AN AUDITORY NEURAL FEATURE EXTRACTION METHOD FOR ROBUST SPEECH RECOGNITION

Wei Guo, Liqing Zhang, and Bin Xia

Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
w_guo@sjtu.edu.cn; zhang-lq@cs.sjtu.edu.cn; xiabin@cs.sjtu.edu.cn

ABSTRACT

This paper proposes a neural mechanism motivated system to extract noise resistant features for robust speech recognition. We use non-negative matrix factorization to construct two layers of auditory neurons which captures the essence of speech patterns. The responses of these neurons to speech are further processed to form an auditory neural cepstral coefficient (ANCC) representation for speech recognition. We test the robustness of ANCC feature on a 51-word corpus, with recognizers trained on clean speech in noisy conditions. Compared with MFCC, ANCC shows less performance degradation and achieves satisfactory recognition accuracies in both non-stationary noise and high noise level conditions.

Index Terms— speech recognition, robustness, feature extraction, auditory system

1. INTRODUCTION

Speech recognizers trained in clean conditions normally perform poorly in noisy environments, due to the mismatch between training and testing data. Numerous efforts have been made to improve either the front-ends [1][2][3] or the back-ends [4][5] of the recognizers for noisy environments. Certain progress have been made, but the overall performance is still not satisfactory, due to several issues. First, many methods impractically require noise information beforehand. Second, methods adapted to certain noise conditions may not generalize well to different noises or even different noise levels. And third, non-stationary noises and low signal-to-noise ratio (SNR) cases is still an open problem.

On the other hand, human auditory system tackles all these issues quite well. Motivated by the mechanism of human hearing, several researchers propose using computational models of human auditory system as front-end to extract noise resistant features, such as PLP model [2], and EIH model [3].These methods utilize computational models of cochlear filtering or hair cell firing to simulate peripheral auditory processing stages, and show certain noise-robust properties.

However, human auditory functions are mainly carried out in the auditory cortex, of which the working mechanism is not yet fully understood. Neurophysiologists found that some neurons in primary auditory cortex respond to specific kind of sound stimulus, such as certain frequency, and tone onset/offset. From the firing history of these neurons, the sound patterns they respond to can be recovered in time-frequency representation called spectro-temporal receptive field (STRF) [6], which is similar to the receptive fields of simple cells in visual system. Working as basic feature detectors, these neurons can also be connected to high level neurons in a hierarchical structure to detect more complex features such as timbres, and frequency modulation. Sparse coding technique was used in [7] to construct receptive fields for visual cells. A similar method was applied in [8] to obtain efficient code of natural sound using 1-dimensional bases, showing some auditory nerve tuning properties. Another sparse coding method called non-negative matrix factorization (NMF) [9] was used in [10] to construct a hierarchical representation of speech spectrogram. But little research has been conducted on computational models of STRFs, and its applications in speech recognition.

In this paper, we propose using NMF to calculate the receptive fields for two layers of auditory neurons, which extract noise resistant features. The layer-1 neurons detect basic spectro-temporal structures in 2-dimensional form, and the layer-2 neurons combine these structures into more complex patterns. This embodies the two stages of computational auditory scene analysis (CASA): segregating and grouping. Using speech signals as stimulus, the training process will produce neurons more sensitive to speech patterns than other patterns. The responses of the layer-2 neurons to the input sound can be further processed into a representation we called auditory neural cepstral coefficients (ANCC), which can work as features for speech recognition. We test the performance of ANCC features by using clean-speech-trained recognizers to recognize speech mixed with various noises in different SNRs without additional processing. Result shows ANCC performs robustly in both stationary and non-stationary noise cases, especially in strong noise conditions.

This paper is organized as follows. The proposed model is described in section 2. The experimental results are presented in section 3. In section 4, we discuss several issues about this model. Finally, conclusions are summarized in section 5.

2. PROPOSED METHOD

In this section, we first introduce the sparse coding technique. A two-layer network framework is propose to extract speech features. Finally an overall description of the system is given.

2.1. Non-negative matrix factorization

Non-negative matrix factorization (NMF) is a technique for decomposing non-negative data. It has been successfully used to discover part-based representation of visual objects, such as human faces [9]. NMF usually produces a sparse representation, which encodes data with only a few components, leaving most others silent. This property is closely related to the neural processing mechanism of brain. In this paper, we use an extended version of NMF, which explicitly controls the sparseness degree of the encoded data to provide more flexibility [11].

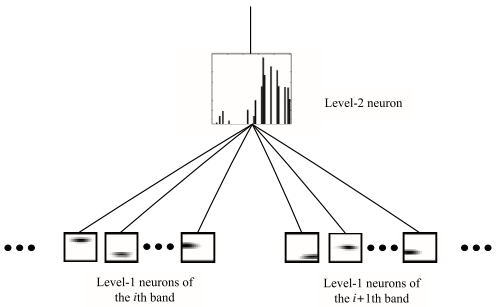


Fig. 1. Bottom: The STRFs of several layer-1 neurons from two different bands. The images indicate the spectro-temporal patterns they respond to. Top: The receptive field of a layer-2 neuron, who responds to an assembly of layer-1 neurons firing together. The bars indicate the connection weights. It is not sensitive to the input of neurons in other bands, which are ignored in this figure.

The objective of NMF is to find an approximate factorization of a non-negative data matrix V into non-negative factors W and H

$$V \approx WH \tag{1}$$

Consider columns of W to be basis vectors, then each column of V is approximately a linear combination of these bases, with coefficients defined by H columns. The sparseness of H columns suggests the number of bases we need to encode each data column.

The sparseness measurement of a vector x is defined using its L_1 and L_2 norms

$$sparseness(x) = \frac{\sqrt{n} - (\sum |x_i|)/\sqrt{\sum x_i^2}}{\sqrt{n} - 1} \tag{2}$$

where n is the length of x. Greater value indicates higher sparseness. For a given vector x, the desired sparseness is achieved by replacing it with the closest non-negative vector s, which preserves the L_2 norm of x but has an appropriate L_1 norm to fit the sparseness constraint. This can be accomplished by the projection operations described in [11].

In order to find the optimum factorization, W and H are calculated by a gradient descent algorithm. When using the squared error function as the optimization goal, W and H can be updated in a multiplicative form

$$H \leftarrow H \frac{W^T V}{W^T W H} \tag{3a}$$

$$W \leftarrow W \frac{V H^T}{W H H^T} \tag{3b}$$

So, within each iteration of the training process, first the sparseness of H is adjusted, then W and H are gradiently updated; this continues until the optimization goal is met.

2.2. Receptive fields of layer-1 neurons

Receptive fields of layer-1 neurons are calculated from the spectrogram of input speech. The input signal $x[n]$ is firstly pre-emphasized, then transformed into a time-frequency representation by short-time fourier transformation (STFT). In practice, we segment the signal samples with a moving Hamming window of length L_w, shifted every L_s, then apply $2N_f$-point FFT. The spectral magnitude is normalized to $[0, 1]$. So the resulting spectrogram $X(f, t)$ is a non-negative matrix of size $N_f \times N_t$, where N_t is the number of frames.

The spectrogram is equally divided into N_b sub-bands in frequency axis to demonstrate the band-pass property of peripheral auditory system. Although log-scale divisions or Mel-scale would physiologically makes more sense, in this model linear-scale division yields better results. And it will be further discussed in section 4. We then chop the spectrogram of each band into temporal frames, with frame length M_t and frame shift M_i. Therefore each frame is a matrix of size $M_f \times M_t$, where $M_f = N_f/N_b$. It represents the spectro-temporal structure of a short-time speech segment in a particular band. For the ith band, the frames from it are reshaped into column vectors and form a matrix $V_1(i)$. We apply NMF to each $V_1(i), (1 \le i \le N_b)$

$$V_1(i) \approx W_1(i) H_1(i) \tag{4}$$

which decomposes columns of $V_1(i)$ into linear combinations of column vectors in $W_1(i)$. These W_i^1 columns are the bases to represent speech frames in the ith band. If reshaped back to $M_f \times M_t$ matrices, they can be viewed as the STRFs of neurons, which specifically sensitive to the information from this band. The column number of $W_1(i)$, which we need to set before training, is the number of neurons for this band. Fig. 1 shows the STRFs of some layer-1 neurons., which clearly describe the spectro-temporal structures they respond to. Therefore the coefficient matrix $H_1(i)$ can be understood as neuron response activities. Sparseness constraint on $H_1(i)$ columns controls the number of neurons respond to each speech frame.

Now we can encode other speech signals using these auditory neurons. For example, to encode a speech waveform $y[n]$, we first need to transform it into sub-band matrix representation $V_1'(i), (1 \le i \le N_b)$. But calculating an $H_1'(i)$ matrix with minimized squared error involves another NMF process, which is computationally complex. We simply approximate the response of a neuron by the inner product of its STRF and the stimulus frame

$$H_1'(i) = W_1(i)^T V_1'(i) \tag{5}$$

This approximation is not only easy to implement, but also consistent to theoretical neural encoding mechanism. The inner-product response shares similar properties with the NMF response, such as structure description of the frame, and sparseness degree. Combining $H_1'(i)$ of all bands from low frequency to high frequency into a matrix H_1' provides a representation of the input waveform by the responses of all layer-1 neurons. This representation preserves spectrogram-related information such as spectral energy patterns, and temporal patterns. It also encodes fine spectral structures in each band, which are often ignored by conventional filter-bank methods. The patterns of these fine structures are important features to distinguish speech and noises. So we use another layer of neurons to discover these structures.

2.3. Receptive fields of layer-2 neurons

Layer-2 auditory neurons receive input from all neurons in the first layer, and respond to certain groups of neurons firing together. Again, these groups can be obtained by NMF training. We normalize the complete layer-1 neural response matrix H_1' to $[0, 1]$, and use it as the training data matrix V_2, without dividing sub-bands or temporal frames. Then NMF is applied to obtain

$$V_2 \approx W_2 H_2 \tag{6}$$

Similarly, the column number of W_2 is the neuron number, and the columns are the receptive fields, describing the groups of neurons they respond to. Fig. 1 also shows one neuron from this layer, and

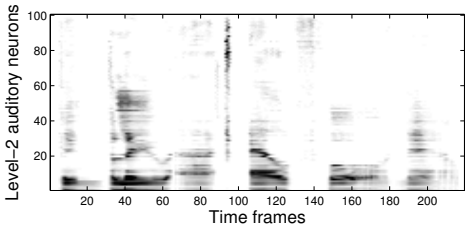


Fig. 2. The responses of layer-2 neurons to a speech sentence. The neurons are sorted according to the frequency region they are most sensitive to.

its connection with several layer-1 neurons. All neurons from the second layer are sorted according their sensible frequency region.

The responses of layer-2 neurons are also approximated by the inner product of receptive fields and the layer-1 input

$$H_2' = {W_2}^T V_1' \tag{7}$$

H_2' is a dimension-reduced version of H_1', with the fine structures encoded, while it still preserves the spectrogram-related information. Fig. 2 shows the neural responses to a speech sentence. Since we use speech signals to train these neurons, they are more sensitive to speech patterns; other input such as noise cannot stimulate these neurons effectively. This property makes this model robust to noises.

2.4. System architecture

Using two layers of pre-trained auditory neurons, we can encode speech signals by their corresponding neural responses. The system structure is shown in Fig. 3. Speech waveforms are transformed into time-frequency domain and further processed into sub-band representation. After two layers of neural filtering, each short-time frame is encoded by a vector describing the responses of layer-2 neurons. A discrete cosine transformation (DCT) module calculates the cepstral coefficients of the vector, which we called auditory neural cepstral coefficients (ANCC). To improve speech recognition performances, the delta and acceleration coefficients of ANCC can also be included. Then speech recognition can be performed on ANCC feature.

3. EXPERIMENTAL RESULTS

3.1. Calculating receptive fields and encoding speech

We use a small subset of TIMIT corpus to train a speaker-independent auditory neural model. 24 speakers, 2 males and 1 female from each dialect region, are selected; each speaker provides one utterance, and these 24 utterances are used to train the two layers of auditory neurons. These sentences cover most sub-band phenomenons, so they can produce neurons sensitive to most speech signals.

The utterances are sampled to 16kHz, and pre-emphasized with coefficient 0.97. The waveforms are transformed into time-frequency representations by 1024-point FFT, using $25ms$ Hamming window, with $\frac{5}{8}ms$ window shifts. The normalized spectrogram is divided equally into 32 sub-bands; and within each band, frames of $20ms$ are chopped with $10ms$ frame shifts. So each frame is a 16×16 matrix. For neurons of each band, frames from all 24 utterances are collected to calculate the receptive fields. We use the NMF algorithm with sparseness constraints 0.6 on $H_1(i)$ columns. The number of neurons in each band is 25. So the first layer contains 800 neurons. The second layer neural receptive fields are also trained using NMF

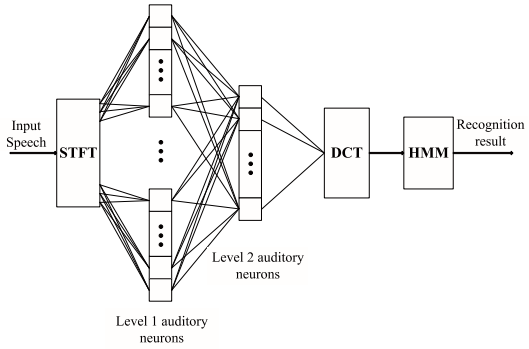


Fig. 3. The system structure.

with sparseness constraint 0.6 on H, and the total neuron number is 100. As noted in section 2, layer-2 neurons are sorted according to their responding frequency ranges.

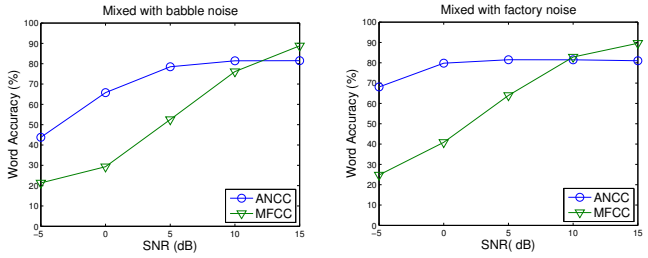
The 100-dimensional layer-2 neural responses to speech input are transformed into 50-dimensional cepstral coefficients by DCT. Together with corresponding delta and acceleration coefficients, each frame of speech is characterized by a 150-dimensional ANCC vector.

3.2. Using ANCC for speech recognition

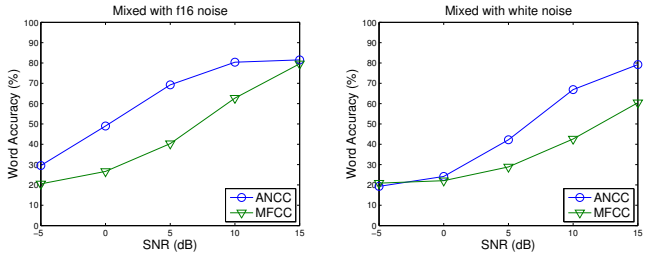
The performance of ANCC in speech recognition is tested on the Grid corpus [12]. The vocabulary of this corpus includes 4 verbs (bin, lay, place, set), 4 colors (blue, green, red, white), 4 preps (at, by, in, with), 25 letters with 'w' excluded, 10 numbers (0 to 9), and 4 codas (again, now, please, soon). Each sentence contains 6 words from these classes respectively. Each speaker recorded 500 such sentences. This corpus is more difficult than digit or letter based corpora, since it contains a more complex phone set.

We select 6 speakers (4 males, 2 females) as the experimental subjects. From the 500 sentences of each speaker, 400 are randomly chosen to train a speaker-dependent recognizer using ANCC feature calculated by the auditory neural system we perviously constructed; a baseline recognizer for each speaker is also trained using 39-dimensional Mel-scale cepstral coefficients (MFCC) as the features. These recognizers are monophone-based HMM systems, where each phone is a 3-state HMM with the probability density functions described by 3-gaussion mixtures. The other 100 sentences are mixed with babble, factory, f16, and white noises from NOISEX corpus in SNR intensities of 15dB, 10dB, 5dB, 0dB, and -5dB. In all testing cases, the performances of 6 recognizers are averaged. The word accuracies of clean speech recognition are ANCC: 83.11%, and MFCC: 93.50%. The performance comparisons of two features in different noise conditions are shown in Fig. 4.

The result shows that although ANCC performs a little poorer than MFCC in clean speech and some low noise conditions, its performance degradation with noise intensity increase is much slower. So it performs significantly better than MFCC in high noise conditions. Such as in 0dB condition of babble and factory noises, ANCC yields results close to clean speeech result, while MFCC has dropped sharply. Even in some negative SNR conditions, ANCC achieves preferable accuracies. The result also proves that this model is not sensitive to noise type.



(a) Babble noise conditions (b) Factory noise conditions



(c) F16 noise conditions (d) White noise conditions

Fig. 4. Speech recognition result comparisons between ANCC and MFCC in various noise conditions.

4. DISCUSSION

This model works in a parallel fashion, where neurons in the same layer do not interfere with each other. So it can be implemented on parallel hardware to improve its computational efficiency.

The auditory model in this paper is speaker-independent. It is obvious that for speaker-dependent applications, utterances from only one speaker are used to train the neurons. However, our experiments do not show that speaker-dependent neurons provide a better performance.

This model is motivated by auditory neural mechanism, but it does not incorporate a cochlear-like peripheral auditory stage. In fact, such pre-processing does not match well with our model. Cochlear-like filtering brings log-scale bandwidth, which compress the information in high frequency region. But consonants such as stops and fricatives distribute most of their energy there. So these consonants have to be represented by more information from low frequency region. MFCC is not affected, since it only encodes spectral energy; but in our model, spectral structures are also encoded. It is difficult for NMF to generate neurons preferably respond to consonant when much more training data are vowels with strong energy. Eventually the resulting neurons fit neither vowels nor consonants perfectly.

This model does not incorporate inhibitory coupling of neurons either, since NMF only generates non-negative outputs. This can cause ambiguity to the sensible patterns of neurons, which may reduce the sparseness of the corresponding neural activity. But the training method we use explicitly controls the resulting sparseness degree, and can alleviate this problem to a certain extent.

The performance of ANCC in clean speech is not satisfactory. We think the main reason is the neural response feature does not have the smooth changing property as MFCC. This may be solved by modeling neurons with non-linear behavior, which smoothes the neural response while still preserve the noise robustness.

5. CONCLUSION

This paper addresses the problem of robust speech recognition by introducing a computational auditory neural model to extracts noise resistant features. This model includes two layers of auditory neurons as feature detectors. A sparse coding method, NMF, is used to construct neurons that are sensitive to speech patterns. The responses of these neurons to speech can be processed into features called ANCC, which shows strong robustness to noise in speech recognition. We test the performance of ANCC using clean-speech trained recognizers to recognize speech in different noise conditions without additional treatments. The result shows that ANCC yields better results than MFCC in most cases, especially in high noise conditions. This model is also insensitive to noise stationarity.

6. ACKNOWLEDGEMENT

We would like to thank Lijuan Wang and Yuandong Tian for helpful discussions. The work was partially supported by the national natural science foundation of China under Grant number 60375015.

7. REFERENCES

[1] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. ASSP*, vol. 27, no. 2, pp. 113–120, 1979.

[2] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *JASA*, vol. 87, pp. 1738, 1990.

[3] O. Ghitza, "Auditory Nerve Representation as a Front-End for Speech Recognition in a Noisy Environment," *Computer Speech and Language*, vol. 1, no. 2, pp. 109–130, 1986.

[4] M.J.F. Gales and S.J. Young, "Robust continuous speech recognition using parallel modelcombination," *IEEE Trans. ASSP*, vol. 4, no. 5, pp. 352–359, 1996.

[5] B. Raj, E.B. Gouvea, P.J. Moreno, and R.M. Stern, "Cepstral compensation by polynomial approximation for environment-independent speech recognition," *Proc. International Conference on Spoken Language Processing*, vol. 4, 1996.

[6] R.C. deCharms, D.T. Blake, and M.M. Merzenich, "Optimizing sound features for cortical neurons.," *Science*, vol. 280, no. 5368, pp. 1439–43, 1998.

[7] B.A. Olshausen and D.J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, 2002.

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

[9] D.D. Lee and H.S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[10] S. Behnke, "Discovering hierarchical speech features using convolutional non-negative matrix factorization," *Proc. International Joint Conference on Neural Networks*, vol. 4, 2003.

[11] P.O. Hoyer, "Non-negative Matrix Factorization with Sparseness Constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[12] Barker J. Cunningham S. P. Cooke, M. P. and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *submitted to JASA, http://www.dcs.shef.ac.uk/spandh/gridcorpus/.*