UNDERSTANDING RAW WAVEFORM BASED CNN THROUGH LOW-RANK SPECTRO-TEMPORAL DECOUPLING

Vinayak Abrol S. Pavankumar Dubagunta
Mathew Magimai.-Doss

Idiap-RR-11-2019

OCTOBER 2019

Centre du Parc, Rue Marconi 19, P.O. Box 592, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

Understanding Raw Waveform based CNN through Low-rank Spectro-Temporal Decoupling

Vinayak Abrol[1], S. Pavankumar Dubagunta[2,3], and Mathew Magimai.-Doss*[2]

[1]Mathematical Institute, University of Oxford, United Kingdom
[2]Idiap Research Institute, Martigny, Switzerland
[3]École polytechnique fédérale de Lausanne (EPFL), Switzerland

October 2019

Abstract

Acoustic modeling using convolutional neural networks with raw waveform input has become popular for many speech processing tasks. In the series of convolutional layers, the first layer acts as a frequency selective filter-bank, and the subsequent ones involve 2D spectro-temporal filters. Since each filter focuses on a different spectro-temporal pattern, there is an inherent redundancy in the learned filters. In this work we exploit this redundancy and propose an alternative approach to train robust networks for speech applications. This is achieved by spectro-temporal decoupling, where each 2D filter is approximated by a linear combination of a low-rank set of spectral/temporal basis filters. This filtering operation can be seen as filtering temporal spectral energy trajectories followed by their combination across spectral bands or vice versa. With a case study on speech recognition, we show that the proposed scheme is robust, efficient and achieves comparable/better baseline accuracy.

Keywords Speech recognition, low-rank modeling, convolutional neural network, end-to-end learning.

1 Introduction

A generic speech processing system consist of a feature extractor and a statistical module for classification/inference. Many such conventional systems employ hand-crafted short-term spectral features based on knowledge from speech production and perception. In the recent years, with advances in neural networks, there has been an interest in reducing as much feature engineering as possible. For instance, 1) by modeling intermediate representations such as filterbank coefficients computed on a linear [1] or Mel scale [2], spectrograms [3, 4], and further, 2) by directly modeling raw speech signals [5, 6, 7, 8, 9, 10, 11, 12] using 1D convolution neural networks (CNNs) at the input stage, in an end-to-end manner. We focus on one such end-to-end raw-waveform CNN architecture from [5], which consists of multiple convolutional layers followed by fully-connected layers. Such systems were shown to achieve competitive performance in multiple applications such as speech recognition, speaker verification, and gender recognition [13, 11, 12, 14].

In general, in the case of raw waveform based CNNs, the first layer learns a short time-frequency decomposition of the signal: for instance in speech recognition, it tends to behave as a log-spaced frequency selective filter-bank [6]. Further, in the case of speaker verification, depending on whether the block-processing is segmental (about $1-3$ pitch period duration) or sub-segmental, the first layer was shown to focus on voice source related [12] and vocal tract system related speaker discriminative information [15], respectively. Following the first layer, each filter in the subsequent layers performs 1D temporal filtering with filters spanning all the spectral bands from the previous layer. In other words, it achieves an effect of 2D spectro-temporal filtering, combining the information from both the spectral and temporal dimensions [7]. In the context of speech signals, while the temporal domain retains its intuitive meaning, the spectral domain refers to the information space defined by the different channels/filters of the previous layer. We show that there is an inherent redundancy in the very nature of how the convolution is performed after the first CNN layer, and one can perform spectro-temporal

*This work was carried out at the Idiap Research Institute, Martigny and was partially funded by the Swiss National Science Foundation (SNSF) through the project Unified Speech Processing Framework For Trustworthy Speaker Recognition (UniTS), and by the Hasler Foundation through the project Flexible Linguistically-guided Objective Speech aSsessment (FLOSS). The authors gratefully thank the funding agencies for their financial support, and all the project partners for a fruitful collaboration. e-mail: abrol@maths.ox.ac.uk,{pavankumar.dubagunta,mathew}@idiap.ch.

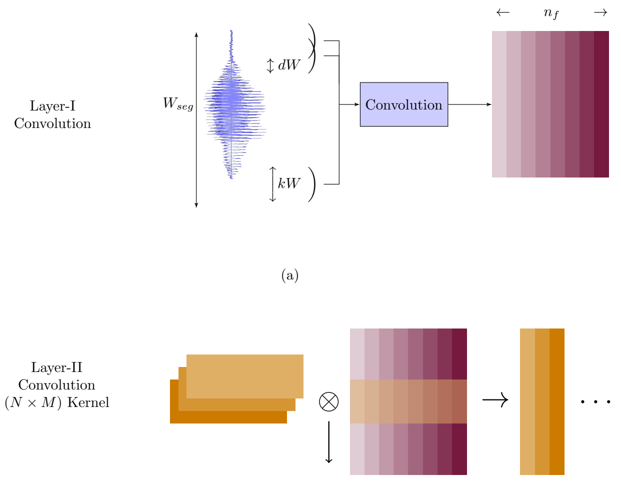
decoupling for efficient modelling of the information in these spaces separately. This is motivated by traditional feature extraction approaches where 1) first the information is combined across frequency bands followed by temporal modelling e.g., cepstrum with delta and acceleration features; 2) first individual temporal spectral trajectories are modeled followed by combination across spectral bands, e.g. RASTA features.

In this work, we propose a framework for spectro-temporal decoupling in CNNs to learn robust and efficient raw waveform based networks. We show that this can be achieved by exploiting the redundancy in a CNN filter using its low rank decomposition. This also provide insights about how the CNNs trained on raw waveforms model the spectro-temporal information from speech signals. In particular, we propose a storage efficient LRC (Low Rank Convolution) based CNN for speech processing, where we approximate a 2D spectro-temporal kernel/filter by a linear combination of a few rank-k basis set of filters without sacrificing the baseline accuracy. Here, rank selection is dependent on the individual layer's temporal context. This filter set is first used to generate intermediate representations, which are then linearly combined to get the same effect as applying a 2D filter. This leads to an efficient architecture with a significant reduction in the amount of trainable parameters. This contrasts with works that takes an already trained network and attempt to perform a low-rank factorization of the filter, e.g. see [16]. To the best of author's knowledge, this is the first work that simply makes LR decomposition the original structure of the network for speech processing. It can be argued that the proposed approach appears to be similar to the concept of recently proposed depthwise separable convolution (DSC) for 2D-CNNs in the context of computer vision [17]. However, inherently DSC does not have the notion of low rank decomposition, and we show that compared to DSC, LRC generalizes well and is more suitable for speech processing applications across datasets. Although the proposed approach can be extended to 2D-CNNs, this paper only focuses on 1D-CNNs in context of raw waveform based processing, and we demonstrate its utility through a study on automatic speech recognition (ASR) systems.

The rest of the paper is organized as follows: Section 2 describes the raw waveform based CNN followed by the proposed low-rank CNN in Section 3. Section 3.3 & 3.4 presents the relevant background work. Section 4 presents the experimental setup, and Section 5 presents the corresponding results obtained with the proposed systems and the baseline systems. Finally, Section 6 concludes the paper.

2 CNN for raw speech modelling

This work extends upon the recent ASR studies using raw waveform based 1D-CNNs [6, 7, 8] and readers are encourage to follow [5] for a detailed description of the baseline system. As depicted in Fig. 1(a), in the first layer, each of the n_f filters of length kW performs a dW strided convolution with the raw input waveform of length W_{seg} resulting in 1D outputs. Parameter kW kernel width determines if the block processing is segmental or sub-segmental. One can relate the output of first layer as output of a frequency selective filter-bank. Following the first layer, each second layer filter, while operating at a time step, processes the responses of all the first layer filters while having a temporal context. In other words, each filter spans the entire spectral dimension and a part of the temporal dimension as shown in Fig. 1(b). This amounts to a 2D filtering operation, where filter moves only in the temporal dimension.



(a)

(b)

Figure 1: Raw waveform CNN: Each color depicts a filter and its corresponding output. ⊗ denotes the convolution operation and ↓ depicts the temporal dimension i.e., the direction of convolution.

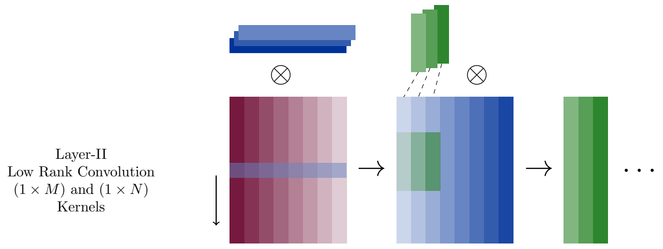


Figure 2: Low-rank convolution (spectral first processing). Each color depicts a filter and its corresponding output. $\otimes$ denotes the convolution operation and ↓ depicts the temporal dimension i.e., the direction of convolution.

3 Proposed CNN with Low-rank Filters

As described in previous section, after the first layer, convolution with a 2D filter amounts to filtering temporal trajectories of scaled spectral energies, followed by an operation to combine them along different spectral bands or vice versa. Since such sub-bands can overlap and there are correlations among adjacent temporal values, the learned 2D filters have a lot of redundancies. Thus, the information of interest lies in a very low dimension and we argue that speech can be efficiently modelled by decoupling the spectral and temporal information learned by these filters. This can be achieved by exploiting the low-rank structure for filter approximation. Mathematically, the convolution of an input $\mathbf{X}$ with a filter $\mathbf{W}$ in the second layer can be expressed as

$$\mathbf{W} \otimes \mathbf{X} \approx \mathbf{U} \otimes (\mathbf{V} \otimes \mathbf{X}) \approx (\mathbf{U} \otimes \mathbf{X}) \otimes \mathbf{V} \tag{1}$$

where $\otimes$ denotes the convolution (sample-by-sample multiplication followed by a sum) and $\mathbf{W} = \mathbf{UV}$ is the rank-k decomposition of the filter $\mathbf{W}$. Fig. 2 depicts the idea of realizing the convolution layer using the proposed low-rank filters compared to the original convolution. For illustration, we show the case of rank-1 decomposition at layer 2, where instead of applying a 2D filter of size $T \times M$, filtering is performed in two steps:

1. first a 1D spectral filter is used to model all the M bands corresponding to each temporal point individually.
2. the resulting outputs are then combined using a 1D temporal filter which models T temporal points simultaneously.

We believe that in this way by increasing the network depth (which increases expressivity) and combining decoupled spectro-temporal information, the network will be more robust to degradations. This has further implications: for e.g. depending on the application one can emphasize either the spectral or temporal related information. We now describe the LRC operations in detail and its rank-k generalization in the next sections.

3.1 Rank-1 Decomposition

Let us assume that, at any layer, the input to the CNN is of dimension $T \times M$, where T is the temporal dimension and M is the number of input channels. Further, there are C channel outputs with filters/kernels of size $N \times M$ each. Using rank-1 decomposition, each CNN layer is realized with two CNN layers using a $N \times 1$ temporal filters and a $1 \times M$ spectral filters. The number of trainable parameters for each filter reduces from NM to $N + M$, which saves a huge amount of computation. Since convolution is a linear operation, depending on the application, there is flexibility of interchanging the operating layers, e.g., apply a $M \times 1$ spectral filter followed by a $1 \times N$ temporal filter (see Fig. 2). We denote these low-rank CNN configurations as '*temporal first*' - processing along dimension N or '*spectral first*' - processing along dimension M.

3.2 Rank-k Decomposition

The proposed LRC filters can be easily generalized for any rank-k. This can be achieved by realizing each CNN filter using a total of $k + 1$ low-rank filters. Here, $k < N$ is the dimensionality of the underlying low-rank subspace. For temporal-first processing we will have projections on k subspaces via $N \times 1$ filters, and a $k \times M$ filter combining outputs from these k subspaces. Similarly, in spectral first processing we will have projection via k $1 \times M$ filters and a combination of resultant outputs via a $N \times k$ filter. Note that in principle each low rank decomposition is achieved with filter pairs i.e., each spectral filter has a corresponding temporal filter. In practice, having a $k \times M$ combination filter is easy to implement rather than performing a sum over outputs of k $1 \times M$ filters.

3.3 Comparison with Depthwise Separable Convolution

DSC is also a form of factorized convolution where a standard convolution layer is replaced by a depthwise convolution (applying a single filter to each input channel individually), followed by combining the resultant outputs by a pointwise convolution [17]. Popular architectures in computer vision such as MobileNet [18], EffNet [19], ShuffelNet [20], and Inception architecture [21] use Depthwise separable convolutions. In contrast, the proposed LRC is a generalized separable convolution based on the principle of low rank decomposition, which can be applied in either of the filter dimensions. DSC does not have any notion of low rank approximation. Further, the underlying mechanism of action of DSC and its interpretation in terms of extreme Inception hypothesis is still not fully understood [21]. Unlike DSC, where each filter processes only one input channel as the first step, each LRC filter processes all the channels one after another. Further, in [17] it was observed that employing a DSC does not work well in the initial CNN layers, while we observed LRC does not have any such bottleneck. In the case of temporal first processing LRC requires $kNC + kMC$ parameters (excluding bias) while DSC requires $kNM + kMC$ parameters, thus DSC will have less trainable parameters than LRC only when $C > M$. We show that, in the context of speech signals, LRC performs better and is more robust than DSC. In addition, previous studies such as [22, 23] have shown that, for speech applications, combining information first along the spectral dimension is always better than temporal-first processing. In such cases, DSC is not applicable since it operates on the channels individually.

3.4 Comparison with Existing Approaches

Recently, many works have also explored low rank structures for compressing neural networks. Most of there works employ singular value decomposition (SVD) method where first a full-size model is trained and then the model size is reduced with SVD [24]. This approach has been mainly shown to be useful for fully connected layers [25, 26, 27]. One can see the SVD based approach as equivalent to inserting a linear bottleneck between two layers of the network. However, Xue et al. [24] confirmed that linear bottlenecks in all layers degrade performance when the network is trained from scratch. Recently, work in [28] verifies the strategy of SVD based training and proposed the mean-normalized SGD optimization method that enables the training from scratch with LR structure in FC layers. In contrast, our work presents an approach to implement LR decomposition in CNN layers as a part of the original structure of the network.

In the context of speech processing, modelling spectral and temporal dynamics of speech has been explored with time-frequency (TF)-LSTM [29, 30]. These systems involve performing 1-D recurrence over the frequency axis followed by recurrence over time axis, or jointly 2D TF modelling. [31] established an equivalence between F-LSTM and CNN. These works were originally carried out on spectrogram or hand crafted features. The present work however deals with modeling of raw waveform.

4 Experimental Setup

4.1 Dataset and Experimental Protocol

We present our case study for ASR systems, where experiments are conducted on TIMIT [32] and Mediaparl [33] corpora. TIMIT contains recordings of phonetically-balanced prompted English speech, while Mediaparl consists of datasets of parliamentary debates in two languages. Table 1 provides a detailed description of the datasets.

Table 1: Dataset information

	Mediaparl		TIMIT
	German	French	
Language	de-CH	fr-CH	en-US
Training hours	14.5	16.1	5.4
Phone set count	57	38	48
Vocabulary size	16.8k	12.4k	-

For TIMIT, we used the standard Kaldi recipe [34], where the pronunciation lexicon is a one-to-one map from each phone to itself, and the bigram phone language model is prepared from the training set. For Mediaparl experiments, we followed the protocols set in [33] for the data preparation, selection of pronunciation lexicon and building bi-gram language models (LM).

For each dataset, the HMM-GMM pipeline is according to Kaldi (mono-tri3) [34]. We additionally built subspace GMMs (SGMMs) and used their alignments to build all the neural networks. The Mediaparl setup

Table 2: CNN architecture for phone classification. The inputs to the network are speech signals of length 250ms. n_f denotes the number of filters in the convolution layer. nhu denotes the number of hidden units in the hidden layer. kW denotes kernel width. dW denotes kernel shift (stride). Mpool+ReLU refers to max pooling followed by ReLU activation.

Layer	kW	dW	n_f/nhu
Conv1	30	10	80
Mpool+ReLU	3	3	-
Conv2	7	1	60
Mpool+ReLU	3	3	-
Conv3	7	1	60
Mpool+ReLU	3	3	-
MLP	-	-	1024

Table 3: Convolutional layer parameter count.

Raw-CNN	**LR-CNN**	**LR-CNN2**	**DS-CNN**
61400	11960	21320	11980

consists of long utterances, and we found utterance-level cepstral mean and variance normalization (CMVN) to be better than that at speaker-level (as done in Kaldi), hence, the former is used in our experiments.

Table 4: WER/PER for ASR on various databases. DS-CNN denotes system based on depthwise separable convolution. LR-CNN denotes system based on the proposed low rank decomposition convolution.

	TIMIT English				**Mediaparl French**				**Mediaparl German**			
System	**Monophone**		**Triphone**		**Monophone**		**Triphone**		**Monophone**		**Triphone**	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
DNN	22.5	24.0	20.6	22.3	23.7	27.9	19.1	21.8	15.0	27.9	10.5	20.5
Raw-CNN	22.8	23.6	20.7	22.1	24.7	27.5	20.0	22.3	13.8	27.5	10.2	21.5
Raw-CNN + Init	-	-	20.4	21.8	-	-	20.0	22.3	-	-	10.2	21.5
DS-CNN	23.9	24.5	22.2	23.2	25.7	29.5	20.4	23.3	15.1	29.3	11.0	22.5
DS-CNN + Init	-	-	21.5	23.3	-	-	20.6	23.3	-	-	10.9	22.7
LR-CNN	23.1	24.4	21.8	23.4	24.3	27.4	20.0	22.3	14.7	27.8	10.7	21.4
LR-CNN + Init	-	-	20.7	22.8	-	-	19.7	21.7	-	-	10.7	21.2
LR-CNN2	22.4	24.1	21.3	22.8	23.8	26.9	19.2	21.8	13.8	27.4	10.1	20.9
LR-CNN2 + Init	-	-	19.9	22.3	-	-	19.7	21.7	-	-	9.8	20.9

4.2 Proposed and Baseline Systems

1) *DNN:* A hybrid DNN/HMM system, with a 3 hidden layer DNN, consisting of 1024 nodes with rectified linear activations in each layer, serves as our acoustic-feature based baseline system, using Keras [35] with a Tensorflow [36] backend. The system is built using a 13-D MFCC feature, extracted over 25ms frames at 100 Hz rate, along with frame splicing of 11 frames, and including delta and acceleration coefficients.
2) *Raw-CNN:* Our raw speech based CNN baseline is adapted from the existing work in [5]. The network architecture consists of a 3-layer CNN with 1 fully connected layer.
3) *DS-CNN:* This system uses the same configuration as Raw-CNN, with the standard convolution replaced by a depthwise separable convolution. The depth multiplier (dm) is set to 1, which is akin to rank-1 decomposition in case of the proposed approach.
4) *LR-CNN:* The proposed LRC based CNN mimics the baseline Raw-CNN configuration and hyperparemeters, except that each CNN filter is realized by rank-k filters as described in Section 3. In this work we consider the spectral-first processing configuration for the proposed CNN where the systems are denoted as LR-CNN ($k = 1$) and LR-CNN2 ($k = 2$).

Table 2 provides the baseline CNN configuration and the hyperparameter details, while Table 3 provides the number of trainable convolutional layer parameters. Both the DNN and CNNs were trained using Keras-Tensorflow framework, with stochastic gradient descent and cross-entropy loss. Learning rate was halved, in the range 10^{-1} to 10^{-6}, between successive epochs whenever the validation-loss was saturated. All network parameters were initialized from a normalized uniform distribution according to [37].

It is worth mentioning that the proposed LR-CNN was implemented using the available customisation layers in Keras, which is currently not optimised in terms of memory usage and speed. This limited us in further validating the approach on larger corpora as well as using increased number of convolution layer parameters.

4.3 Implementation

The proposed low-rank decomposition has been implemented using Keras/TensorFlow. For the ASR experiments, we interfaced this setup with Kaldi. These implementations, along with pre/post processing scripts, are available online[1].

5 Results

Table 4 presents the results obtained with the baseline and the proposed CNN-based systems, in terms of word error rate (WER) for Mediaparl databases and phoneme error rate (PER) for the TIMIT database. As reported in earlier studies, we observe that the performance of the Raw-CNN system is similar to or better than the feature based DNN system across databases. The proposed LR-CNN performs comparable to the Raw-CNN system. Although the results with DS-CNN are promising, the system significantly lags behind LR-CNN, even with more number of parameters. This verifies the claim that for speech signals decoupling convolutions across channels creates a performance bottleneck.

5.1 Effect of Rank

Rank-k is an important parameter in the proposed LR-CNN, and accounts for the number of subspaces used to approximate the original convolutional filter. As expected, increasing the rank from 1 to 2 gives a significant boost to the performance. Depending on the temporal context the performance saturates after a particular value of k. An equivalent configuration in DSC is with the parameter dm, where we compute multiple outputs for each input channel. Experimentally, for $dm = 2$ we observed a marginal improvement in the performance, yet the systems still lag behind LR-CNN.

5.2 Generalization for Context Dependent Modeling

Given a network architecture, a triphone system has significantly larger number of output phone states as compared to a monophone system. This may affect the convergence of the LR-CNN if trained from scratch. To investigate the generalization ability, we retrained the triphone systems with the filter weights initialized from the monophone system (denoted as LR-CNN + Init). As a comparison we also performed a similar experiment with the baseline and DS-CNN systems. It can be observed that, on TIMIT, there is a significant performance boost for LR-CNN to the extent that it matches the performance of Raw-CNN triphone system. However, DS-CNN does not seem to benefit from such an initialization, which again verifies the claim that the proposed approach has better generalization. On Mediaparl, such initialization only resulted in slight improvements but the proposed system achieved the best overall performance. This indicates that with more training data LR-CNN converges well and can even perform better than the baseline CNN.

6 Conclusion

Exploiting the idea that, when modeling raw speech signal with CNNs, the first layer of the CNN performs spectro-temporal filtering and the learned information resides in a low dimensional subspace, we proposed a low rank convolution based CNN architecture for speech modeling. The low rank decomposition decouples the short-term spectral envelope and the long term temporal modulation information present in the speech signals leading to parameter saving and better generalization. Our experimental studies also showed that the proposed LR-CNN approach is better than the existing depthwise separable convolution approach. The proposed LR-CNN approach first combines information across spectral frequencies or bands, and then models the temporal information. This is interesting, as conventional spectro-temporal modeling approaches, such as TRAPS [38, 39], advocate modeling temporal modulations in spectral bands first, followed by combining the information across the spectral bands. Our future work will build on these observations and study the LR-CNN approach in relation to the spectro-temporal modeling LSTM architectures developed in [31].

[1] https://github.com/idiap/LR-CNN

References

[1] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, “Deep neural networks for small footprint text-dependent speaker verification,” in *Proc. of ICASSP*, 2014.

[2] Ossama Abdel-Hamid, Abdel-Rahman Mohamed, Hui Jiang, and Gerald Penn, “Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition,” in *Proc. of ICASSP*, 2012.

[3] Chunlei Zhang and Kazuhito Koishida, “End-to-end text-independent speaker verification with triplet loss on short utterances,” *Proc. of Interspeech*, 2017.

[4] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: a large-scale speaker identification dataset,” in *Interspeech*, 2017.

[5] Dimitri Palaz, Ronan Collobert, and Mathew Magimai.-Doss, “Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks,” in *Proc. of Interspeech*, 2013.

[6] Tara N. Sainath, Ron J. Weiss, Andrew Senior, Kevin W. Wilson, and Oriol Vinyals, “Learning the speech front-end with raw waveform CLDNNs,” in *Proc. of Interspeech*, 2015.

[7] P. Golik, Z. Tüske, R. Schlüter, and H. Ney, “Convolutional neural networks for acoustic modeling of raw time signal in LVCSR,” in *Proc. of Interspeech*, 2015, pp. 26–30.

[8] George Trigeorgis, Fabien Ringeval, Raymond Brueckner, Erik Marchi, Mihalis A. Nicolaou, Björn W. Schuller, and Stefanos Zafeiriou, “Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network,” in *Proc. of ICASSP*, 2016.

[9] Rubén Zazo, Tara N. Sainath, Gabor Simko, and Carolina Parada, “Feature learning with raw-waveform CLDNNs for voice activity detection,” in *Proc. of Interspeech*, 2016.

[10] Heinrich Dinkel, Nanxin Chen, Yanmin Qian, and Kai Yu, “End-to-end spoofing detection with raw waveform CLDNNS,” in *Proc. of ICASSP*, 2017.

[11] Hannah Muckenhirn, Mathew Magimai.-Doss, and Sébastien Marcel, “End-to-end convolutional neural network-based voice presentation attack detection,” in *Proc. of Int. Joint Conf. on Biometrics*, 2017.

[12] Hannah Muckenhirn, Mathew Magimai.-Doss, and Sébastien Marcel, “Towards directly modeling raw speech signal for speaker verification using CNNs,” in *Proc. of ICASSP*, 2018.

[13] Dimitri Palaz, Mathew Magimai.-Doss, and Ronan Collobert, “Analysis of CNN-based speech recognition system using raw speech as input,” in *Proc. of Interspeech*, 2015.

[14] Selen Hande Kabil, Hannah Muckenhirn, and Mathew Magimai.-Doss, “On learning to identify genders from raw speech signal using CNNs,” in *Proc. of Interspeech*, 2018.

[15] Hannah Muckenhirn, Mathew Magimai.-Doss, and Sébastien Marcel, “On learning vocal tract system related speaker discriminative information from raw signal using CNNs,” in *Proc. of Interspeech*, 2018.

[16] Changliang Liu, Jinyu Li, and Yifan Gong, “Svd-based universal dnn modeling for multiple scenarios,” in *Proc. of Interspeech*, 2015.

[17] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 1800–1807.

[18] Andrew G Howard et al., “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.

[19] I. Freeman, L. Roese-Koerner, and A. Kummert, “Effnet: An efficient structure for convolutional neural networks,” in *Proc. of 25th IEEE International Conf. on Image Processing (ICIP)*, Oct 2018, pp. 6–10.

[20] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun, “ShuffleNet: An extremely efficient convolutional neural network for mobile devices,” in *Proc. of IEEE Conf. on CVPR*, June 2018.

[21] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proc. of IEEE Conf. CVPR*, June 2015, pp. 1–9.

[22] S. Thomas, S. Ganapathy, and H. Hermansky, “Phoneme recognition using spectral envelope and modulation frequency features,” in *Proc. of ICASSP*, April 2009, pp. 4453–4456.

[23] Sriram Ganapathy, Samuel Thomas, and Hynek Hermansky, “Modulation frequency features for phoneme recognition in noisy speech,” *The Journal of the Acoustical Society of America*, vol. 125, no. 1, pp. EL8–EL12, 2009.

[24] Jian Xue, Jinyu Li, and Yifan Gong, “Restructuring of deep neural network acoustic models with singular value decomposition,” in *Proc. of Interspeech*, 2013.

[25] Zichao Yang, Marcin Moczulski, Misha Denil, Nando de Freitas, Alex Smola, Le Song, and Ziyu Wang, “Deep fried convnets,” in *The IEEE Int. Conference on Computer Vision (ICCV)*, December 2015.

[26] J. Xue, J. Li, D. Yu, M. Seltzer, and Y. Gong, “Singular value decomposition based low-footprint speaker adaptation and personalization for deep neural network,” in *Proc. of ICASSP*, May 2014, pp. 6359–6363.

[27] Yifan Sun, Liang Zheng, Weijian Deng, and Shengjin Wang, “Svdnet for pedestrian retrieval,” in *The IEEE Int. Conference on Computer Vision (ICCV)*, October 2017.

[28] S. Wiesler, A. Richard, R. Schlüter, and H. Ney, “Mean-normalized stochastic gradient for large-scale deep learning,” in *Proc. of ICASSP*, May 2014, pp. 180–184.

[29] J. Li, A. Mohamed, G. Zweig, and Y. Gong, “Lstm time and frequency recurrence for automatic speech recognition,” in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Dec 2015, pp. 187–191.

[30] J. Li, A. Mohamed, G. Zweig, and Y. Gong, “Exploring multidimensional lstms for large vocabulary asr,” in *Proc. of ICASSP*, March 2016, pp. 4940–4944.

[31] Tara N. Sainath and Bo Li, “Modeling time-frequency patterns with LSTM vs. convolutional architectures for LVCSR tasks,” in *Proc. of Interspeech*, 2016.

[32] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, “DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1,” *NASA STI/Recon Technical Report N*, vol. 93, 1993.

[33] David Imseng et al., “Mediaparl: Bilingual mixed language accented speech database,” in *Proc. of Spoken Language Technology (SLT) workshop*, Dec 2012, pp. 263–268.

[34] Daniel Povey et al., “The Kaldi speech recognition toolkit,” in *Proc. of IEEE Workshop on ASRU*, 2011.

[35] François Chollet et al., “Keras,” `https://github.com/fchollet/keras`, 2015.

[36] Martín Abadi et al., “TensorFlow: Large-scale machine learning on heterogeneous systems,” `http://tensorflow.org/`, 2015.

[37] Xavier Glorot and Yoshua Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proc. of the Thirteenth Int. Conf. on Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[38] Hynek Hermansky and Sangita Sharma, “TRAPS - classifiers of temporal patterns,” in *Proc. of ICSLP*, 1998.

[39] Barry Chen, Shuangyu Chang, and Sunil Sivadas, “Learning discriminative temporal patterns in speech: Development of novel traps-like classifiers,” in *Proc. of Eurospeech*, 2003.