# DEEP SCATTERING SPECTRUM WITH DEEP NEURAL NETWORKS

*Vijayaditya Peddinti*[†*], *Tara N. Sainath*[‡], *Shay Maymon*[‡]
*Bhuvana Ramabhadran*[‡], *David Nahamoo*[‡], *Vaibhava Goel*[‡]

†Center for Language and Speech processing, Johns Hopkins University, MD 21218, USA
‡ IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
vijay.p@jhu.edu, {tsainath,maymon,bhuvana,nahamoo,vgoel}@us.ibm.com

## ABSTRACT

State-of-the-art convolutional neural networks (CNNs) typically use a log-mel spectral representation of the speech signal. However, this representation is limited by the spectro-temporal resolution afforded by log-mel filter-banks. A novel technique known as Deep Scattering Spectrum (DSS) addresses this limitation and preserves higher resolution information, while ensuring time warp stability, through the cascaded application of the wavelet-modulus operator. The first order scatter is equivalent to log-mel features and standard CNN modeling techniques can directly be used with these features. However the higher order scatter, which preserves the higher resolution information, presents new challenges in modelling. This paper explores how to effectively use DSS features with CNN acoustic models. Specifically, we identify the effective normalization, neural network topology and regularization techniques to effectively model higher order scatter. The use of these higher order scatter features, in conjunction with CNNs, results in relative improvement of 7% compared to log-mel features on TIMIT, providing a phonetic error rate (PER) of **17.4%**, one of the lowest reported PERs to date on this task.

***Index Terms***— deep scattering spectrum, neural networks

## 1. INTRODUCTION

Learning representations optimal for a given classification task, in conjunction with the classifier is an attractive proposition, as all intermediate processing stages from the raw signal up to the classification are learned to minimize the task specific objective function [1]. However representation learning from the raw speech signal, while making minimal assumptions, has proven to be challenging in acoustic modeling. For example, Palaz *et al* [2] showed that learning filters from a raw-signal jointly within a neural network framework was slightly worse than feeding log-mel spectra as input into the network. Since using lower-level representations of the signal can be challenging, to date state-of-art acoustic modeling techniques use a higher level representation of speech signal, like log-mel spectra.

Design of these higher level feature representations is done to satisfy the goals of preserving detail in the signal, necessary for classification, while remaining invariant/stable to non-informative distortions. A key step for representing speech in a stable fashion is to focus on elements of the signal that are important for speech recognition. This helps reduce the variance in the representation, due to non-informative elements and channel distortions [3]. Conventional feature extraction techniques like MFCC [4], PLP [5] and RASTA [6] operate on this principle. However, these features are often designed with an invariance in mind, and without explicit knowledge of the classifier objective. This might remove potentially important information.

For example, the sub-10ms temporal dynamics of speech, essential to capture transient phenomena and finer modulation structure of the speech sounds, are not captured by the log-mel spectra [7]. Even though better estimation techniques can be designed to preserve higher resolution detail [8], (e.g. using auto-regressive modeling techniques), even these high resolution representations are processed using short term smoothing operators for deformation stability [9]. In short, designing a representation to both, preserve the relevant detail in the speech signal and provide stability/invariance to distortions, is a non-trivial task.

To over come these limitations there has been recent effort in learning invariance transforms from lower level representations, while improving over the performance of the higher level representations. For example, [1] looked at starting with a raw acoustic representation of the signal (i.e., power spectrum), and learning the filter banks in a neural network framework. However, even this work used the power-spectra as an input representation, which is computed from a fixed window-length, which thus also removes information from the signal.

Deep scattering networks (DSN)[10] have recently been introduced to address some of the above challenges. DSNs take *a raw-signal* and generate a contractive representation, which preserves signal energy, while ensuring Lipschitz continuity to deformations ( [11]and [12]). A scattering representation includes log-mel like measurements (first-order scatter) together with higher-order co-occurence coefficients that can preserve greater detail in the speech signal [13]. Second order scatter coefficients, for example, preserve the higher resolution information in the signal, such as transient phenomena or amplitude modulations. The representation generated by these networks, called Deep Scattering Spectrum (DSS), is locally translation invariant representation and stable to time varying deformations [13]. Thus, a benefit of the DSS is that it allows us to start from the raw-signal and provides a set of wavelet-modulus transforms which try to minimize loss of information in the input signal. This is very different than conventional speech representations such as log-mel.

In this paper, we explore using DSS features for acoustic modeling. As deep neural networks (DNNs) are now considered state-of-the-art in acoustic modeling [14], we explore how to use DSS features in an effective manner with DNNs, which to date has not been done. Within this, we discuss the challenges in DNN-based acoustic

[*] This author performed the work while at IBM T.J. Watson Research Center

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

modeling using the DSS. This includes exploring how to effectively combine first-order scatter and higher scatter coefficients, how the features should be normalized within the DNN without losing information, and how the network should be regularized to handle these features.

For the purposes of this paper, we create DSS features ahead of time, and then feed this into a DNN. However, the DSS coefficients are computed by cascading wavelet filter banks and followed by a modulus non-linearity, which can be interpreted as a convolutional neural network. Thus learning the DSN and DNN parameters jointly from the raw signal is an obvious next step.

The rest of this paper is organized as follows. Section 2 describes the DSS computation process. It introduces normalization technique called scattering transfer, which is used in conjunction with this representation. Section 3 presents pre-processing techniques for higher order scatter coefficients, network architecture for combining first-order and higher-order scatter using DNNs. Sections 4 and 5 describes the multi-resolution filter-bank approach for extracting higher order scatter at various resolutions which helps in achieving state-of-art TIMIT phoneme recognition results. Finally, Section 6 concludes the paper and discusses future work.

## 2. SCATTERING TRANSFORM

In this section we briefly describe the Deep Scattering Spectrum (DSS) representation for speech signals, introduced in [13]. Joakim et al [13] have shown that the mel-frequency spectrogram (or log-mel) coefficients, computed as frequency average of the linear frequency spectrogram using mel-windows, can be approximated by the time average of the demodulated sub-band signals, extracted using a wavelet filter-bank ($\psi_{\lambda_1}$). The operations of demodulation, accomplished using a modulus($|.|$), and time averaging, performed by low-pass filtering ($\phi(t)$), emulate an amplitude demodulator. To generate the logmel representation, time support of this averaging filter $\phi(t)$ is chosen to be $\sim 25$ ms and $\psi_{\lambda_1}$ is chosen to be a constant-Q filter-bank with Q=8.

Time averaging provides features which are locally invariant to small translations and are stable to distortions. However this averaging operation leads to loss of information of interest, which in speech signals corresponds to the transient phenomena or finer amplitude modulations. To recover this lost information another decomposition of the sub-band signals is performed using a second wavelet filter-bank($\psi_{\lambda_2}$). This second decomposition captures the information in the sub-band signal, $|x * \psi_{\lambda_1}|$, left out by the averaging filter $\phi(t)$. The decomposed sub-band signals $|x * \psi_{\lambda_1}| * \psi_{\lambda_2}$, are once again passed through the amplitude demodulator ($||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi(t)$) to extract stable features. The information left out by this second averaging operation can be once again be isolated using a third decomposition and so on.

Thus using a cascade of these decomposition and modulus operations, referred to as the wavelet-modulus operator, the higher resolution detail in the signal can be preserved, to the extent desired for any speech task. It is important to note that though this high resolution information is preserved, the representation is still locally deformation stable, to the extent determined by the averaging filter $\phi(t)$.

For speech signals the first order scatter ($|x * \psi_{\lambda_1}| * \phi(t)$), represented as $S_1x(t, \lambda_1)$, is the logmel equivalent and second order scatter ($||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi(t)$), represented as $S_2x(t, \lambda_1, \lambda_2)$, is the amplitude modulation spectrogram equivalent. The second order scatter is computed using a constant-Q filter-bank with $Q = 1$. Each of the decompositions $||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi(t)$, has limited number of non-zero coefficients, due to the band-limited nature of the signals $|x * \psi_{\lambda_1}|$.

To ensure that the higher order scatter just depend on the amplitude modulation component of the speech signal, [13] suggested the use of the "scatter transfer" operator, which is effectively the normalization of the higher order scatter by the lower order scatter. The scatter transfer operation, defined by Equation 1, is applied to the higher order scatter.

$$T(t, \lambda_1, \lambda_2) = \frac{S_2x(t, \lambda_1, \lambda_2)}{S_1x(t, \lambda_1)} \quad (1)$$

For the sake of brevity, in this paper we represent the first order scatter $S_1x(t, \lambda_1)$ with $S_1$ and the normalized second order scatter, given in Equation 1, with $S_2$.

## 3. MODELING DEEP SCATTER SPECTRUM

In this section, we present a detailed analysis and optimal recipe for modeling scatter coefficients using neural networks. These are broken down into the broad categories of feature representation, normalization, network architecture and non-linearity.

### 3.1. Experiment Design

All results presented in this section are evaluations on the TIMIT phoneme recognition task [15]. The baseline speaker-independent CNN system is trained with 40 dimensional log mel-filter bank coefficients including their $\Delta$ and $\Delta\Delta$ coefficients. The architecture of the CNN is similar to [16], which was found to be optimal for speech tasks. Specifically, the CNN has 2 full weight sharing convolutional layers with 256 hidden units, and 3 fully connected layers with 1,024 hidden units per layer. Our experiments explore using a context-independent tree of 147 output targets, as well as a context-dependent tree of 2400 output targets [17].

### 3.2. Normalization

Feature normalization is critical in neural network training to achieve good convergence in training. As discussed in [18], when features are not centered around zero, network updates will be biased towards a particular direction and this will slow down learning. The paper even discusses an extreme case when all inputs into a layer are positive. This causes all weights to increase or decrease together for a given input, and thus the weight vector can only change direction by zigzagging which is extremely slow.

In our paper, we perform a straightforward mean and variance normalization of $S_1$ features, similar to log-mel features. However, [13] performed a "scatter transfer" operation, shown in Equation 1, which was found to be effective for $S_2$. We use this normalization scheme for higher-order scatter and apply a mean-only normalization on top of this, as preliminary experiments showed that applying variance normalization after *scatter transfer* operation was not optimal.

### 3.3. Feature Representation

The first order scatter coefficients ($S_1$) are similar to the log-mel coefficients. Hence the standard $\Delta$ and $\Delta\Delta$ features were computed for this feature stream. To verify the equivalence of $S_1$ and log-mel features, we trained a CNN with both feature sets. Table 1 shows that the phonetic error rate (PER) for both feature sets using context-independent (CI) and context-dependent (CD) state targets,

is roughly equivalent, with small differences likely due to the randomness of training batches [18] during CNN training. Note that rectified linear units (ReLUs) [19] are used for these experiments.

| Feature | PER | |
|---|---|---|
| | CI | CD |
| $logmel + \Delta + \Delta\Delta$ | 19.3 | 18.7 |
| $S_1x(t, \lambda_1) + \Delta + \Delta\Delta$ | 19.0 | 18.7 |

**Table 1**. Comparison of $S_1x(t, \lambda_1)$ and logmel

$\Delta$ and $\Delta\Delta$ coefficients were not computed for the higher order scatter features $S_2$. On inspection, it was noted that the scatter transfer operation, described above in Equation 1, also enhances the transitions in the higher order scatter coefficient trajectories, which is similar to the functionality of the $\Delta$ features. The application of the $\Delta$ filter on top of $S_2$ increases the variance in the representation, and was not found to be beneficial.

### 3.4. Network Architecture

CNNs reduce spectral variations and model spectral correlations which exist in speech signals and provide substantial gains in acoustic modeling compared to DNNs on both small and large vocabulary tasks ([16],[20]). The conventional spectro-temporal representation $S_1$ (or log-mel) preserves locality in both time and frequency, and it can be directly used as an input to the CNN, along with the $\Delta$ and $\Delta\Delta$ channels.

The second order scatter, $S_2x(t, \lambda_1, \lambda_2)$, which is the decomposition of amplitude modulations in each sub-band of the first-order filter-bank ($|x * \psi_{\lambda_1}|$), preserves the locality of information, for a given sub-band $\lambda_1$. Thus it can be used as the input to a CNN. However each of these sub-band decompositions has limited number of non-zero coefficients (see Section 2). As a result the filter size in the CNN, that can be meaningfully chosen, is small. Thus, a DNN was chosen to process the higher order scatter.

Therefore, the first order scatter is processed using convolutional layer(s), while the second (or higher) order scatter is processed using fully connected layer(s). Hence a CNN/DNN combination network was used to process these features, which has been explored before for combining multiple feature streams [21, 22]. The architecture of this network is shown in Figure 1. With this architecture, we initially explore using 1,024 hidden units for the first DNN layer, similar to all other fully connected DNN layers.

Table 2 compares the performance of first and second-order scatter features using DNNs and the joint CNN/DNN architecture. For these experiments, the sigmoid non-linearity was used instead of ReLU. The joint architecture offers improvements over just the DNN, and so we adopt this architecture for subsequent experiments. It should be noted that the PER increases for the DNNs when going from $S_1$ to $S_2$, and one possible explanation is that these feature streams use different normalization schemes so their dynamic ranges are different.

| Feature | DNN | CNN (for $S_1$)+DNN (for $S_2$) |
|---|---|---|
| $S_1 + \Delta + \Delta\Delta$ | 23.9 | 21.3 |
| $S_1 + \Delta + \Delta\Delta + S_2$ | 24.2 | 20.9 |

**Table 2**. Comparison of DNN and CNN-DNN architecture on CI phoneme recognition

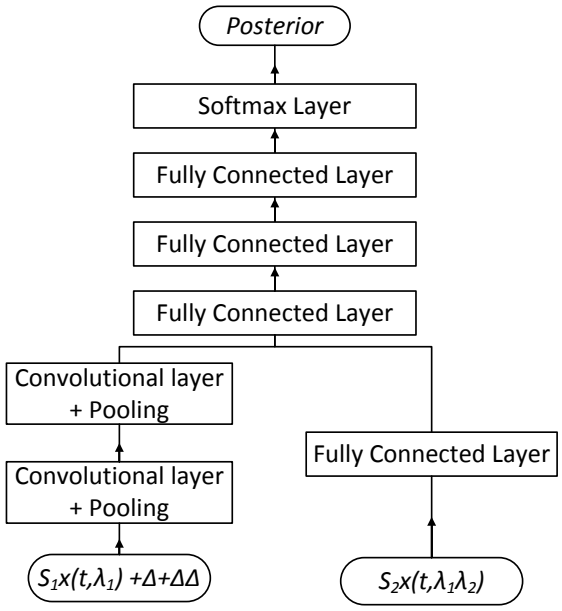


**Fig. 1**. CNN/DNN architecture

### 3.5. Non-linearity

Rectified linear units (ReLUs), $f(u) = max(0, u)$, have been shown to be suitable non-linearity for learning in DNNs due to properties of better generalization, faster convergence, easier optimization and faster computation [19]. They are particularly useful if the input features are not normalized to unit-variance, as ReLUs are unbounded along the positive axis. Since the higher-order scatter features used the "scatter transfer" normalization, we explore if ReLUs are beneficial for these features. Because ReLUs are not bounded, we also explore using additional regularization schemes such as maxnorm ([23]) and dropout ([24]). Table 3 indicates that the PER improves for both first and second order scattering features by using ReLUs, with additional improvements coming from including regularization.

| Non-linearity | $S_1 + \Delta + \Delta\Delta$ | $S_1 + \Delta + \Delta\Delta + S_2$ |
|---|---|---|
| Sigmoid | 21.3 | 20.9 |
| ReLU | 20.0 | 20.3 |
| ReLU+regularization | 19.0 | 18.8 |

**Table 3**. Comparison of Non-linearities on CI phoneme recognition

## 4. MULTI-RESOLUTION SCATTERING FEATURES

Multi-stream processing of speech at various spectral and temporal resolutions derives its inspiration from neurophysiological evidence of parallel processing streams ([25],[26]). These parallel streams capture spectral and temporal dynamics of the signal at various resolutions. Furthermore, in [13], it was shown that using multi-resolution scattering features improved performance over just a single resolution on the TIMIT classification task.

The scattering representation was used to separate the temporal dynamic information among scatter features of various orders. To capture spectral dynamics at various resolutions filter-banks with different spectral resolutions (Q=13,Q=4,Q=1) denoted by $\psi_{\lambda_a}$,$\psi_{\lambda_b}$ and $\psi_{\lambda_c}$ were used to compute the first decomposition in the scatter representation. The second decomposition in the cascade was performed with a single filter-bank (Q=1),$\psi_{\lambda_d}$. This decomposition results in 6 different feature streams $S_1x(t,\lambda_a)$, $S_1x(t,\lambda_b)$, $S_1x(t,\lambda_c)$, $S_2x(t,\lambda_a,\lambda_d)$, $S_2x(t,\lambda_b,\lambda_d)$ and $S_2x(t,\lambda_c,\lambda_d)$.

The three first order scatter feature streams $\{S_1x(t,\lambda_a)$, $S_1x(t,\lambda_b)$, $S_1x(t,\lambda_c)\}$, represented by $G_1$, along with $\Delta$ and $\Delta\Delta$, were processed through three different convolutional layers. The second order scatter $\{S_2x(t,\lambda_a,\lambda_d)$, $S_2x(t,\lambda_a,\lambda_d)$, $S_2x(t,\lambda_c,\lambda_d)\}$, represented by $G_2$, was processed through a single fully-connected DNN layer. In addition to this, another network was trained with just $G_1+\Delta+\Delta\Delta$, to provide a baseline for the multi-resolution feature stream systems.

For all the below experiments the neural network training recipe used ReLU non-linearity with max-norm regularization. Table 4 shows the PER for multi-resolution processing. First, notice that by using the three stream $G_1+\Delta+\Delta\Delta$ compared to the single stream $S_1+\Delta+\Delta\Delta$, the PER reduces from 19.0% to 18.4%. However, when including multi-resolution processing of first and second order scattering ($G_1$+$\Delta$+$\Delta\Delta$ +$G_2$), the PER is at 19.1% which is higher than the single stream feature of $S_1$+$\Delta$+$\Delta\Delta$+$S_2$ at 18.8%.

A closer look shows that the dimension of $G_2$ is 270, and this is input with a context of 11 into the first DNN layer shown in Figure 1, which has 1,024 hidden units. This accounts for roughly 3 million parameters in just this single DNN layer alone, which is roughly 20% of the total number of parameters of the network. We hypothesize that the network could be over fitting, and thus explore reducing the number of hidden units for the DNN layer. Table 4 shows that by reducing the number of hidden units and thus overall number of parameters, the PER for multi-resolution $G_1+\Delta+\Delta\Delta+G_2$ improves. Overall, we see that multi-resolution scatter at a PER of 18.2% offers a 4% relative improvement over the first-order scatter (i.e., log-mel) at a PER of 19.0%.

| Feature Stream | PER |
|---|---|
| $S_1$+$\Delta$+$\Delta\Delta$ | 19.0 |
| $G_1$+$\Delta$+$\Delta\Delta$ | 18.4 |
| $S_1$+$\Delta$+$\Delta\Delta$+$S_2$ | 18.8 |
| $G_1$+$\Delta$+$\Delta\Delta$ +$G_2$+1024 HU | 19.1 |
| $G_1$+$\Delta$+$\Delta\Delta$ +$G_2$+256 HU | 18.7 |
| $G_1$+$\Delta$+$\Delta\Delta$ +$G_2$+128 HU | **18.2** |
| $G_1$+$\Delta$+$\Delta\Delta$ +$G_2$+ 64 HU | 18.6 |

[a] HU - Hidden Units

**Table 4**. Comparison of multi-resolution features on CI phoneme recognition

## 5. SCATTERING FEATURES WITH CONTEXT-DEPENDENT STATES

In [27] it was observed that performance of CNNs in TIMIT improved by using context-dependent states rather than context-independent states. We explore the behavior of scattering features using 2,400 CD-state output targets [17].

Table 5 first shows that using first and second order scatter, single stream ($S_1$+$\Delta$+$\Delta\Delta$+$S2$) has a PER of 17.9%, and offers an improvement over first-order scatter alone ($S_1$+$\Delta$+$\Delta\Delta$) at a PER of 18.7%. Furthermore, by including multi-resolution first and second order scatter, we achieve a PER of 17.4%, which is a 7% relative improvement over the first-order scatter (i.e., log-mel) at a PER of 18.7%. To our knowledge the PER of 17.4% is one of the lowest reported on TIMIT to date, compared to the previously best reported number of 17.7% [28] using RNNs and 17.8% using CNNs [27]. A variant of the multi-resolution scattering features were used to achieve the lowest error rate of 15.8% for TIMIT phoneme classification by Joakim *et al* [13].

| Feature Stream | PER |
|---|---|
| $S_1$+$\Delta$+$\Delta\Delta$ | 18.7 |
| $S_1$+$\Delta$+$\Delta\Delta$+$S_2$ 128 HU | 17.9 |
| $G_1$+$\Delta$+$\Delta\Delta$ +$G_2$+128 HU | **17.4** |

[a] HU - Hidden Units

**Table 5**. Comparison of features on CD phoneme recognition

## 6. CONCLUSIONS AND FUTURE WORK

In this paper we showed that using higher resolution detail, in the form of deep scatter spectra, is useful for acoustic modeling. We detailed the challenges in modeling higher order spectra, in combination with neural networks. We identified the appropriate network topology and recipe for effectively training networks with scattering features. On TIMIT, we found that the scattering features provided a PER of 18.2% on a CI system, a 4% relative improvement over the baseline log-mel system at 19.0%. In addition, using a CD system, scattering features provides a PER of 17.4%, a 7% relative improvement over the baseline log-mel system at 18.7%.

In the future, we plan to extend this work in many directions. First, it would be interesting to explore the behavior of scattering transforms on LVCSR tasks where the end goal is word recognition. We suspect that in tasks such word recognition, where the interest is in word events which occur at a longer time scale, averaging filters with a larger time span can be applied. These help in achieving invariance/stability to distortions over longer durations. Furthermore, learning the scattering transforms jointly with the rest of the CNN, motivated by work in [1] would also be interesting.

Further an extensive comparison of the Deep scatter spectrum features with TRAPs, MRASTA and other modulation spectra based features has to be done.

## 7. ACKNOWLEDGEMENTS

The authors would like to thank Joakim Anden and Joan Bruna for helpful discussions related to scattering transforms. Also, thank you to Hagen Soltau for contributions towards the joint CNN/DNN architecture.

## 8. REFERENCES

[1] T. N. Sainath, B. Kingsbury, A. Mohamed, and B. Ramabhadran, "Learning Filter Banks Within a Deep Neural Network Framework," in *Proc. of ASRU*, 2013.

[2] D. Palaz, R. Collobert, and M. Magimai.-Doss, "Estimating phoneme class conditional probabilities from raw speech sig-

nal using convolutional neural networks," in *Proc. Interspeech*, 2013, pp. p1–p2.

[3] S. Greenberg and B. Kingsbury, "The modulation spectrogram: In pursuit of an invariant representation of speech," in *In ICASSP*. 1997, pp. 1647–1650, IEEE.

[4] S. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences ," *IEEE Transacations on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357 – 366, 1980.

[5] H. Hermansky, "Perceptual linear predictive (plp) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, pp. 1738, 1990.

[6] H. Hermansky and N. Morgan, "RASTA Processing of Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 578 – 589, 1994.

[7] M. Athineos and D. Ellis, "Frequency-domain linear prediction for temporal features," in *Proc. of ASRU*. IEEE, 2003, pp. 261–266.

[8] M. Athineos and D. Ellis, "Autoregressive modeling of temporal envelopes," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5237–5245, 2007.

[9] S. Ganapthy, *Signal Analysis using Autoregressive Models of Amplitude Modulation*, Ph.D. thesis, Johns Hopkins University, 7 2012.

[10] S. Mallat, "Group invariant scattering," *Communications on Pure and Applied Mathematics*, vol. 65, no. 10, pp. 1331–1398, 2012.

[11] S. Mallat, "Deep learning by scattering," *CoRR*, vol. abs/1306.5532, 2013.

[12] J. Bruna, S. Mallat, et al., "Invariant scattering convolution networks.," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1872–1886, 2013.

[13] J. Andén and S. Mallat, "Deep scattering spectrum," *Submitted to IEEE Transactions on Signal Processing*, 2013.

[14] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[15] L. Lamel, R. Kassel, and S. Seneff, "Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus," in *Proc. of the DARPA Speech Recognition Workshop*, 1986.

[16] T. N. Sainath, A. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep Convolutional Neural Networks for LVCSR," in *Proc. ICASSP*, 2013.

[17] T. N. Sainath, B. Ramabhadran, M. Picheny, D. Nahamoo, and D. Kanevsky, "Exemplar-Based Sparse Representation Features: From TIMIT to LVCSR," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 8, pp. 2598–2613, 2011.

[18] Y. LeCun, L. Bottou, G. Orr, and K. Muller, "Efficient Backprop," in *Neural Networks: Tricks of the Trade*, G. Orr and Muller K., Eds. 1998, Springer.

[19] V. Nair and G. Hinton, "Rectified Linear Units improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[20] O. Abdel-Hamid, A. Mohamed, J. Hui, and G. Penn, "Applying Convolutional Neural Networks concepts to hybrid NN-HMM model for speech recognition," in *Proc. ICASSP*, 2012, pp. 4277–4280.

[21] T. N. Sainath, B. Kingsbury, A. Mohamed, G. Dahl, G. Saon, H. Soltau, T. Beran, A. Y. Aravkin, and B. Ramabhadran, "Improvements to Deep Convolutional Neural Networks for LVCSR," in *Proc. ASRU*, 2013.

[22] H. Soltau, G. Saon, and T. N. Sainath, "Joint Training of Convolutional and Non-Convolutional Neural Networks," in *Proc. ICASSP*, 2014.

[23] N. Srebro and A. Shraibman, "Rank, trace-norm and max-norm," in *Proceedings of the 18th annual conference on Learning Theory*. Springer-Verlag, 2005, pp. 545–560.

[24] G. Dahl, T.N. Sainath, and G. Hinton, "Improving Deep Neural Networks for LVCSR using Rectified Linear Units and Dropout," in *Proc. ICASSP*, 2013.

[25] S.K. Nemala, K. Patil, and M. Elhilali, "A multistream feature framework based on bandpass modulation filtering for robust speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 2, pp. 416–426, 2013.

[26] H. Hermansky and P. Fousek, "Multiresolution RASTA filtering for TANDEM-based ASR," in *Proc. of Interspeech 2005*, 2005, pp. 361–364.

[27] L. Toth, "Convolutional Deep Rectifier Neural Nets for Phone Recognition," in *Proc. Interspeech*, 2013.

[28] A. Graves, A. Mohamed, and G. Hinton, "Speech Recognition with Deep Recurrent Neural Networks.," in *Proc. ICASSP*, 2013.