Informative Spectro-Temporal Bottleneck Features for Noise-Robust Speech Recognition

Shuo-Yiin Chang 1,2, Nelson Morgan 1,2

1 EECS Department, University of California-Berkeley, Berkeley, CA, USA
2 International Computer Science Institute, Berkeley, CA, USA
shuoyiin@eecs.berkeley.edu, morgan@icsi.berkeley.edu

Abstract

Spectro-temporal Gabor features based on auditory knowledge have improved word accuracy for automatic speech recognition in the presence of noise. In previous work, we generated robust spectro-temporal features that incorporated the power normalized cepstral coefficient (PNCC) algorithm. The corresponding power normalized spectrum (PNS) is then processed by many Gabor filters, yielding a high dimensional feature vector. In tandem processing, an MLP with one hidden layer is often employed to learn discriminative transformations from front end features, in this case Gabor filtered power spectra, to probabilistic features, which are referred as PNS-Gabor MLP. Here we improve PNS-Gabor MLP in two ways. First, we select informative Gabor features using sparse principle component analysis (sparse PCA) before tandem processing. Second, we use a deep neural network (DNN) with bottleneck structure. Experiments show that the high-dimensional Gabor features are redundant. In our experiment, sparse principal component analysis suggests Gabor filters with longer time scales are particularly informative. The best of our experimental modifications gave an error rate reduction of 15.5% relative to PNS-Gabor MLP plus MFCC, and 41.4% better than an MFCC baseline on a large vocabulary continuous speech recognition task using noisy data.

Index Terms: LVCSR, spectro-temporal feature, robust speech recognition, deep neural network, sparse principal component analysis

1. Introduction and Previous work

When they are paying attention, humans speech recognition is far better than our automatic systems, particularly in the presence of noise. This often inspires researchers to seek more robust features based on auditory models. In the past decade, physiological experiments on different mammalian species have revealed that the neurons in the primary auditory cortex are sensitive to particular spectro-temporal patterns referred to as spectro-temporal receptive fields (STRFs) [1]. Based on these experimental results, spectro-temporal features, which serve as a model for STRFs, have been applied to ASR. Several studies have successfully incorporated Gabor function approximations into ASR [2][3][4]. In general, these approaches define a series of spectral, temporal, and spectral-temporal modulation filters which can be seen as roughly modeling neuron firing patterns for particular spectro-temporal signal components, but in any case they provide a large range of transformations of the time-frequency plane. Purely temporal features such as TRAPS [5] and HATS [6] can be regarded as special cases of spectro-temporal features. Spectro-temporal features have been successfully used for recognition of both clean and noisy speech, which provides a different solution to most existing robustness methods focusing on compensating the difference of clean and noisy speech in the models [7][8] or the features [9][10][11]. In [12], a more robust spectro-temporal representation incorporating key parts of the power normalized cepstral coefficients (PNCCs) algorithm [10] was employed. We referred to this feature set as power-normalized spectrum Gabor (PNS-Gabor).

However, due to the use of a large number of Gabor filters, the feature dimension is quite high, and it is likely that many of the features are redundant. Tsai and Morgan [13,14] showed that a small number of filters dominated the performance for the speech activity detection task. In this paper, we employ sparse PCA [15][16] as a feature selection tool to find informative Gabor features, which maximize data variance. Sparse PCA extended classical PCA by maximizing data variance using sparse principal vectors. The property makes it easy to interpret and also is useful for feature selection to discard uninformative components.

In tandem processing, an MLP with single hidden layer is commonly employed to transform front end values (e.g., the Gabor filtered time-frequency plane) to probabilistic features [17]. An MLP with a bottleneck layer is also commonly used [18][19]. However, it is difficult benefit from increasing the number of hidden layers without using some kind of initialization, particularly because the layers far from the target are little changed by the usual stochastic gradient learning algorithms. Hinton et al [20] proposed an unsupervised training algorithm based on restricted Boltzman machine (RBM) moving the parameters to a good initial. Pre-trained deep neural networks have been used to decrease WER [21][22]. Here, we exploit pre-trained deep neural network using Gabor input to improve robustness for recognition of noisy speech.

Comparison of the proposed method and the previous PNS-Gabor MLP(3l) [12] is shown in Fig. 1.

![Figure 1 Comparison of proposed method (top path) and our previous work (bottom path).](image-url)

Power normalized spectrum
Gabor filters
Sparse principle component analysis
Bottleneck Deep Neural Network
Informative PNSGb-BN DNN

Gabor filters
Conventional 3-layers MLP

PNNSGb-MLP(3l)

Index Terms: LVCSR, spectro-temporal feature, robust speech recognition, deep neural network, sparse principal component analysis

1. Introduction and Previous work

When they are paying attention, humans speech recognition is far better than our automatic systems, particularly in the presence of noise. This often inspires researchers to seek more robust features based on auditory models. In the past decade, physiological experiments on different mammalian species have revealed that the neurons in the primary auditory cortex are sensitive to particular spectro-temporal patterns referred to as spectro-temporal receptive fields (STRFs) [1]. Based on these experimental results, spectro-temporal features, which serve as a model for STRFs, have been applied to ASR. Several studies have successfully incorporated Gabor function approximations into ASR [2][3][4]. In general, these approaches define a series of spectral, temporal, and spectral-temporal modulation filters which can be seen as roughly modeling neuron firing patterns for particular spectro-temporal signal components, but in any case they provide a large range of transformations of the time-frequency plane. Purely temporal features such as TRAPS [5] and HATS [6] can be regarded as special cases of spectro-temporal features. Spectro-temporal features have been successfully used for recognition of both clean and noisy speech, which provides a different solution to most existing robustness methods focusing on compensating the difference of clean and noisy speech in the models [7][8] or the features [9][10][11]. In [12], a more robust spectro-temporal representation incorporating key parts of the power normalized cepstral coefficients (PNCCs) algorithm [10] was employed. We referred to this feature set as power-normalized spectrum Gabor (PNS-Gabor).

However, due to the use of a large number of Gabor filters, the feature dimension is quite high, and it is likely that many of the features are redundant. Tsai and Morgan [13,14] showed that a small number of filters dominated the performance for the speech activity detection task. In this paper, we employ sparse PCA [15][16] as a feature selection tool to find informative Gabor features, which maximize data variance. Sparse PCA extended classical PCA by maximizing data variance using sparse principal vectors. The property makes it easy to interpret and also is useful for feature selection to discard uninformative components.

In tandem processing, an MLP with single hidden layer is commonly employed to transform front end values (e.g., the Gabor filtered time-frequency plane) to probabilistic features [17]. An MLP with a bottleneck layer is also commonly used [18][19]. However, it is difficult benefit from increasing the number of hidden layers without using some kind of initialization, particularly because the layers far from the target are little changed by the usual stochastic gradient learning algorithms. Hinton et al [20] proposed an unsupervised training algorithm based on restricted Boltzman machine (RBM) moving the parameters to a good initial. Pre-trained deep neural networks have been used to decrease WER [21][22]. Here, we exploit pre-trained deep neural network using Gabor input to improve robustness for recognition of noisy speech.

Comparison of the proposed method and the previous PNS-Gabor MLP(3l) [12] is shown in Fig. 1.
2. Proposed Method

2.1. PNS Gabor features

Gabor-filtered power spectra based on mel-bands are easily corrupted by noise. While no one has a general solution for this problem, we found in [12] that using PNCC generated a more robust spectro-temporal representation. We referred to it as the power normalized spectrum and the corresponding output filtered by Gabor filters as PNS-Gabor features. PNCC [10] differs from MFCC in three aspects: (1) gammatone filter (2) medium-duration bias subtraction (3) power-law nonlinearity. An example of mel spectrum and power normalized spectrum is shown in Fig. 2 illustrating greater insensitivity to noise for the power normalized spectrum. PNS Gabor features are obtained by convolving two dimensional modulation filters and power normalized spectrum. To generate filters serving as model for spectro-temporal receptive fields (STRFs), we multiply a complex sinusoid with a Hanning envelope. The complex sinusoid has an analytic form:

\[
s(n,k) = \exp[i\omega_n(n-n_0)+i\omega_k(k-k_0)]
\]

while the Hanning envelope is given (with \(W_n\) and \(W_k\) denote window length)

\[
h(n,k) = 0.5 - 0.5 \cdot \cos(\frac{2\pi(n-n_0)}{W_n} + 1) \cdot \cos(\frac{2\pi(k-k_0)}{W_k} + 1)
\]

By tuning parameters of spectral and temporal modulation frequency, Gabor functions have different extent and orientation for a given number of oscillations under the envelope as used in this study. The Gabor filter bank used here has been adapted from [4]. The 59 Gabor filters emphasizing on different temporal and spectral modulation frequencies are used. However, the filters with a large spectral extent result in high correlations between frequency channels. Hence, a subset of the possible combinations are used to avoid high correlations of feature components, resulting in an 814-dimensional feature. In [12] we found that medium-duration bias subtraction and Gabor filtering are two key factors for PNS-Gabor when used for Aurora2 and noisy WSJ.

2.2. Feature Selection via Sparse PCA

While a number of experiments have shown the utility of using spectro-temporal Gabor filters for automatic speech recognition, some features generated by Gabor filters may not be informative. Here we propose an approach to select informative Gabor features using Gabor filters may not be informative. Here we propose an approach to select informative Gabor features using sparse principle component analysis. Based on the leading sparse principal vectors, we discard Gabor features that might prove to be less useful.

Classical PCA is a widely used tool for dimensionality reduction, providing a linear combination of “all” features that maximizes data variance. However, in general, the principal vectors are dense (i.e., the entries are non-zero), which makes the results difficult to interpret. Sparse PCA on the other hand, solves the same problem using “sparse” principal vectors. The objective function of sparse PCA is modified from classical PCA with an \(\ell_1\)-norm penalty, so that sparse principal vectors are favored. The objective function is represented as:

\[
\max_v \Sigma v - \rho \|v\|_1, \text{ subject to } \|v\|_2 = 1
\]

where \(\rho\) is a non-negative parameter controlling the sparsity of principal vector, \(\Sigma\) is the empirical covariance matrix calculated from observations \(v\) and \(v\) is the leading sparse principal vector. The first term in (3) is the objective function of classical PCA. Formula (3) can be rewritten as:

\[
\max_v \text{Tr}(\Sigma vv^T) - \rho (\|v\|_1)
\]

subject to \(\|v\|_2 = 1\) (4)

To get a robust interpretation, we reformulate the \(\ell_1\)-norm penalty:

\[
\rho (\|v\|_1) = \min U \text{Tr}(VV^T), -\rho \leq U_{ij} \leq \rho
\]

We obtain formula (6) by applying (5) to (4):

\[
\max \min_v \text{Tr}(\Sigma vv^T) - \rho (\|v\|_1)
\]

subject to \(\|v\|_2 = 1\) (5)

By switching the trace of the product in (6), it is equivalent to:

\[
\max \min_v (\Sigma + U vv^T) : \|v\|_2 = 1, -\rho \leq U_{ij} \leq \rho
\]

The dual problem then can be shown as:

\[
\min \max_v (\Sigma + U vv^T) : \|v\|_2 = 1, -\rho \leq U_{ij} \leq \rho
\]

or \(\lambda_{\max}(\Sigma + U) - \rho \leq U_{ij} \leq \rho\) (8)

where \(\lambda_{\max}(\Sigma + U)\) means maximum eigenvalue of \(\Sigma + U\)

Based on (9), we can solve the original problem by searching for the minimum of the largest eigenvalue of the covariance matrix with a noise matrix. Sparsity is accomplished by eliminating small values imposed on the empirical covariance matrix by component-wise noise bounded by \(\rho\). In this paper, we apply the Augmented Lagrangian Method (ALM) algorithm to speed the search for \(U\), as derived in [16].

We compute only the leading sparse eigenvector, which is sufficient for selecting informative features in our experiment. The feature variables corresponding to zero entries in the sparse principal vector are less useful to account for data variance. Hence, only informative features corresponding to non-zero entries are selected. The dimensionality of the selected features is much smaller than the original.
2.3. Bottleneck features using Pre-trained Deep Neural Network

A bottleneck feature vector commonly is generated by a narrow-dimensional layer in the middle of a five-layer MLP. While using even more layers can improve the network, better initialization can be a significant help when using more layers. Restricted Boltzman machine (RBM) pre-training [22], which is unsupervised, is one method that has been introduced to initialize the parameters of neural network. For completeness, we briefly remind the reader of the main idea here.

RBM is a bipartite graph modeling the joint distribution of a layer of stochastic visible units and hidden units. Only visible-hidden connections are allowed. For Bernoulli-Bernoulli RBM, the joint distribution is defined as:

$$P(v, h) = \frac{e^{-E(v, h)}}{Z}, E(v, h) = -h^T v - c^T h - v^T Wh$$

where $Z$ is the normalization term, $E(v, h)$ is energy function, $b$ and $c$ are the bias for visible layer $v$ and hidden layer $h$ while $W$ is a symmetric weight matrix between $v$ and $h$. Based on (10), the posterior of hidden units can be obtained as:

$$P(h = 1|v) = \text{sigmoid}(c + v^T W)$$

Formula (11) is essentially the forward propagation procedure of neural network. Hence, RBM can be applied to model each pair of layers of a neural network. The input layer usually consists of real-valued variables, so a Gaussian-Bernoulli RBM is employed. The energy function for this is revised as:

$$E(v, h) = \frac{1}{2} (v - b)^T (v - b) - c^T h - v^T Wh$$

while (11) still holds. The parameters of the MLP, therefore, get good initialization by maximizing the log likelihood of the RBM. Training of RBM can be found in [21].

3. Experimental Setup

The proposed approach was evaluated using “re-noised” Wall Street Journal (WSJ) speech and far-field meeting corpus. For WSJ, we started out with clean data taken from 77.8 hours of the WSJ1 dataset (284 speakers) for training and 0.8 hours of the WSJ-eval94 dataset (20 speakers) for testing. Estimated additive and channel noise from degraded recordings was applied to both training and testing dataset using “renoiser” tool [23]. Designed for use in the DARPA RATS project, the system analyzes data from RATS rebroadcast example signals (in this case, LDC2011E20) to estimate the noise characteristic including SNRs and frequency-shifts; the original data is described in [24] and consists of a variety of continuous speech sources that have been transmitted and received over 8 different radio channels, resulting in significant signal degradations. The 8 radio channel characteristics were specified in [12]. We applied the same noise characteristics to WSJ data and it is referred as “re-noised WSJ”. The results reported here are WERs averaging over clean and 8 noisy channels. For the far-field part of the meeting data, we used a single distant microphone channel from a dataset of spontaneous meeting speech recorded at ICSI [25]. Our training set was based on the meeting data used for adaptation in the SRI-ICSI meeting recognition system [26]. For the test set we used the ICSI meetings drawn from the NIST RT eval sets [27]. The resulting training set was 20 hours long (26 speakers) and the testing set was 1 hour of data (18 speakers). We used the HTK toolkit [28] for both training and decoding with both databases. The acoustic models were cross-word triphones estimated with maximum likelihood. For re-noised WSJ, the resulting triphone states were clustered to 5000 tied states, each of which was modeled by 32-component of Gaussian mixture model. We used version 0.6 of the CMU pronunciation dictionary and the standard 5k bigram language model created at Lincoln Labs for the 1992 evaluation. For the single distant microphone (SDM) portion of the far-field meeting corpus, 2500 states were modeled by Gaussian mixture models with 8 components per state. The 10k trigram SRI LM [29] was trained by interpolating Switchboard, meetings, Fisher, Hub4-LM96, and TDT4. To be compatible with the SRI LM, we used the SRI pronunciation dictionary.

All the networks were trained with a temporal context window of 9 successive frames. The output layer consisted of 41 context-independent phonetic targets. We used a learning rate of .002 in the RBM pre-training step. For back propagation following the pre-training, we began with a learning rate of .008 and reduced the learning rate by factors of two once cross-validation indicated limited progress with each learning rate, and continued until cross-validation showed essentially no further progress. We took the logarithm of the posterior probability features generated from the output of the MLP, and reduced the dimensionality via PCA to 25. For bottleneck features, the size of the bottleneck layer was set to 25. In both cases, MFCCs were concatenated with the MLP-based features, resulting in a 64-dimensional feature vector. Means and variances were normalized per utterance before HMM training and testing for all the features in this paper.

4. Results and Discussion

We first compared different configurations of neural network using 814 dimensional PNS-Gabor inputs with 9 frames, or 7326 inputs. Next, feature selection using sparse PCA was studied. Finally, we compared the result with state-of-the-art features and showed that the informative feature selection is fairly invariant to differences between the two corpora used.

In Fig 3, we show results for several different configurations of neural network using PNS-Gabor input on re-noised WSJ: a three-layer MLP, a four-layer deep neural network (DNN), a five-layer bottleneck DNN and a six-layer bottleneck DNN. The 3-layer MLP used one large hidden layer. For the 4-layer DNN, two 500 unit hidden layers were used to model phone classes. In the 5-layer DNN, these 2 hidden layers were separated by the small (25) bottleneck layer. The 6-layer bottleneck configuration included an additional 500-unit layer after the first hidden layer, so that the initial transformations are similar to those of the 4-layer.

Figure 3: Re-noised WSJ WERs of 814-d PNS-Gabor input using four different neural network architectures
network, followed then by the bottleneck portion. The number of parameters remained quite similar, ranging from 3.7M to 3.9M. The specific configurations are described in Fig. 3.

As shown in Fig 3, the 4-layer DNN and the 6-layer bottleneck DNN performed significantly better. Comparing the 6-layer bottleneck DNN to the 4-layer DNN, bottleneck compression slightly decreases WER. Another observation is that RBM always helps when training deep neural networks, as has been observed by many others. The 6-layer bottleneck DNN with RBM pre-training was best in our experiments, so we used this architecture for the rest of the work reported here.

In Table 1, we compare 814 dimensional PNS-Gabor features and the dimensionally reduced “informative” PNS-Gabor feature selected via sparse PCA. In this experiment, $\rho$ was set to .008, resulting in 271 dimensional features. The size of hidden layers was increased to 1000 so that total number of parameters was comparable for the comparison. As shown in Table 1, the sparse-PCA selection was 6.7% relative better than using all of the features. Aside from this improvement, the approach also made feature generation more efficient. The sparse PCA approach was furthered compared with classical PCA. Results also suggest that using the reduced “informative” features is better than using the linear combination of all features obtained by classical PCA.

Using the sparse principal vector, we observed the suggested importance of Gabor filters in Fig. 4. The darker areas represent the degree of importance (at least in terms of variance) based on how many channels generated by the filter were selected. As shown in Fig 4, filters with low temporal modulation frequency appear to be useful for maximizing data variance. The features emphasizing on temporal modulation frequency of 0, 2, 4 and 3.9 Hz account for 94.1% of the weights of the leading sparse principal vector. The corresponding filter lengths are roughly 1, 0.7 and 0.5 sec. It suggests that Gabor filters with longer time scales are particularly informative for ASR, at least for these tests.

In Table 2, we compared the proposed feature with MFCC baseline and state-of-the-art features: the so-called “advanced frontend” (AFE) [10] and PNCC [11] (For the matched clean training-testing condition, WER for MFCC was 6.7%, but here we focus on the multi-conditional training task). Our previous feature set [12] was shown in column (4). A six-layer bottleneck DNN was employed in column (5) while informative selection was further applied in column (6). Overall, the proposed feature in (6) is 15.5% better than previous PNS-Gabor MLP(3) plus MFCC and 41.4% relative better than MFCC on re-noised WSJ. For far-field meeting corpus, we can reduce word error rate by 14% relative to MFCC.

### 5. Conclusions

Here we report our use of sparse PCA, informative PNS-Gabor features, and bottleneck deep neural networks using two large hidden layers following the input layer. The key factor distinguishing the proposed feature set from our earlier work is discarding uninformative shorter features. Non-linear dimension reduction from the bottleneck feature also gave slight improvement. Overall, the proposed feature is 15.5% relative better than previous PNS-Gabor MLP(3) plus MFCC and 41.4% relative better than MFCC on re-noised WSJ. For far-field meeting corpus, we can reduce word error rate by 14% relative to MFCC.

### 6. Acknowledgements

We thank Bernd Meyer, Dan Ellis for help in setting up the feature extraction and re-noised WSJ data. We thank to Chanwoo Kim and Richard Stern for use of PNCC. This material is based on work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. D10PC20024. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the DARPA or its Contracting Agent, the U.S. Department of the Interior, National Business Center, Acquisition & Property Management Division, Southwest Branch.

### Table 1: Informative and original PNS-Gabor input for re-noised WSJ

<table>
<thead>
<tr>
<th>Feature</th>
<th>WER</th>
<th>Dim.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNS-Gabor</td>
<td>31.02</td>
<td>814</td>
</tr>
<tr>
<td>PNS-Gabor, classical PCA</td>
<td>29.57</td>
<td>271</td>
</tr>
<tr>
<td>Informative PNS-Gabor</td>
<td>28.93</td>
<td>271</td>
</tr>
</tbody>
</table>

### Table 2: Re-noised WSJ WERs for state-of-the-art feature

<table>
<thead>
<tr>
<th>Feature</th>
<th>WER</th>
<th>Dim.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNS-Gabor</td>
<td>31.02</td>
<td>814</td>
</tr>
<tr>
<td>PNS-Gabor, classical PCA</td>
<td>29.57</td>
<td>271</td>
</tr>
<tr>
<td>Informative PNS-Gabor</td>
<td>28.93</td>
<td>271</td>
</tr>
</tbody>
</table>

### Table 3: WERs for far-field meeting corpus

<table>
<thead>
<tr>
<th>Feature</th>
<th>WER</th>
<th>Dim.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNS-Gabor</td>
<td>31.02</td>
<td>814</td>
</tr>
<tr>
<td>PNS-Gabor, classical PCA</td>
<td>29.57</td>
<td>271</td>
</tr>
<tr>
<td>Informative PNS-Gabor</td>
<td>28.93</td>
<td>271</td>
</tr>
</tbody>
</table>
7. References


