Novel Unsupervised Auditory Filterbank Learning Using Convolutional RBM for Speech Recognition

Hardik B. Sailor, *Student Member, IEEE*, and Hemant A. Patil, *Member, IEEE*

***Abstract*—To learn auditory filterbanks, recently, we have proposed an unsupervised learning model based on convolutional restricted Boltzmann machine (RBM) with rectified linear units. In this paper, theory, training algorithm of our proposed model, and detailed analysis of learned filterbank are being presented. Learning of the model with different databases shows that the model is able to learn cochlear-like impulse responses that are localized in frequency-domain. An auditory-like scale obtained from filterbanks learned from clean and noisy datasets resembles the Mel scale, which is known to mimic perceptually relevant aspect of speech. We have experimented with both cepstral (denoted as ConvRBM-CC) as well as filterbank features (denoted as ConvRBM-BANK). On large vocabulary continuous speech recognition task, we achieved relative improvement of 7.21–17.8% in word error rate (WER) compared to Mel frequency cepstral coefficient (MFCC) features and 1.35–6.82% compared to Mel filterbank (FBANK) features. On AURORA 4 multicondition training database, the relative improvement in WER by 4.8–13.65% was achieved using a Hybrid Deep Neural Network-Hidden Markov Model (DNN-HMM) system with ConvRBM-CC features. Using ConvRBM-BANK features, we achieve absolute reduction of 1.25–3.85% in WER on AURORA 4 test sets compared to FBANK features. A context-dependent DNN-HMM system further improves performance with a relative improvement of 3.6–4.6% on an average for bigram 5k and tri-gram 5k language models. Hence, our proposed learned filterbank performs better than traditional MFCC and Mel-filterbank features for both clean and multicondition automatic speech recognition (ASR) tasks. A system combination of ConvRBM-BANK and FBANK features further improve performance in all ASR tasks. Cross-domain experiments where subband filters trained on one database are used for the ASR task of another database show that model learns generalized representations of speech signals.**

***Index Terms*—Auditory processing, ConvRBM, filterbank, subband filters, speech recognition.**

I. Introduction

REPRESENTATION of a speech signal based on human speech perception is of significant interest in developing features for speech processing applications. To mimic the human auditory processing, classical auditory models were developed during the 1980s. As reviewed in [1], Seneff, Ghitza, and Lyon's auditory models have made a huge impact on many recent computational models. These auditory models are based on mathematical modeling of auditory processing or psychophysical and physiological experiments. Mel Frequency Cepstral Coefficients (MFCC) and Perceptual Linear Prediction (PLP) coefficients are state-of-the-art auditory-based features for speech recognition. Such hand-crafted features rely on simplified auditory models [1], [2]. However, such physiological models often do not reflect the full complexity of human auditory system (HAS) which, for example, is able to adapt readily to much variability in acoustic conditions. There are many approaches that are based on data-driven learning and/or optimization of parameters of auditory models. Data-driven learning or representation learning can be supervised (i.e., with label information) or unsupervised (where no such labels are available for each class).

Recently, representation learning has gained significant interest for feature learning in various signal processing areas including speech processing [3]. As discussed in [4], features for human speech perception, vision and other cognition tasks are learned from experience, simulating human learning as we grow. Supervised learning approaches for raw speech signal include work in [5]–[11] which are end-to-end approaches for acoustic modeling in Automatic Speech Recognition (ASR). Unsupervised learning is one of the important forms of representation learning since many human learning tasks are unsupervised [12]. For example, we listen to many sounds as we grow and we are usually not told every time about the type of sound, speech and their sources (such as speaker aspects, e.g., gender, age). Another example is language acquisition by infants during initial stages of their growth, which is also a type of unsupervised learning. Most work on unsupervised learning for speech signals is based on cochlear filterbank learning to model auditory processing. The first approach was to use Independent Component Analysis (ICA) as a learning model applied on small windows of speech signals [13]–[15]. To model a Mel-like filterbank, Nonnegative Matrix Factorisation (NMF) was applied to power spectra of speech signals [16]. In [17], nonlinearity associated to the auditory system is optimized using a data-driven method. Based on local geometries of the feature vector-domain and the perceptual auditory-domain, MFCC features were optimized in [18]. Restricted Boltzmann Machine (RBM) with Rectified Linear Units (ReLU) was also used to learn features using segments of raw speech signals [19].

The unsupervised learning methods described above are based on processing small segments of speech signals or operating on Short-Time Fourier Transform (STFT) of speech signals.

Manuscript received February 5, 2016; revised July 16, 2016; accepted August 21, 2016. Date of publication September 8, 2016; date of current version September 27, 2016. This work was supported in part by the Department of Electronics and Information Technology, Government of India, through two consortium projects, TTS Phase-II and ASR Phase-II, and in part by the authorities of DA-IICT, Gandhinagar, India. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mohamed Afify.

The authors are with Speech Research Lab, Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT), Gandhinagar 382007, Gujarat, India (e-mail: sailor_hardik@daiict.ac.in; hemant_patil@daiict.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2016.2607341

However, there are many disadvantages of such block-based (or window-based) speech signal processing as discussed in [20]. In particular, speech signal has very brief transient sounds as well as quasi-periodic voiced sounds; fixed window segments of speech may smear these sounds. In addition, representation is very sensitive to temporal shifts of windows as experimentally proved in [20]. To avoid these problems, sparse spike-coding is used to learn filterbank from speech signals [20], [21]. However, spike-coding does not include any *nonlinearity* in the model and optimization is performed on the linear superposition of kernel functions [20]. To obtain MFCC-like representation using time-domain formulation, scattering transform was proposed [22]. Scattering wavelets on Mel-scale is convolved with speech signals and averaged later using lowpass filter. However, study reported in [22] does not involve learning of filters.

To alleviate the problems discussed above, recently we have proposed unsupervised filterbank learning model which is shown to perform better than MFCC and Mel filterbank (denoted as FBANK) features for speech recognition task [23]. Novelty of the proposed model lies in learning directly from the speech signals of any *arbitrary* lengths in order to alleviate artifacts of windowing. In addition, it includes nonlinearity in learning and model is stochastic in nature. Our proposed model is based on Convolutional Restricted Boltzmann Machine (ConvRBM) which was proposed in [24] to improve the scalability of RBM. Earlier ConvRBM was applied on spectrograms of speech signals to model Temporal Receptive Fields (TRFs) in auditory cortex [25]. In [26], we have introduced ReLU in ConvRBM to learn TRFs. We have developed ConvRBM to model auditory processing in human ear using the raw speech signals. We have used ReLU to increase the sparsity and inference is based on noisy ReLU (NReLU). Compared to recent approaches for filterbank learning in convolutional networks [8]–[10], our model is unsupervised and probabilistic in nature.

This paper is an extension of our recent work reported in [23] in terms of including more detailed theory, learning algorithm, additional analysis and further experimentations on new datasets. In particular, we have discussed in more detail of our proposed model architecture and training method. Analysis of filterbanks is extended in detail to study how model learns filterbank for various datasets compared to standard filterbanks. In addition to our recent work [23], this paper extends experiments on the WSJ Large Vocabulary Continuous Speech Recognition (LVCSR) and AURORA 4 multi-condition ASR tasks. Learning the model with different speech corpora, we have shown that proposed model is able to learn auditory-like filterbank with time-domain subband filters resemble gammatone filters. Experiments on clean and multi-condition training databases show that our model can improve recognition performance in clean and noisy test conditions. Further improvements can be achieved by combining systems trained using both Mel filterbanks and learned filterbanks as done in [8]. Cross-domain experiments were also conducted to verify the generalized representations the model can learn.

The organization of rest of the paper is as follows: In Section II, we have described our proposed model and algorithm for learning weights and biases of the model. Analysis of model, filterbank and feature representation using learned filterbank is presented in Section III. Details of the experimental setup and database description are given in Section IV. Experimental results in ASR task w.r.t. model parameter tuning and comparison with standard spectral features for various databases are discussed in Section V. Finally, Section VI concludes the paper along with future research directions.

II. Convolutional RBM for Speech Signals

A. Proposed Model Architecture

ConvRBM has two layers, namely, a visible layer and a hidden layer [24], [25]. The input to visible layer (denoted as $\mathbf{x}$) is an entire speech signal of length *n*-samples. Hidden layer (denoted as $\mathbf{h}$) consists of *K*-groups (i.e., number of filters) with filter length *m*-samples in each. Weights (also called as filters or in fact, *subband* filters with respect to speech perception mechanism in hearing [27]) are shared between visible and hidden units amongst all the locations in each group [24]. Weight sharing reduces the number of parameters compared to fully connected RBM and helps model to learn structures in speech signals as discussed later in this Section. Denoting b_k as the hidden bias for the kth group, the response of the convolution layer for the kth group is given as:

$$\mathbf{I}_k = (\mathbf{x} * \tilde{\mathbf{W}}^k) + b_k, \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ are samples of speech signal, $\mathbf{W}^k = [w_1^k, w_2^k, \ldots, w_m^k]$ is a weight vector (i.e., kth subband filter) and $\tilde{\mathbf{W}}$ denote *flipped* array [24]. The architecture of weight connections with input in K groups are shown in Fig. 1. We can see that for the kth group, all hidden units share the same weight vector $\mathbf{W}^k$ and hidden bias b_k. Now, we demonstrate that it is the wiring of neurons that leads to a *convolution* operation for each group. In particular, for the kth group, the input to hidden units is given by a weighted sum as follows:

$$\begin{aligned} y_1^k &= x_1 w_1^k + x_2 w_2^k + \cdots + x_m w_m^k, \\ y_2^k &= x_2 w_1^k + x_3 w_2^k + \cdots + x_{m+1} w_m^k, \\ \vdots \; &= \qquad\qquad\qquad \vdots \\ y_{n-m+1}^k &= x_{n-m+1} w_1^k + x_{n-m+2} w_2^k + \cdots + x_n w_m^k. \end{aligned} \tag{2}$$

In the matrix form, it can be written as,

$$\mathbf{y}^k = H\mathbf{x}^T, \tag{3}$$

where H is a *valid* convolution matrix of weights given as,

$$H = \begin{bmatrix} w_1^k & w_2^k & w_3^k & \ldots & w_m^k & 0 & 0 & \ldots & 0 \\ 0 & w_1^k & w_2^k & w_3^k & \ldots & w_m^k & 0 & \ldots & 0 \\ 0 & 0 & w_1^k & w_2^k & \ldots & w_{m-1}^k & w_m^k & \ldots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 0 & w_1^k & \ldots & w_{m-1}^k & w_m^k \end{bmatrix}$$

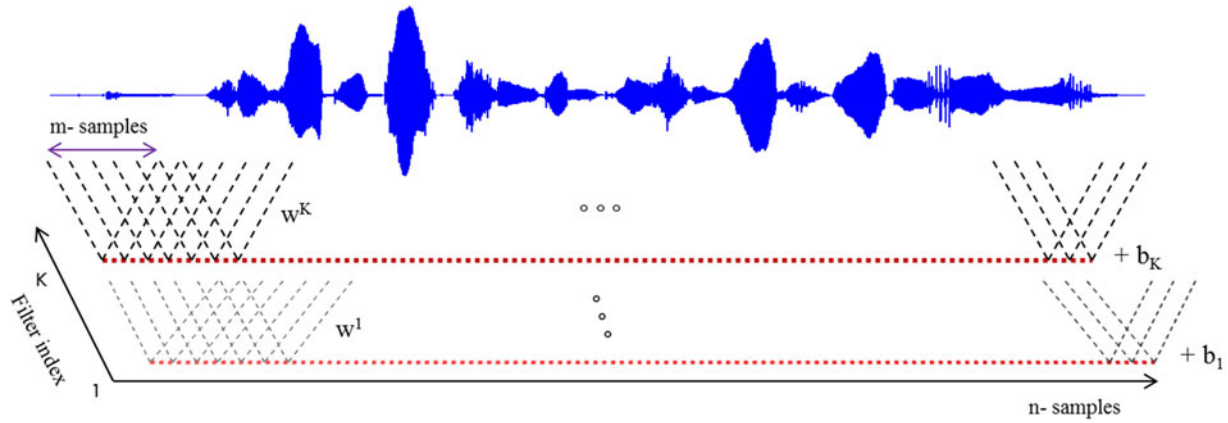

Fig. 1. The arrangement of hidden units in K groups and the corresponding weight connections. The filter index axis is perpendicular to the plane of this paper. Each hidden unit (red dots) in the kth group is wired such that it results in a valid convolution between the speech signal and weights $\mathbf{W}^k$.

Eq. (3) represent the matrix form of convolution operation (valid length). Hence, such a full weight sharing technique leads to convolution between input speech signal and weights in each group. For ConvRBM with visible units $\mathbf{x}$ and hidden units $\mathbf{h}$, the energy function of the model is given as,

$$E(\mathbf{x},\mathbf{h}) = \frac{1}{2\sigma_x^2}\sum_{i=1}^{n} x_i^2 - \frac{1}{\sigma_x}\sum_{k=1}^{K}\sum_{j=1}^{l}\sum_{r=1}^{m}\left(h_j^k w_r^k x_{j+r-1}\right) - \sum_{k=1}^{K} b_k \sum_{j=1}^{l} h_j^k - \frac{1}{\sigma_x^2} c \sum_{i=1}^{n} x_i, \tag{4}$$

where c is a visible bias which is also shared. As derived from eq. (2) and eq. (3), we have used 'valid' length convolution and hence, the length of each group is $l = n - m + 1$. Each speech utterance is normalized to zero-mean and unit-variance. Hence, variance (σ_x) in eq. (4) is set to 1. The probability of joint distribution of visible and hidden units is,

$$p(\mathbf{x},\mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{x},\mathbf{h})}, \tag{5}$$

where Z is the partition function $Z = \sum_{(\mathbf{x},\mathbf{h})} e^{-E(\mathbf{x},\mathbf{h})}$ which normalizes the energy making it a probability distribution function (PDF). In case of sigmoid hidden units, the sampling equations of visible and hidden units are given as [25],

$$\mathbf{h}^k \sim \text{sigmoid}((\mathbf{x} * \tilde{\mathbf{W}}^k) + b_k),$$
$$\mathbf{x}_{\text{recon}} \sim \mathcal{N}\left(\sum_{k=1}^{K}(\mathbf{h}^k * \mathbf{W}^k) + c, 1\right), \tag{6}$$

where $\mathcal{N}(\mu, 1)$ is a Gaussian distribution with mean μ and variance 1. Generalization of binary hidden units is achieved by replacing each binary units with infinite copies of binary units that all have the same weights and progressively more negative bias [28]. This set of units is called as Stepped Sigmoid Units (SSU). It is shown in [28] that SSU can be approximated well by noisy rectified linear units (NReLU). With NReLU, following are the sampling equations for hidden and visible units (reconstruct speech signal $\mathbf{x}_{\text{recon}}$ to further update hidden units) [23]:

$$\mathbf{h}^k \sim \max(0, \mathbf{I}_k + N(0, \sigma(\mathbf{I}_k))),$$
$$\mathbf{x}_{\text{recon}} \sim \mathcal{N}\left(\sum_{k=1}^{K}(\mathbf{h}^k * \mathbf{W}^k) + c, 1\right), \tag{7}$$

where $N(0, \sigma(I_k))$ is a Gaussian noise with zero-mean and sigmoid of $\mathbf{I}_k$ as it's variance. While calculating the relationship between hidden and visible units, deterministic ReLU (i.e., $\max(0, \mathbf{I}_k)$) is used as an activation function of hidden units as discussed in Section II-B. With a convolution layer and ReLU nonlinearity, an example of processing stages is shown in Fig. 2. The convolution with subband filters (i.e., weights of the proposed model) decomposes the speech signal into different subbands. We will see in Section III-A that such decomposition of the speech signal is due to using different subband filters that are localized in frequency-domain. Learning of such subband filters is possible because of weight sharing in ConvRBM. ReLU reduces the information by making negative values to zero that leads to *sparsity* in the hidden units.

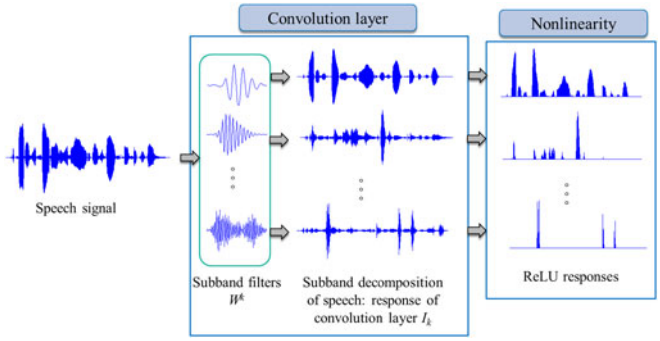

Fig. 2. Example of decomposition of a speech signal using weights of ConvRBM with convolution layer followed by ReLU nonlinearity.

B. Model Learning

For probabilistic model ConvRBM with parameters $\theta\left(\mathbf{W}^k, b_k, c\right)$, the gradient of log-likelihood is given as [29],

$$\frac{\partial}{\partial\theta}\ell(\mathbf{x};\theta) = -\int_{-\infty}^{\infty} p(\mathbf{h}|\mathbf{x})\frac{\partial}{\partial\theta}E(\mathbf{x},\mathbf{h})\mathrm{d}\mathbf{h} + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(\mathbf{x},\mathbf{h})\frac{\partial}{\partial\theta}E(\mathbf{x},\mathbf{h})\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{h}. \quad (8)$$

With the notations used in [29], we can write log-likelihood in terms of expectations as:

$$\frac{\partial}{\partial\theta}\ell(\mathbf{x};\theta) = -\mathbb{E}_{p(\mathbf{h}|\mathbf{x})}\left[\frac{\partial}{\partial\theta}E(\mathbf{x},\mathbf{h})\right] + \mathbb{E}_{p(\mathbf{h},\mathbf{x})}\left[\frac{\partial}{\partial\theta}E(\mathbf{x},\mathbf{h})\right], \approx -\left\langle\frac{\partial}{\partial\theta}E(\mathbf{x},\mathbf{h})\right\rangle_{\text{data}} + \left\langle\frac{\partial}{\partial\theta}E(\mathbf{x},\mathbf{h})\right\rangle_{\text{model}}, \quad (9)$$

where $\langle\cdot\rangle$ is the sample mean under distribution used to calculate expectations. Here, $\langle\cdot\rangle_{\text{data}}$ is the sample mean calculated when visible units are clamped to speech signal (i.e., input data) and $\langle\cdot\rangle_{\text{model}}$ is the sample mean estimated when visible and hidden units are sampled from a model distribution. The first part of eq. (9), for $\mathbf{W}^k$ as a model parameter, can be computed by taking a derivative of eq. (4) w.r.t. $\mathbf{W}^k$ which gives us following equation (details are given in Appendix A):

$$\frac{\partial}{\partial\mathbf{W}^k}E(\mathbf{x},\mathbf{h}) = -\sum_{j=1}^{l} h_j^k x_{j+r-1}, = -\text{conv}(\mathbf{x},\tilde{\mathbf{h}^k}), \quad (10)$$

where $\tilde{\mathbf{h}^k}$ is a *flipped* array to represent the convolution operation denoted as $conv(\cdot)$. The length of this valid convolution between the input of length n samples and the kth hidden group of length $l = n - m + 1$ (obtained from the convolution of input and weights), is $n - l + 1 = m$ samples. This term is easy to calculate. We clamp visible units to speech signal $\mathbf{x}$ and find hidden unit activations. Hidden unit activations can be found by passing convolution responses $\mathbf{I}_k$ from the deterministic ReLU nonlinearity. Then the relationship between hidden and visible units can be found using eq. (10). This is called the *positive phase* of CD learning [30].

The second term in eq. (9) require samples from a model distribution which is very difficult to obtain. Infinite steps in Gibbs sampling can be well approximated in finite time using a technique called as *Contrastive Divergence* (CD) [30]. Instead of sampling infinite times, we can sample only up to *N* times called as (CD-N) or it is shown in [30] that even single step gives a good approximation called as CD-1. We have used a single step CD learning as shown in Fig. 3. Updating hidden units using reconstructed speech signal is called the *negative phase* of CD learning [30].

The second term in eq. (9) can be written as,

$$\frac{\partial}{\partial\mathbf{W}^k}E(\underline{\mathbf{x}},\underline{\mathbf{h}}) = -\text{conv}(\underline{\mathbf{x}},\underline{\tilde{\mathbf{h}}^k}), \quad (11)$$

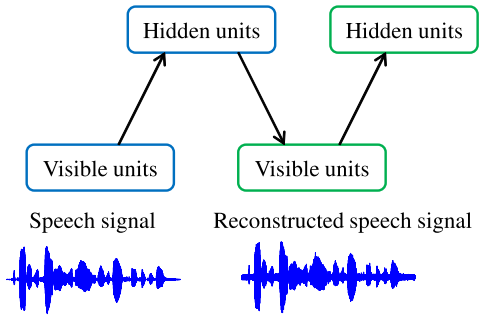

Fig. 3. Demonstration of single-step Constrastive Divergence (CD-1) learning.

where the underline symbol denotes visible ($\underline{\mathbf{x}} = \mathbf{x}_{\text{recon}}$) and hidden states ($\underline{\tilde{\mathbf{h}}^k}$) in the CD-1 stage (negative phase). We obtain samples of visible and hidden units using eq. (7). For weights of the model, eq. (9) can now be written as,

$$\frac{\partial}{\partial\mathbf{W}^k}\ell(\mathbf{x};\theta) = \mathbb{E}_{p(\mathbf{h}|\mathbf{x})}\left[\text{conv}(\mathbf{x},\tilde{\mathbf{h}^k})\right] - \mathbb{E}_{p(\mathbf{h},\mathbf{x})}\left[\text{conv}(\underline{\mathbf{x}},\underline{\tilde{\mathbf{h}}^k})\right], \approx \left\langle\text{conv}(\mathbf{x},\tilde{\mathbf{h}^k})\right\rangle_{\text{data}} - \left\langle\text{conv}(\underline{\mathbf{x}},\underline{\tilde{\mathbf{h}}^k})\right\rangle_{\text{model}}. \quad (12)$$

The corresponding gradient-ascent update for weights is now written as,

$$\nabla\mathbf{W}^k = \epsilon\left(\left\langle\text{conv}(\mathbf{x},\tilde{\mathbf{h}^k})\right\rangle_{\text{data}} - \left\langle\text{conv}(\underline{\mathbf{x}},\underline{\tilde{\mathbf{h}}^k})\right\rangle_{\text{model}}\right), \quad (13)$$

where ϵ is a learning rate parameter.

The gradient-ascent update equations for hidden and visible biases can be derived as below:

$$\nabla b_k = \epsilon\left(\left\langle\sum_{j=1}^{l} h_j^k\right\rangle_{\text{data}} - \left\langle\sum_{j=1}^{l}\underline{\tilde{h}^k}\right\rangle_{\text{model}}\right), \quad \nabla c = \epsilon\left(\left\langle\sum_{i=1}^{n} x_i\right\rangle_{\text{data}} - \left\langle\sum_{i=1}^{n}\underline{x_i}\right\rangle_{\text{model}}\right). \quad (14)$$

The iterative updates for model parameters $\theta\left(\mathbf{W}^k, b_k, c\right)$ are given as below [29]:

$$\theta^{(t+1)} = \theta^{(t)} + \nabla\theta^{(t)} + \eta\theta^{(t-1)}, \quad (15)$$

where the momentum term with parameter η is known to help against oscillatory behavior in parameter space and accelerate the learning process [29]. Steps for model learning using CD-1 are described in Algorithm 1.

C. Feature Extraction

After ConvRBM is trained, pooling is applied to reduce the representation of ConvRBM filter responses in the temporal-domain. Our proposed model is different from the one used in [25] where probabilistic max-pooling was used in inference stage itself for binary hidden units. Our approach resembles the method used in [31] where time-domain gammatone responses

Algorithm 1: Proposed Algorithm for ConvRBM Training Applied on Speech Signals.

Input: Speech signals $\mathbf{x}$ with arbitrary length n samples.
Output: Weights $\mathbf{W}$, hidden biases b and visible bias c.

for each training iteration t **do**
  Use weights and biases updated during last iteration $t-1$
  for each training example $\mathbf{x}$ **do**
    for each kth group **do**
      Convolution response $\mathbf{I}_k = (\mathbf{x} * \tilde{\mathbf{W}}^k) + b_k$
      $\mathbf{h}^k_{\text{act}} = \max(0, \mathbf{I}_k)$
      $\mathbf{h}^k_{\text{sample}} \sim \max(0, \mathbf{I}_k + N(0, \sigma(\mathbf{I}_k)))$
      $VH = \text{conv}(\mathbf{x}, \mathbf{h}^k_{\text{act}})$
      $H_{\sum} = \sum(\mathbf{h}^k_{\text{act}})$
    end for
    Sample visible units (reconstruct speech signal) from hidden units as:
    $\mathbf{x}_{\text{recon}} \sim \mathcal{N}\left(\sum_k (\mathbf{h}^k_{\text{sample}} * \mathbf{W}^k) + c, 1\right)$
    for each kth group **do**
      Convolution response $\underline{\mathbf{I}}_k = (\mathbf{x}_{\text{recon}} * \tilde{\mathbf{W}}^k) + b_k$
      $\underline{\mathbf{h}}^k_{\text{act}} = \max(0, \underline{\mathbf{I}}_k)$
      $\underline{\mathbf{h}}^k_{\text{sample}} \sim \max(0, \underline{\mathbf{I}}_k + N(0, \sigma(\underline{\mathbf{I}}_k)))$
      $\underline{VH} = \text{conv}(\mathbf{x}_{\text{recon}}, \underline{\mathbf{h}}^k_{\text{act}})$
      $\underline{H}_{\sum} = \sum(\underline{\mathbf{h}}^k_{\text{act}})$
    end for
    $\nabla \mathbf{W}^{(t)} = [VH - \underline{VH}]/n$
    $\nabla \mathbf{b}^{(t)} = [H_{\sum} - \underline{H}_{\sum}]/n$
    $\nabla c^{(t)} = [\sum(\mathbf{x}) - \sum(\mathbf{x}_{\text{recon}})]/n$
    $\mathbf{W}^{(t+1)} \leftarrow \mathbf{W}^{(t)} + \epsilon \nabla \mathbf{W}^{(t)} + \eta \mathbf{W}^{(t-1)}$
    $\mathbf{b}^{(t+1)} \leftarrow \mathbf{b}^{(t)} + \epsilon \nabla \mathbf{b}^{(t)} + \eta \mathbf{b}^{(t-1)}$
    $c^{(t+1)} \leftarrow c^{(t)} + \epsilon \nabla c^{(t)} + \eta c^{(t-1)}$
  end for
end for

were reduced using average-based framing, which is a pooling-like operation. Such an approach is also used in [22] where after convolution with scattering wavelets, averaging is performed using lowpass filtering. Here, pooling in the time-domain is equivalent to short-time averaging in spectral features such as MFCC and lowpass filtering in scattering wavelets. For a speech signal of sampling frequency, $Fs = 16$ kHz, pooling is applied using 25 ms (i.e., 400 speech samples) window length (wl) and 10 ms (i.e., 160 speech samples) shift (ws). We used this setup to compare standard spectral features (e.g., MFCC) extracted using same windowing parameters. Pooling is performed across time and separately for each subband filter. The speech signal with n-samples has $F = \frac{n-wl+ws}{ws}$ number of frames. We have experimented with both average and max-pooling and found better results with average pooling. After the pooling operation, stabilized logarithm $\log(\cdot + \delta)$ as a compressive nonlinearity, is applied as done in [32] (with $\delta = 0.0001$).

The block diagram for feature extraction procedure (described above) is shown in Fig. 4. To obtain the same length as the speech signal, 'same' length convolution is used. During feature extraction stage, we have used deterministic ReLU nonlinearity $\max(0, \mathbf{I}_k)$ as an activation function of hidden units. The pooling operation reduce temporal resolution from $K \times n$ samples to $K \times F$ frames. Logarithmic nonlinearity compresses the dynamic range of features which was found to improve performance in ASR tasks [32]. The feature extraction steps involved in this ordering resemble the auditory processing in human audition [1], [33].

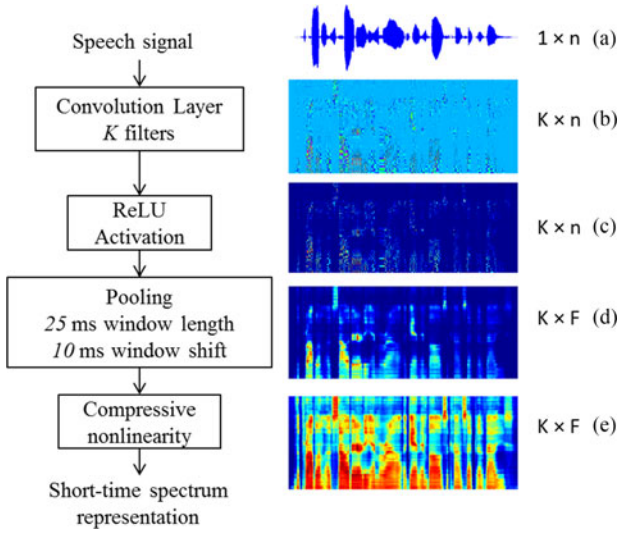

Fig. 4. Block diagram of stages in feature representation using ConvRBM. To show figures on right side, filters were arranged in increasing order of their center frequencies. (a) speech signal, (b) and (c) responses from convolution layer and ReLU nonlinearity, respectively, (d) reduced representation after pooling, (e) logarithmic compression. After [23].

III. Analysis of ConvRBM

A. Analysis of Learned Subband Filters

For analysis of the subband filters, we found the center frequencies (CFs) of subband filters as described in [11]. We have analyzed the model with $K = 60$ subband filters, i.e., 60 groups in hidden layer. Examples of subband filters learned using ConvRBM on TIMIT, WSJ1 and AURORA 4 databases are shown in Fig. 5. Filters were arranged according to their increasing order of CFs. Weights of ConvRBM were initialized randomly and there is no constraint on filter shapes; still, the model was able to learn meaningful representation from the speech signals. Weights of the model called an impulse responses of subband filters in time-domain are shown in Fig. 5(a)–(c). We can see that for all the three databases, many filters are very similar to auditory gammatone filters (i.e., primarily motivated by the studies reported in [14], [21]). Unlike the filters derived using RBM [19], our learned subband filters resemble more closely to auditory subband filters for speech signals [14]. This may be due to the fact that RBM was trained on *randomly* selected smaller windows of the speech signal and hence, they were in any random temporal phase [19]. We have trained our model on speech signals in time-domain without windowing to learn filters and each subband responses are pooled later to get the short-time spectral representation of the speech signal. Fig. 5(d)–(f)

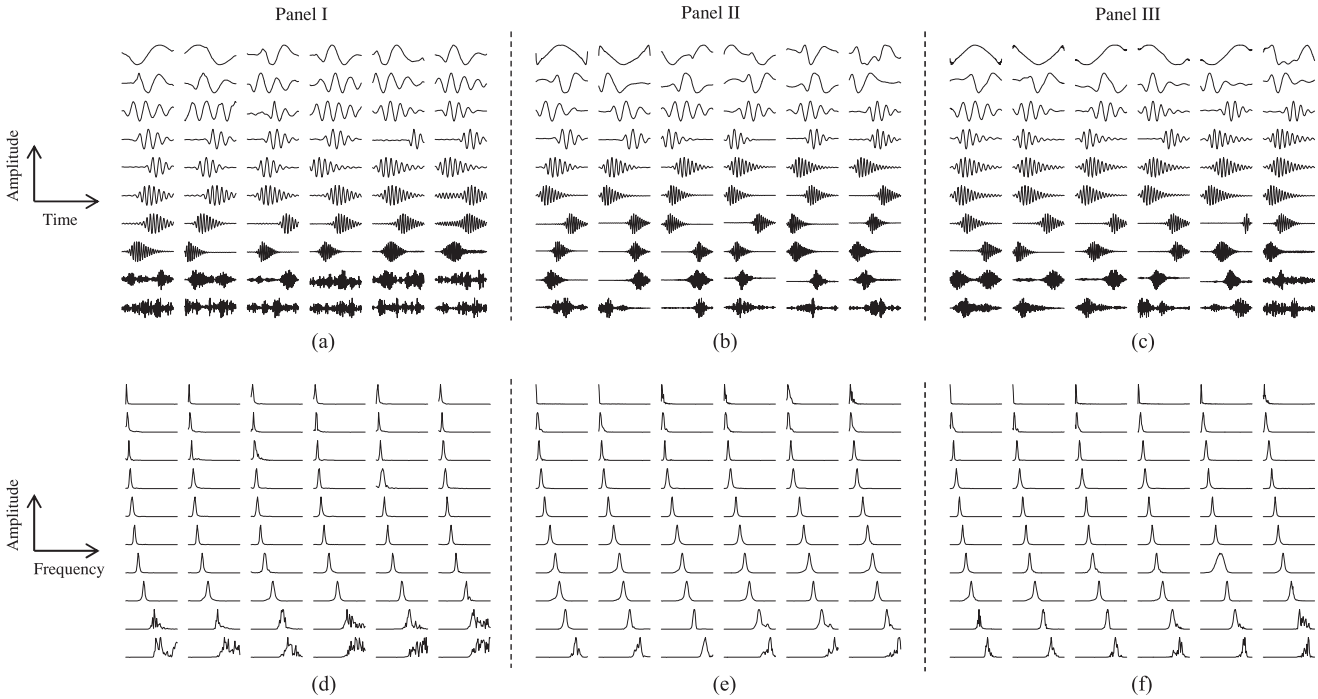

Fig. 5. Examples of subband filters trained on TIMIT (Panel I), WSJ1 (Panel II) and AURORA 4 (Panel III) databases, respectively: (a)-(c) subband filters in time-domain (i.e., impulse responses), (d)-(f) subband filters in frequency-domain (i.e., frequency responses).

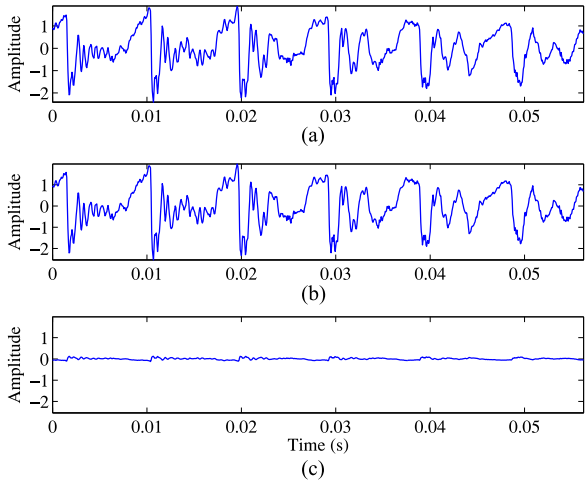

Fig. 6. (a) Segment of speech signal, (b) reconstructed speech from proposed model, (c) residual error. Root Mean Squared Error (RMSE) between original and reconstructed speech signal is 0.032.

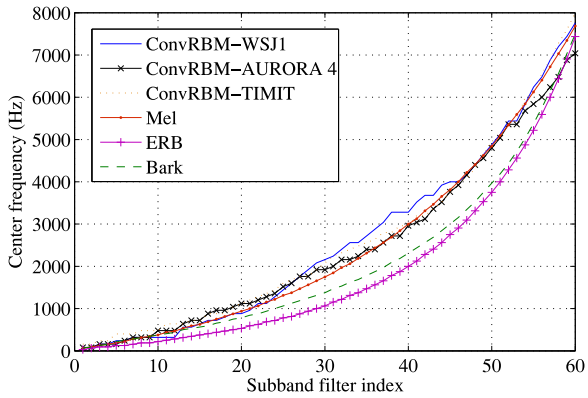

Fig. 7. Comparison of filterbank learned using ConvRBM with auditory filterbanks. After [23].

shows the frequency-domain representation of corresponding time-domain impulse responses. We can see that all the subband filters are localized in the frequency-domain with different CFs. Filters with lower CFs are highly localized in the frequency-domain while with higher CFs are more broader in terms of their bandwidth.

Proposed subband filters can also accurately reconstruct speech signal even after ReLU nonlinearity. Small segment of the original speech signal (about 500 samples) from WSJ1 database, a segment of a reconstructed speech from model and the residual error is shown in Fig. 6. From the residual error (RMSE = 0.032), we can see a very accurate reconstruction of the speech signal.

B. Comparison With Standard Auditory Filterbanks

In order to compare learned filterbank with standard auditory filterbanks, we have shown a CF *vs*. subband filter index plot in Fig. 7 for a filterbank learned on three databases. We can see that the ConvRBM filterbank has also a nonlinear relationship between CF and filter ordering similar to as other auditory filterbanks (more closely with Mel scale). This represents the placement of subband filters on the basilar membrane (BM) in the cochlea. Out of 60 subband filters, more than 40 subband filters have CFs below 4 kHz. Low frequency regions are represented by more number of subband filters learned by the model

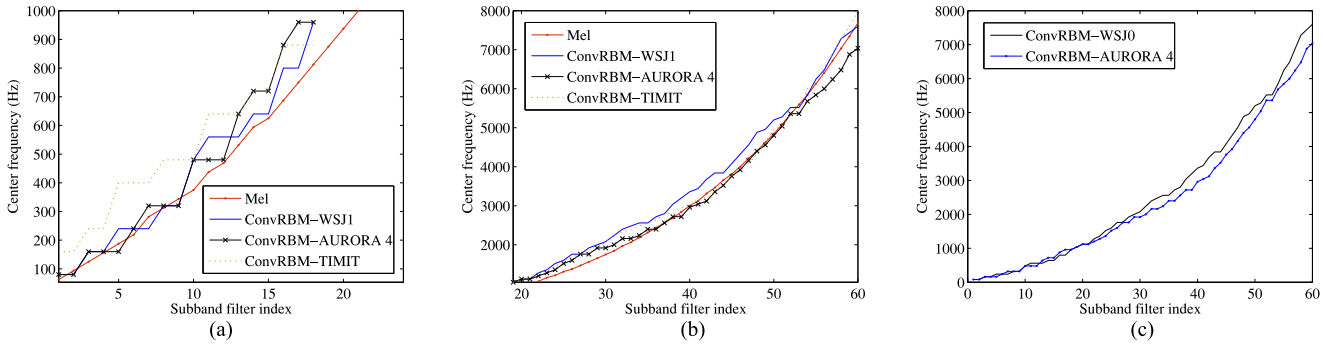

Fig. 8. Comparison of filterbank learned using ConvRBM with auditory filterbanks: (a) CF up to 1 kHz, (b) CF from 1 to 8 kHz, (c) between clean and multicondition training database.

compared to high frequency regions (similar to the Mel scale). Hence, learned filters can represent *frequency tuning* in the human cochlea which can be modeled more effectively using a bank of subband filters.

A detailed comparison of filterbanks is shown in Fig. 8. In Fig. 8(a), filterbanks are compared with CFs up to 1 kHz. We can see that in all learned filterbanks, some of the subband filters have similar CFs. This redundancy is only observed for CFs up to 1 kHz and not in CFs above 1 kHz as shown in Fig. 8(b). This may be due to lack of sparsity regularization even though rectified units are used. We have also compared filterbanks trained on clean WSJ0 database and multi-condition training database AURORA 4 in Fig. 8(c). The difference between both filterbanks can be seen after 2 kHz since AURORA 4 filterbanks use more subband filters compared to clean WSJ0 in low frequency regions. This observation is different than the one reported in [8] where filterbank trained on a clean database uses more subband filters than the noisy database. However, the major difference is that, the weights of our model were randomly initialized. Hence, there were no constraints on learning the weights while in [8] weights were initialized using gammatone to compare filterbanks on clean and the noisy database.

The spectrum representation of the speech signal using subband filters is compared with log-Mel spectrogram in Fig. 9. Similar to a log-Mel spectrogram, a ConvRBM spectrogram indeed represents spectrum information such as formant contours, voiced and unvoiced sounds, etc. The regions marked using solid lines shows that learned subband filters are capturing spectrum information. However, the filterbank scale is slightly different from the Mel scale as seen from Fig. 7. We also have noticed that for the ConvRBM filterbank, the resolution is slightly poor at higher frequencies compared to the log-Mel spectrogram (the region marked by the dotted circles).

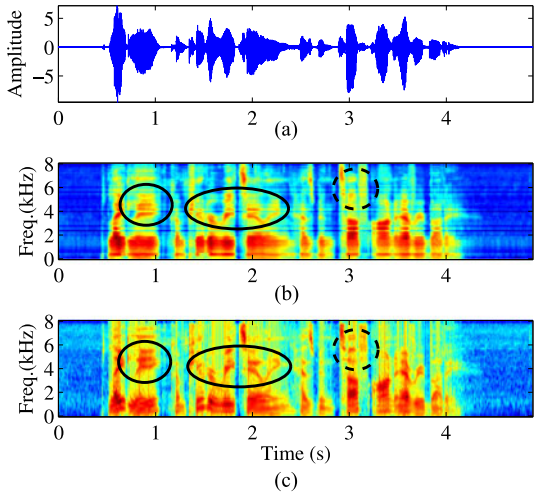

Fig. 9. (a) Speech signal, spectrogram using (b) ConvRBM filterbank, (c) log-Mel spectrogram. Full line regions are marked to see similarities and dotted circle indicates differences in representation in spectrogram.

IV. Experimental Details

A. Speech Databases

1) Small Vocabulary Speech Database: We have used the TIMIT database for phone recognition task [34]. In the TIMIT database, all SA category sentences (same sentences spoken by all the speakers) were removed as they may bias the speech recognition performance. Training data contains utterances from 462 speakers. Development set and test set contains utterances from 50 and 24 speakers, respectively.

2) Large Vocabulary Speech Databases: LVCSR tasks were performed using the Wall Street Journal databases [35]. Both WSJ0 SI-84 (the subset of WSJ) and full WSJ corpus are used for experiments. WSJ0 SI-84 training data consists of 14 hours of speech data which includes 7138 utterances spoken by 84 speakers. Two Nov'92 evaluation sets, namely, 5K-word and 20K-word vocabulary (denoted as Eval92_5K and Eval92_20K, respectively) were used for testing. WSJ1 training database is of 81 hours. Notations of the development and the evaluation sets for WSJ1 are as follows: D1 and D2 for the development sets Dev93 and Dev93_5k, E1 and E2 for the evaluation sets Eval93 and Eval93_5k, E3 and E4 for the evaluation sets Eval92 and Eval92_5k, respectively.

3) Noisy Speech Database: We have also used the AURORA 4 database (obtained from WSJ0 database) which was created using six different types of additive noises, namely, car, a crowd of people (babble), restaurant, street, airport and train station [36]. Multi-condition training database was prepared with 7138 utterances from WSJ0 database with half of them recorded with

the Sennheiser microphone and the other half recorded with the second microphone. The type of noise is randomly chosen out of 6 noises in total and at a randomly chosen SNR between 10 dB and 20 dB. A set of 330 utterances has been designated in the ARPA evaluation to perform a baseline recognition on the 5K word vocabulary. The test set consists of 14 subsets, each with 330 utterances denoted as T1 to T14. The test sets are grouped into four categories, namely, A: clean (set T1), B: noisy (set T2 to set T7), C: clean with channel distortion (set T8) and D: noisy with channel distortion (set T9 to T14).

B. Training of ConvRBM and Feature Extraction

We have trained ConvRBM on each individual speech databases. Each speech signal after mean-variance normalization was applied to ConvRBM. The learning rate was chosen to be 0.005, which was fixed for first 10 epoch and decayed later at each epochs for stable learning of model parameters. We observed that, with rectified linear units, only 25-35 training epochs were sufficient. For first five training epochs, momentum was set to 0.5 and after that, it was set to 0.9. We have trained the model with a different lengths of ConvRBM filters and with different number of subband filters. After the model was trained, features were extracted from speech signal as shown in Fig. 4. To reduce the dimension and compare proposed features with MFCC features, the Discrete Cosine Transform (DCT) was applied and only first 13-D were retained. Delta and delta-delta features were also appended resulting in 39-D cepstral features (indicated as ConvRBM-CC). We have not used DCT for filterbank experiments and restrict to use only 40 filters of ConvRBM (indicated as ConvRBM-BANK) similar to 40-D Mel filterbanks.

C. ASR System Building

Baseline monophone GMM-HMM systems and hybrid DNN-HMM systems were built using 39-D MFCC features and 120-D FBANK features, respectively, for all the databases used for experiments in this paper. MFCC features were extracted from windowed speech signal with 25 ms length window and 10 ms window shift similar as in parameters of pooling. For the TIMIT database, 48 phones were used for training and mapped to 39 phones during scoring [37]. Language modeling (LM) was performed using a bi-gram language model. For WSJ databases (WSJ0 and WSJ1), tri-gram 5K and 20K LM were used. The bi-gram and tri-gram 5K LM were used for AURORA 4 test sets. In this paper, all ASR systems were built using the KALDI speech recognition toolkit [38]. We also experimented with the hybrid DNN-HMM system using forced-aligned labels obtained from corresponding GMM-HMM systems. The results are reported using DNN with 3 hidden layers, 11 frame context-window and 3000 hidden units. The DNN-HMM system combination is performed using Minimum Bayes Risk (MBR) technique [39]. Lattices generated by N different systems are combined to get optimal word sequence as follows [39]:

$$W^* = \arg\min_{W} \left\{ \sum_{i=1}^{N} \lambda_i \sum_{W'} P_i(W'|\mathbf{O}) L(W, W') \right\}, \quad (16)$$

where $L(W, W')$ is the Levenshtein edit distance between two word sequences, $P_i(W'|\mathbf{O})$ is the posterior probability of the word sequence W' given the acoustic observation sequence $\mathbf{O}$ and λ_i is the weight assigned to the ith system.

TABLE I
% PER FOR COMPARISON OF NUMBER OF SUBBAND FILTERS (K), FILTER LENGTH (m) AND POOLING TYPE ON TIMIT DATABASE

K	m	Pooling type	Dev	Test
40	128	Avg	32.0	32.6
60	128	Avg	**31.2**	**31.8**
80	128	Avg	31.5	31.9
60	96	Avg	31.4	32.5
60	160	Avg	31.7	33.0
60	256	Avg	32.8	33.5
60	128	Max	32.6	33.5

Avg = Average Pooling, Max = Maximum Pooling

V. EXPERIMENTAL RESULTS

The significance of the learned filterbanks using various datasets is verified using a phone recognition task, a LVCSR task and speech recognition in degraded conditions. We will first fine-tune the parameters of the model for each individual database and use the optimal set of parameters in corresponding speech recognition experiments.

A. Experiments on TIMIT Database

In this Section, the effect of a number of subband filters (K), filter length (m) and pooling type is verified through experiments on the TIMIT database using GMM-HMM systems and results are reported in Table I [23]. We can see that optimal filter length corresponding to the least Phone Error Rate (PER) (100 - % Phone Recognition Accuracy) is 128 samples on development (Dev) and test set. A filter length of 128 samples (i.e., 8 ms) is sufficient to capture small temporal variations in speech signals [14]. In our case, average pooling works better than max-pooling. Since we are using the rectifier nonlinearity, it eliminates the cancellations between neighboring filter outputs when combined with average pooling [40]. Hence, we achieved good performance with average pooling. Best performance is obtained relatively with 60 subband filters, 128 samples filter length and using average pooling.

The experimental results are reported in % PER and % relative improvement (in brackets) in Table II [23]. The relative improvements due to ConvRBM-CC and ConvRBM-FBANK are shown with respect to MFCC and FBANK, respectively. We can see that ConvRBM-CC perform better than MFCC features giving an absolute reduction of 1.5% in PER on the development set and 1.7% on the test set. Table II shows that for DNN-HMM systems, there is an absolute reduction of 1.1% in PER using ConvRBM-CC features and 0.7% using ConvRBM-BANK on the development set. We achieved an absolute reduction of 0.7% (3.15% relative) in PER using ConvRBM-CC and 0.6% (2.56% relative) using ConvRBM-BANK on the test set. Combining systems (denoted as $\oplus$) trained using both filterbank

TABLE II
% PER AND RELATIVE IMPROVEMENTS FOR TIMIT DATABASE

Feature Set	System	Dev	Test
MFCC	GMM-HMM	32.7	33.5
ConvRBM-CC	GMM-HMM	**31.2** (4.59)	**31.8** (5.07)
MFCC	DNN-HMM	23.0	24.0
ConvRBM-CC	DNN-HMM	**21.9** (4.78)	**23.3** (2.92)
A:FBANK	DNN-HMM	22.2	23.4
B:ConvRBM-BANK	DNN-HMM	**21.5** (3.15)	**22.8** (2.56)
A $\oplus$ B	DNN-HMM	**20.5** (7.66)	**21.7** (7.26)
CNN with the raw speech [41]		-	29.9

TABLE III
% WER FOR COMPARISON OF NUMBER OF SUBBAND FILTERS (K), FILTER LENGTH (m) AND POOLING TYPE ON WSJ0 DATABASE

K	m	Pooling type	Eval92_5K	Eval92_20K
40	128	Avg	13.49	26.21
60	128	Avg	12.96	25.80
80	128	Avg	13.41	25.66
60	96	Avg	13.97	25.94
60	160	Avg	13.25	26.1
60	256	Avg	13.75	27.01
60	128	Max	13.50	26.80

TABLE IV
% WER AND RELATIVE IMPROVEMENTS FOR WSJ0 DATABASE

Feature Set	System	Eval92_5K	Eval92_20K
MFCC	GMM-HMM	13.95	27.72
ConvRBM-CC	GMM-HMM	**12.96** (7.09)	**25.80** (6.93)
MFCC	DNN-HMM	6.30	15.70
ConvRBM-CC	DNN-HMM	**6.05** (3.97)	**13.40** (14.65)
A:FBANK	DNN-HMM	6.07	14.32
B:ConvRBM-BANK	DNN-HMM	**5.85** (3.62)	**13.52** (5.59)
A $\oplus$ B	DNN-HMM	**5.40** (11.04)	**13.08** (8.66)

features give the absolute reduction of 1% in PER compared to ConvRBM-BANK and 1.7% PER compared to FBANK features. The comparison of supervised CNN trained on the raw speech signals shows that unsupervised ConvRBM features indeed perform better on small size datasets. However, later we observed in Section V-C that supervised CNN perform well with larger datasets such as WSJ.

B. Experiments on WSJ0 Database

The effects of parameters of ConvRBM were tested on the WSJ0 database and results are reported in Table III. We can see a similar set of parameters as TIMIT that resulted in low % Word Error Rate (WER). Using these parameters, results of ASR experiments are reported in Table IV in terms of % (WER) [23]. There is an absolute reduction of 0.99% WER on eval92_5K test set and 1.92% WER on eval92_20K test set over MFCC features using GMM-HMM system. Significant absolute reduction of 2.3% WER is obtained for the 20K test set using DNN-HMM systems. The lowest WER 5.85% (3.6% relative improvement) for the 5K test is achieved with

TABLE V
% WER AND % RELATIVE IMPROVEMENTS FOR WSJ1 LVCSR TASK

Features	D1	D2	E1	E2	E3	E4
GMM-HMM system						
MFCC	37.45	23.04	30.95	17.87	26.60	13.94
ConvRBM-CC	**34.37**	**21.19**	**29.35**	**17.53**	**23.75**	**12.22**
	(8.22)	(8)	(5.17)	(1.9)	(10.71)	(12.34)
DNN-HMM system						
MFCC	20.94	11.40	17.75	9.76	13.93	5.47
ConvRBM-CC	**18.92**	**9.87**	**16.47**	**8.02**	**12.19**	**4.75**
	(9.65)	(13.42)	(7.21)	(17.82)	(12.49)	(13.16)
A:FBANK	18.70	9.66	17.31	8.10	12.26	4.39
B:ConvRBM-BANK	**17.96**	**9.53**	**16.13**	**7.92**	**11.80**	4.91
	(3.96)	(1.35)	(6.82)	(2.22)	(3.75)	(-11.84)
A $\oplus$ B	**17.26**	**9.04**	**15.86**	**7.76**	**11.31**	**4.22**
	(7.7)	(6.42)	(8.38)	(4.2)	(7.75)	(3.87)
ConvRBM-BANK with CD-DNN-HMM, bi-gram 5k LM						6.4
CNN with the raw speech, bi-gram 5k LM [10]						5.6

ConvRBM-BANK while improvement is less using ConvRBM-CC. There is a relative improvement of 14.6% over MFCC and 5.6% over FBANK features is achieved for 20K test set. The ConvRBM-CC and ConvRBM-BANK yielded almost similar WER for the 20K test set. This may be due different number of subband filters in ConvRBM features ($K = 60$ followed by 13-D DCT) and ConvRBM-BANK ($K = 40$) to compare with results with MFCC and FBANK, respectively. However, compared to FBANK features, absolute reduction of 1.24% (8.66% relative) in WER for 20K test set and 0.67% (11.40% relative) for 5K test set was achieved by combining both filterbanks trained DNN-HMM systems.

C. Experiments on Full WSJ Database

With the parameters of ConvRBM obtained from WSJ0, we experimented on the full WSJ database (i.e., WSJ1). All the systems were tested on different development and evaluation sets. The results are reported in Table V in terms of % Word Error Rate (WER). We can see that our ConvRBM-CC features gives an improvement on all test sets. Using GMM-HMM systems, we achieved the relative improvement of 8% (3–1.85% absolute) on development sets, 1.9–5.2% on evaluation sets: E1 and E2 and 10.7–12.3% (2.85–1.72% absolute) on evaluation sets: E3 and E4. Using DNN-HMM systems, ConvRBM-CC gives relative improvement of 9.65–13.42% on development sets, 7.21–17.8% on evaluation sets E1 and E2 and 12.49–13.16% on evaluation sets E3 and E4. Experiments on filterbank features also show improvements (1.35–6.82% relative improvements) using ConvRBM-BANK compared to FBANK except on test set E4. System combination of both filterbanks gives further improvements with absolute reduction of 1.44%, 1.45% and 0.95% in WER for test set D1, E1 and E3, respectively, compared to FBANK features. The comparison with supervised CNN (for Nov92 5K test set E4) trained on the raw speech signals shows that on larger data sets, supervised method perform slightly better compared to our unsupervised method.

TABLE VI
% WER FOR COMPARISON OF NUMBER OF SUBBAND FILTERS (K), FILTER LENGTH (m) AND POOLING TYPE ON AURORA 4 DATABASE

K	m	Pooling type	A	B	C	D	Avg
40	128	avg	21.65	35.70	38.71	51.65	36.92
60	128	avg	22.53	32.77	**36.63**	49.04	35.24
60	128	max	21.48	**32.72**	37.34	49.08	35.15
80	128	avg	21.95	34.2	36.93	50.53	35.90
60	160	avg	**21.02**	32.76	37.08	**48.95**	**34.95**

TABLE VII
% WER AND % RELATIVE IMPROVEMENTS FOR AURORA 4 DATABASE

Feature set	A	B	C	D	Avg
GMM-HMM system tri-gram 5k LM					
MFCC	22.62	33.21	39.4	49.51	36.18
ConvRBM-CC	**21.02**	**32.76**	**37.08**	**48.95**	**34.95**
	(7.07)	(1.35)	(5.89)	(1.13)	(3.39)
CI-DNN-HMM system with tri-gram 5k LM					
MFCC	17.92	26.63	32.97	43.36	30.22
ConvRBM-CC	**17.06**	**24.84**	**28.47**	**40.31**	**27.67**
	(4.8)	(6.72)	(13.65)	(7.03)	(8.44)
S1:FBANK	12.33	21.59	29.35	38.57	25.46
S2:ConvRBM-BANK	**10.76**	**20.34**	**25.5**	**36.84**	**23.36**
	(12.73)	(5.79)	(13.12)	(4.49)	(8.24)
S1 $\oplus$ S2	**10.65**	**19.22**	**26.42**	**36.38**	**23.17**
	(13.63)	(10.98)	(10)	(5.68)	(8.99)
CD-DNN-HMM system with bi-gram 5k LM					
S3:FBANK	10.61	14.85	20.38	30.71	19.12
S4:ConvRBM-BANK	**9.68**	**14.81**	**19.58**	**29.69**	**18.44**
	(8.77)	(0.3)	(3.9)	(3.32)	(3.6)
S3 $\oplus$ S4	**9.47**	**13.91**	**18.85**	**28.52**	**17.69**
	(10.74)	(6.33)	(7.5)	(7.31)	(7.48)
CD-DNN-HMM system with tri-gram 5k LM					
S5:FBANK	5.62	9.29	15.15	24.27	13.58
S6:ConvRBM-BANK	**4.89**	**9.15**	**13.86**	**23.93**	**12.95**
	(12.98)	(1.5)	(8.5)	(1.4)	(4.6)
S5 $\oplus$ S6	**4.71**	**8.43**	**13.53**	**22.74**	**12.35**
	(16.19)	(9.26)	(10.69)	(6.3)	(9.06)

D. Experiments on AURORA 4 Database

We used multi-condition training data for ASR system building. ConvRBM parameter tuning experiments on AURORA 4 database are shown in Table VI. Here, we got different settings of parameters for test sets. Since we want robustness against signal degradation conditions, we choose % WER of test sets, *B* and *D* as ConvRBM parameter selection criteria. ConvRBM with filter lengths 160 and 60 number of filters, is found to perform relatively best for test sets *B* and *D* (however, the difference in % WER using both sets of parameters is very small).

The comparison of different features is given in Table VII. Performance on test sets with channel distortions is improved using context-independent DNN-HMM (CI-DNN-HMM) systems. We obtained relative reduction of 6.7% WER on test set *B* and an absolute reduction of 4.5% and 3.05% in WER for test sets *C* and *D*, respectively. ConvRBM-BANK features give

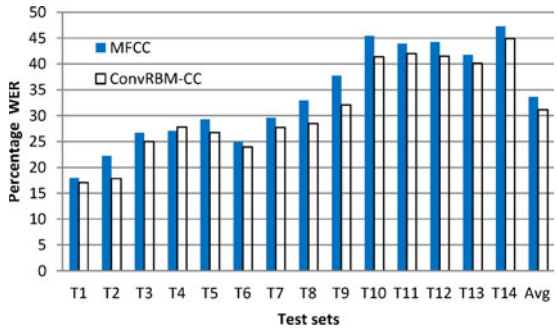

Fig. 10. Detailed evaluation of AURORA 4 test sets using MFCC and ConvRBM-CC features.

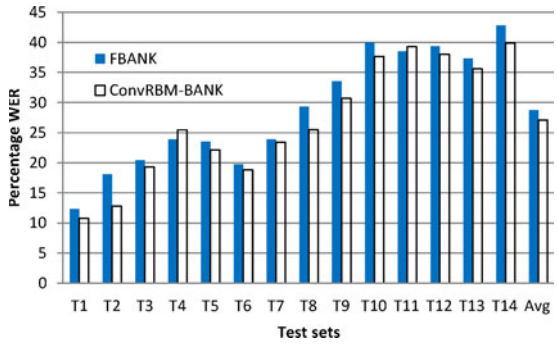

Fig. 11. Detailed evaluation of AURORA 4 test sets using FBANK and ConvRBM-BANK features.

absolute reduction of 1.25% on test set B and 1.73% on test set D over FBANK features. For test sets A and C, an absolute reduction of 1.57% and 3.85%, respectively, is achieved using ConvRBM-BANK features compared to FBANK features. High absolute reduction in % WER is obtained using system combination S1 $\oplus$ S2 of FBANK and ConvRBM-BANK trained systems, respectively.

We have also reported results on context-dependent DNN-HMM systems (denoted as CD-DNN-HMM) with force-aligned labels obtained from triphone GMM-HMM system. In both bi-gram and tri-gram 5K LM cases, ConvRBM-BANK showed improvements compared to FBANK features. With bi-gram 5K LM, 3% the relative improvement are achieved for channel distortion test sets. With tri-gram 5K LM, 1.4–12.94% relative improvements are achieved for test sets (however, less improvements for test set B and D). System combination for both LMs gives significant improvements compared to baseline FBANK systems. This shows that complementariness of both filterbanks further helps for robust speech recognition.

Detailed evaluations of AURORA 4 test sets are shown in Figs. 10 and 11 for CI-DNN-HMM systems. From Fig. 10, we can see that, in all test conditions, ConvRBM-CC features are performing better than MFCC except T4 test set. ConvRBM-FBANK features also perform better than FBANK features except restaurant noise conditions (i.e., test sets T4 and T11). To justify the improvements in ASR task using AURORA 4 database, we have investigated log-spectrum amplitude

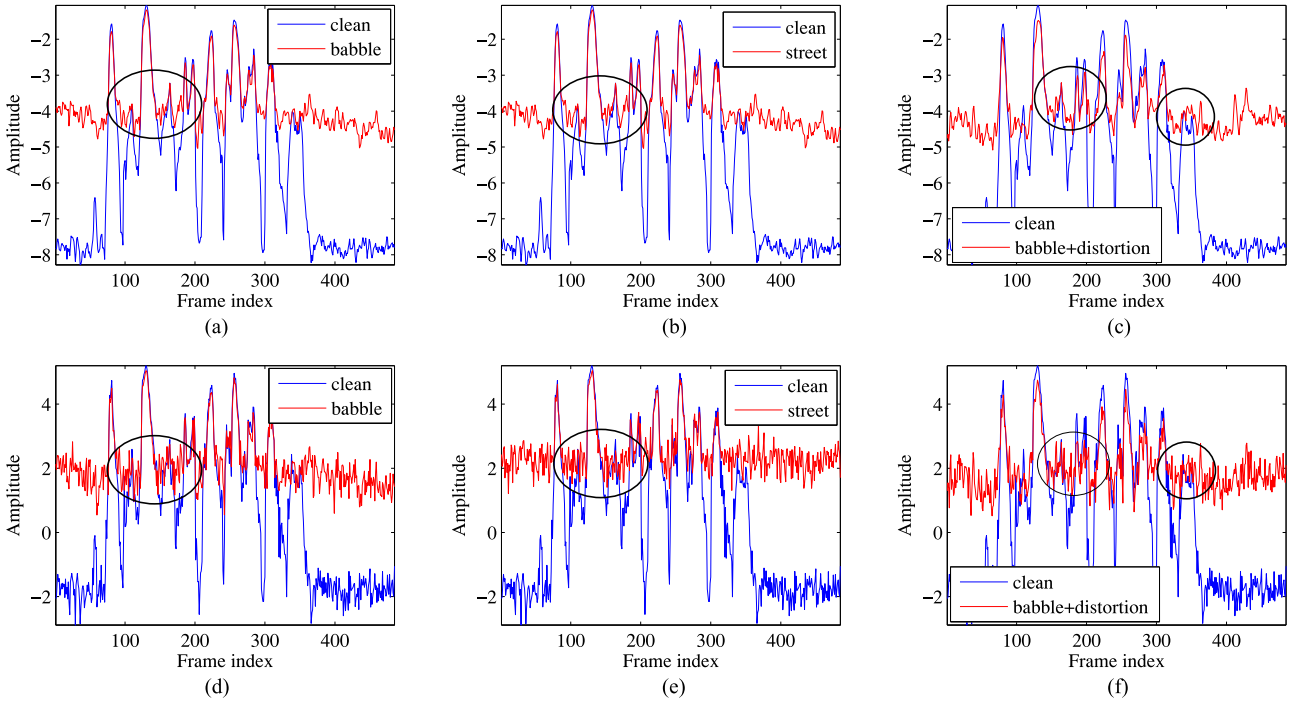

Fig. 12. Comparison of spectrum for one subband filter on AURORA 4 test sets. (a)-(c) ConvRBM generated spectrum, (d)-(f) Mel spectrum. Highlighted regions shows examples of regions distorted due to noise.

TABLE VIII
COMPARISON OF % WER FOR AURORA 4 DATABASE USING DIFFERENT APPROACHES

Approaches	A	B	C	D	Avg
Our approach (S4, 3 layers)	9.68	14.81	19.58	29.69	18.44
Our approach (S3 $\oplus$ S4, 3 layers)	9.47	13.91	18.85	28.52	17.69
DNN (5 layers) [44]	10.6	16.4	15.8	26.6	20.3
CNN (5 layers, 1-D filters) [44]	9.5	14.8	14.6	23.6	18.2
PNS-CNN (2-D filters) [45]	7.4	13.4	12.8	24.7	17.8
CNN (2-D filters) [42]	5.1	8.8	8.5	20.1	13.4
AD Maxout CNN [43]	4.0	7.8	6.7	14.9	10.5

TABLE IX
RESULTS OF TIMIT PHONE RECOGNITION TASK USING AURORA 4 TRAINED CONVRBM IN % PER

ConvRBM training database	Test
TIMIT	22.8
AURORA 4	23.6

variations during the time for three test conditions (namely, babble, street, and babble+distortion) against a clean log-spectrum as a reference. Log spectrum for a subband filter with CF 2.16 kHz is plotted for ConvRBM in Fig. 12(a)–(c) and for Mel spectrum in Fig. 12(d)–(f). Since we have not applied any mean-variance normalization on the spectrum, there is a difference in amplitude level in all noisy spectra. It is clearly seen that Mel spectrum is very much affected by noise (examples of selected regions are marked in Fig. 12(d)–(f)) in all test conditions compared to ConvRBM spectrum. Hence, ConvRBM trained filterbank is likely to reduce noise distortions which may improve ASR performance in degraded conditions as well.

Comparison of our approach using a bi-gram 5K LM with other approaches (specifically, convolutional networks) is given in Table VIII. Our supervised back-end is DNN with 3 hidden layers as we have discussed in Section IV-C. Many recent architectures such as [42] and [43] are able to perform quite well for AURORA 4 task. Our future work is to use these recent convolutional networks as the back-end for acoustic modeling to further improve the performance.

E. Cross-Domain Experiments

We have also experimented using a TIMIT trained ConvRBM filterbank for AURORA 4 speech recognition task and vice-versa to see whether ConvRBM subband filters are a generalized representation of auditory processing? We have changed the filterbanks in the front-end to extract features from the TIMIT and AURORA 4 databases. Once the features were extracted, acoustic modeling was performed as per the TIMIT and AURORA 4 task. Following are the results of cross-domain experiments:

Table IX shows that relative % PER of subband filters of AURORA 4 database is 3.4% higher (an absolute difference of 0.8%) compared to subband filters of TIMIT database. However, there is no significant difference in % PER when we have used subband filters trained on the different database (along with different training conditions). Table X shows that % WER of all AURORA 4 test sets are similar for subband filters of

TABLE X
RESULTS OF AURORA 4 TASK USING TIMIT TRAINED CONVRBM

ConvRBM training database	A	B	C	D	Avg
AURORA 4	9.68	14.81	19.58	29.69	18.44
TIMIT	9.9	14.92	19.3	29.18	18.52

both databases. These results also explain that even with small amount of TIMIT training data, we can achieve similar gains on the larger AURORA 4 database.

VI. SUMMARY AND CONCLUSIONS

In this paper, we have presented the theory and learning algorithm of our proposed unsupervised model to learn an auditory filterbank. The novelty of our proposed model lies in learning a filterbank without the need of windowing a speech signal and thus, alleviating artifacts of block-based processing for filterbank learning. Use of rectified linear units helps in learning sparse representations and keeping the very low mean value of hidden units. Weights of the model represent time-domain subband filters which are gammatone-like and corresponding frequency-domain filters are bandpass in nature. Our proposed learned filterbanks resemble other auditory filterbanks and hence, can represent frequency tuning in the cochlea. We have experimented on various standard datasets including large datasets and multi-condition dataset such as AURORA 4. Experimental results presented in this paper shows that learning the filterbank from speech data indeed helps in reducing WER in all test conditions. Complementary information in both the filterbanks helps to further reduce error rates in ASR. Cross-domain experiments on the TIMIT and AURORA 4 databases shows that ConvRBM is able to more general representations of the speech signals. Our future research efforts will be directed towards modeling auditory cortex using 2-D ConvRBM and possible learning of spectro-temporal receptive fields by extending our recent work in [46].

APPENDIX A
DERIVATION OF GRADIENT OF WEIGHTS

The gradient for weights in each kth group is given as,

$$\frac{\partial}{\partial w_r^k} E(\mathbf{x}, \mathbf{h}) = -\frac{\partial}{\partial w_r^k} \left[\sum_{k=1}^{K} \sum_{j=1}^{l} \sum_{r=1}^{m} \left(h_j^k w_r^k x_{j+r-1} \right) \right]. \quad (17)$$

For $r = 1$ to m, eq. (17) can be written as a set of equations as follows:

$$\begin{aligned} \frac{\partial}{\partial w_1^k} E(\mathbf{x}, \mathbf{h}) &= \sum_{j=1}^{l} \left(h_j^k x_j \right), \\ &\vdots \\ \frac{\partial}{\partial w_m^k} E(\mathbf{x}, \mathbf{h}) &= \sum_{j=1}^{l} \left(h_j^k x_{j+m-1} \right). \end{aligned} \quad (18)$$

Since $\mathbf{W}^k = [w_1^k, w_2^k, \ldots, w_m^k]$ is a weight vector, we can write this as a gradient of weight vector $\mathbf{W}^k$,

$$\begin{aligned} \therefore \left[\frac{\partial}{\partial w_1^k} E(\mathbf{x}, \mathbf{h}), \ldots, \frac{\partial}{\partial w_m^k} E(\mathbf{x}, \mathbf{h}) \right] &= \frac{\partial}{\partial \mathbf{W}^k} E(\mathbf{x}, \mathbf{h}), \\ &= -\sum_{j=1}^{l} h_j^k x_{j+r-1}, \\ &= -\text{conv}(\mathbf{x}, \tilde{\mathbf{h}^k}). \end{aligned} \quad (19)$$

REFERENCES

[1] R. M. Stern and N. Morgan, "Features based on auditory physiology and perception," in *Tech. Noise Robustness Autom. Speech Recognit.*, T. Virtanen, B. Raj, and R. Singh, Eds. New York, NY, USA: Wiley, 2012, pp. 193–227.

[2] R. Stern and N. Morgan, "Hearing is believing: Biologically inspired methods for robust automatic speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 34–43, Nov. 2012.

[3] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[4] G. Hinton, "Where do features come from?" *Cogn. Sci.*, vol. 38, no. 6, pp. 1078–1101, 2014.

[5] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. 31st Int. Conf. Mach. Learn.*, 2014, pp. 1764–1772.

[6] D. Amodei, R. Anubhai, E. Battenberg, "Deep speech 2: End-to-end speech recognition in English and Mandarin," in *Proc. 33rd Int. Conf. Mach. Learn. (ICML)*, NY, USA, 2015.

[7] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *2015 IEEE Workshop Autom. Speech Recognit. Underst.*, Dec. 2015.

[8] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNs," in *Proc. INTERSPEECH*, Sep. 2015, pp. 1–5.

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

[10] D. Palaz, M. Magimai-Doss, and R. Collobert, "Convolutional neural networks-based continuous speech recognition using raw speech signal," in *Proc. 40th Int. Conf. Acoust., Speech Signal Process.*, 2015, pp. 4295–4299.

[11] Z. Tüske, P. Golik, R. Schlüter, and H. Ney, "Acoustic modeling with deep neural networks using raw time signal for LVCSR," in *Proc. INTERSPEECH*, Sep. 2014, pp. 890–894.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[13] J. Lee *et al.*,"Speech feature extraction using independent component analysis," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, 2000, pp. 1631–1634.

[14] M. S. Lewicki, "Efficient coding of natural sounds," *Nature Neurosci.*, vol. 5, no. 4, pp. 356–363, 2002.

[15] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Comput.*, vol. 12, no. 2, pp. 337–365, Feb. 2000.

[16] A. Bertrand, K. Demuynck, V. Stouten, and H. V. hamme, "Unsupervised learning of auditory filter banks using non-negative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2008, pp. 4713–4716.

[17] Y.-H. Chiu, B. Raj, and R. Stern, "Learning-based auditory encoding for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2010, pp. 4278–4281.

[18] S. Chatterjee and W. Kleijn, "Auditory model-based design and optimization of feature vectors for automatic speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 6, pp. 1813–1825, Aug. 2011.

[19] N. Jaitly and G. Hinton, "Learning a better representation of speech soundwaves using restricted Boltzmann machines," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2011, pp. 5884–5887.

[20] E. Smith and M. S. Lewicki, "Efficient coding of time-relative structure using spikes," *Neural Comput.*, vol. 17, no. 1, pp. 19–45, Jan. 2005.

[21] E. C. Smith and M. S. Lewicki, "Efficient auditory coding," *Nature*, vol. 439, no. 7079, pp. 978–982, 2006.
[22] J. Anden and S. Mallat, "Deep scattering spectrum," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4114–4128, Aug. 2014.
[23] H. B. Sailor and H. A. Patil, "Filterbank learning using convolutional restricted Boltzmann machine for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2016, pp. 5895–5899.
[24] H. Lee, R. B. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 609–616.
[25] H. Lee, P. T. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Proc. 23rd Annu. Conf. Neural Inf. Process. Syst.*, 2009, pp. 1096–1104.
[26] H. B. Sailor and H. A. Patil, "Unsupervised learning of temporal receptive fields using convolutional RBM for ASR task," in *Proc. Eur. Signal Process. Conf.*, Aug./Sep. 2016, pp. 873–877.
[27] J. B. Allen, "How do humans process and recognize speech?" *IEEE Trans. Speech Audio Process.*, vol. 2, no. 4, pp. 567–577, Oct. 1994.
[28] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.
[29] A. Fischer and C. Igel, "An introduction to restricted Boltzmann machines," in *Proc. Progress Pattern Recognit., Image Anal., Comput. Vis., Appl.*, 2012, pp. 14–36.
[30] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, 2002.
[31] J. Qi, D. Wang, Y. Jiang, and R. Liu, "Auditory features based on gammatone filters for robust speech recognition," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2013, pp. 305–308.
[32] Y. Hoshen, R. J. Weiss, and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2015, pp. 4624–4628.
[33] X. Yang, K. Wang, and S. Shamma, "Auditory representations of acoustic signals," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 824–839, Mar. 1992.
[34] Garofolo *et al.*, "DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1–1.1," *NASA STI/Recon, Hampton, VA, USA, Tech. Rep. No. 93*, p. 27403, 1993.
[35] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proc. Workshop Speech Natural Lang.*, 1992, pp. 357–362.
[36] N. Parihar and J. Picone, "AURORA working group: DSR front end LVCSR evaluation," *Inst. for Signal and Inf. Process.*, Mississippi State University, MS, USA, Tech. Rep. AU/384/02, 2002.
[37] K.-F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Trans. Acoust., Speech Signal Process.*, vol. 37, no. 11, pp. 1641–1648, Nov. 1989.
[38] D. Povey *et al.*, "The KALDI speech recognition toolkit," in *Proc. IEEE Workshop Autom. Speech Recognit. Underst.*, Dec. 2011.
[39] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum Bayes risk decoding and system combination based on a recursion for edit distance," *Comput. Speech Lang.*, vol. 25, no. 4, pp. 802–828, 2011.
[40] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep. 2009, pp. 2146–2153.
[41] D. Palaz, R. Collobert, and M. M. Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks," in *Proc. INTERSPEECH*, 2013, pp. 1766–1770.
[42] J.-T. Huang, J. Li, and Y. Gong, "An analysis of convolutional neural networks for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2015, pp. 4989–4993.
[43] S. J. Rennie, V. Goel, and S. Thomas, "Annealed dropout training of deep networks," in *Proc. IEEE Spoken Lang. Technol. Workshop.*, 2014, pp. 159–164.
[44] V. Mitra *et al.*,"Evaluating robust features on deep neural networks for speech recognition in noisy and channel mismatched conditions," in *Proc. INTERSPEECH*, 2014, pp. 895–899.
[45] S.-Y. Chang and N. Morgan, "Robust CNN-based speech recognition with gabor filter kernels," in *Proc. INTERSPEECH*, 2014, pp. 905–909.
[46] H. B. Sailor and H. A. Patil, "Unsupervised deep auditory model using stack of convolutional RBMs for speech recognition," in *Proc. INTERSPEECH*, San Francisco, CA, USA, Sep. 2016, pp. 3379–3383.

Hardik B. Sailor received the B.E. degree from Govt. Engg. College (GEC), Surat, India, in 2010, and the M.Tech degree from Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT), Gandhinagar, India, where he is working toward the Ph.D. degree.

His research interest includes modeling auditory processing, unsupervised deep learning and speech recognition. His main research is focused on developing unsupervised deep learning to model human speech perception. He is a Student Member of the IEEE Signal Processing Society and International Speech Communication Association (ISCA).

Hemant A. Patil received the B.E. degree from North Maharashtra University, Jalgaon, India, in 1999, the M.E. degree from Swami Ramanand Teerth Marathwada University, Nanded, India, in 2000 and Ph.D. degree from the Indian Institute of Technology Kharagpur, Kharagpur, India, in 2006.

From February. 2007 to March 2012, he served as Assistant Professor and since April 2012, he has been an Associate Professor at Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT) Gandhinagar, India. His research interests include speech processing, pattern recognition, voice biometrics, wavelet signal processing, and infant cry analysis. He has coedited a book with Dr. Amy Neustein (Editor-in-Chief, IJST, Springer-Verlag) on Forensic Speaker Recognition: Law Enforcement and Counter-Terrorism, Springer-Verlag, New York, USA. Dr. Patil is PI/Co-PI for three DeitY and two DST sponsored projects. Dr. Patil is an affiliate member of IEEE SLTC, IEEE, IEEE Signal Processing Society, IEEE Circuits and Systems Society (Awards), and International Speech Communication Association.