of Alignment

For Aurora-4 in multi-style mode, the training data is noisy. Compared with clean-training mode, the noisy data could induce some level of uncertainty in the learnt model and its output. When doing alignment with such model to get the training labels, the labels are likely to be noisier, giving rise to *teacher error*. This error further affects the raw waveform models as they operate in a feature space with remarkably higher dimension. If this hypothesis is true, then deploying a better alignment should be more beneficial to the raw waveform models.

Generally, a system with a better WER, does not necessarily supply a better alignment. To get a reliably better alignment, we take advantage of a special property of the Aurora-4: the noisy signals are generated by synthetically adding noise; hence, the alignment of the clean signals and their noisy version is exactly the same. Therefore, we compute the alignment for the clean signals using a model trained only on clean data and use that alignment for training the DNN in multi-style mode using the noisy counterpart of the clean signals. As seen in Table 3, by doing so, the raw waveform clearly outperforms the classic features. It substantiate our hypothesis and shows that the raw waveform models are more sensitive to the teacher error.

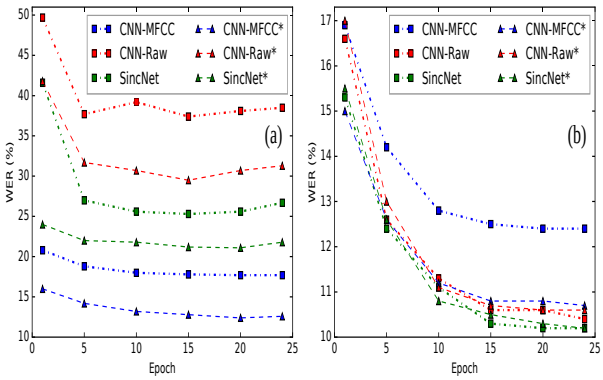

Figure 5: *Effect of MFCC and raw waveform normalisation on average WER for Aurora-4. Training mode: (a) clean, (b) multi.*

Table 3: *Aurora-4 WER for multi-style training mode when alignments are taken from model trained in clean mode.*

Feature	A	B	C	D	*Ave*
CNN-MFCC*	3.5	6.1	4.6	8.3	6.7
CNN-FBank*	3.0	5.2	3.3	6.4	5.4
CNN-Raw	2.7	4.4	4.0	6.4	5.1
SincNet-Raw	2.9	4.6	3.9	6.7	5.3

5. Conclusion

The first layer of the raw waveform models performs a quasi time-frequency analysis. However, owing to its location as the first layer in a deep structure, and the vanishing gradient phenomenon, it is susceptible to under-training. We demonstrated that the first layer of a raw waveform model efficiently learns optimal filters and can pass or block clean or noisy subbands with high spectral resolution. Moreover, it was shown that the dynamics of the first layer is highly correlated with the evolution of the performance measures such as CE loss and WER over time. We also investigated the robustness of the raw waveform models in matched and mismatched conditions. In mismatched conditions we observed a performance gap with the MFCC-based system, which was noticeably reduced through feature normalisation. In matched conditions, comparable or better results was achieved and using a better alignment notably improved the performance. Studying the dynamics of the first layer jointly with higher layers is recommended for future work.

6. References

[1] S.B. 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.

[2] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, Apr. 1990.

[3] D.A. Leigh and K.K. Paliwal, "Short-time phase spectrum in speech processing: A review and some experimental results," *Digital Signal Processing*, vol. 17, no. 3, pp. 578 – 616, 2007.

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

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

[6] P. Mowlaee, J. Kulmer, J. Stahl, and F. Mayer, *Single Channel Phase-Aware Signal Processing in Speech Communication: Theory and Practice*, Wiley, 2016.

[7] Erfan Loweimi, *Robust Phase-based Speech Signal Processing; From Source-Filter Separation to Model-Based Robust ASR*, Ph.D. thesis, University of Sheffield, Feb 2018.

[8] D. Palaz, R. Collobert, and M. Magimai-Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks," in *INTERSPEECH*, 2013, pp. 1766–1770.

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

[10] 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.

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

[12] D. Palaz, M. Magimai-Doss, and R. Collobert, "Convolutional neural networks-based continuous speech recognition using raw speech signal," in *ICASSP*, 2015.

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

[14] P. Golik, Z. Tüske, R. Schlüter, and H. Ney, "Convolutional neural networks for acoustic modeling of raw time signal in LVCSR," in *INTERSPEECH*, 2015.

[15] P. Ghahremani, V. Manohar, D. Povey, and S. Khudanpur, "Acoustic modelling from the signal domain using CNNs," in *INTERSPEECH*, 2016.

[16] Z. Zhu, J. H. Engel, and A. Hannun, "Learning multiscale features directly from waveforms," in *INTERSPEECH*, 2016.

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

[18] N. Zeghidour, N. Usunier, I. Kokkinos, T. Schatz, G. Synnaeve, and E. Dupoux, "Learning filterbanks from raw speech for phoneme recognition," in *ICASSP*, 2018.

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

[20] N. Parihar and J. Picone, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," Tech. Rep., Inst. for Signal and Information Process, Mississippi State University, 2002.

[21] M. Ager, Z. Cvetkovic, P. Sollich, and B. Yu, "Towards robust phoneme classification: Augmentation of PLP models with acoustic waveforms," in *2008 16th European Signal Processing Conference, EUSIPCO 2008, Lausanne, Switzerland, August 25-29, 2008*. 2008, IEEE.

[22] M. Ager, Z. Cvetkovic, and P. Sollich, "Robust phoneme classification: Exploiting the adaptability of acoustic waveform models," in *17th European Signal Processing Conference, EUSIPCO 2009, Glasgow, Scotland, UK, August 24-28, 2009*. 2009, pp. 530–534, IEEE.

[23] J. Yousafzai, Z. Cvetkovic, and P. Sollich, "Subband acoustic waveform front-end for robust speech recognition using support vector machines," *2010 IEEE Spoken Language Technology Workshop*, pp. 253–258, 2010.

[24] Navdeep Jaitly and Geoffrey E. Hinton, "Learning a better representation of speech soundwaves using restricted boltzmann machines.," in *ICASSP*, 2011, pp. 5884–5887.

[25] D. Pearce and H.G. Hirsch, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *INTERSPEECH*, 2000, pp. 29–32.

[26] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *ICASSP*, 2015.

[27] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv:1312.4400*, 2013.

[28] P. von Platen, C. Zhang, and P. C. Woodland, "Multi-span acoustic modelling using raw waveform signals," in *INTERSPEECH*, 2019.

[29] P. Noé, T. Parcollet, and M. Morchid, "Cgcnn: Complex gabor convolutional neural network on raw speech," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7724–7728.

[30] T. Parcollet, M. Morchid, and G. Linarès, "E2e-sincnet: Toward fully end-to-end speech recognition," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7714–7718.

[31] E. Loweimi, P. Bell, and S. Renals, "On learning interpretable CNNs with parametric modulated kernel-based filters," in *INTERSPEECH*, 2019.

[32] J. Fainberg, O. Klejch, E. Loweimi, P. Bell, and S. Renals, "Acoustic model adaptation from raw waveforms with SincNet," in *ASRU*, 2019.

[33] 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," 1993.

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

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

[36] 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.

[37] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *AISTATS*, 2011, pp. 315–323.

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

[39] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, 1992, pp. 899–902.

[40] L. J. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv:1607.06450*, 2016.

[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.