INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Improved Example-based Speech Enhancement by Using Deep Neural Network Acoustic Model for Noise Robust Example Search

Atsunori Ogawa, Keisuke Kinoshita, Marc Delcroix, and Tomohiro Nakatani

NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan
{ogawa.atsunori,kinoshita.k,marc.delcroix,nakatani.tomohiro}@lab.ntt.co.jp

Abstract

Example-based speech enhancement is a promising single-channel approach for coping with highly nonstationary noise. Given a noisy speech input, it first searches in a noisy speech corpus for the noisy speech examples that best match the input. Then, it concatenates the clean speech examples that are paired with the matched noisy examples to obtain an estimate of the underlying clean speech component in the input. The quality of the enhanced speech depends on how accurate an example search can be performed given a noisy speech input. The example search is conventionally performed using a Gaussian mixture model (GMM) with mel-frequency cepstral coefficient features (MFCCs). To improve the noise robustness of the GMM-based example search, instead of using noise sensitive MFCCs, we have proposed using bottleneck features (BNFs), which are extracted from a deep neural network-based acoustic model (DNN-AM) built for automatic speech recognition. In this paper, instead of using a GMM with noise robust BNFs, we propose the direct use of a DNN-AM in the example search to further improve its noise robustness. Experimental results on the Aurora4 corpus show that the DNN-AM-based example search steadily improves the enhanced speech quality compared with the GMM-based example search using BNFs.
Index Terms: example-based speech enhancement, example search, noise robustness, deep neural network acoustic model

1. Introduction

Speech enhancement is an essential technology for improving the quality of speech-based applications in adverse environments. A lot of effort has been spent over the years on developing various types of effective speech enhancement approaches [1]. In particular, single-channel approaches have been extensively studied, e.g. [2–25], since they impose very few hardware constraints compared with multi-channel approaches.

Generative model-based single-channel approaches have been actively investigated. These approaches are explicitly based on the generation model of noisy speech. Parameters of the model are estimated, for example, by using heuristics such as minimum statistics [2–5], or by statistical model-based approaches based on, e.g. Gaussian mixture models (GMMs) [6, 7] and nonnegative matrix factorization [8, 9].

In contrast to the generative model-based approaches, more recently, deep neural network (DNN)-based discriminative single-channel approaches, i.e. denoising autoencoders (DAEs) [10–15], have started to attract a lot of attention. Unlike the generative model-based approaches, a DAE does not explicitly consider the generation process of noisy speech and, given a noisy speech input, it directly estimates the underlying clean speech component of the input. The DAE achieves this direct estimation (mapping) by nonlinearly transforming input noisy speech features into the corresponding clean speech features through its stacked hidden layers. The parameters of the hidden layers are estimated using a large-scale noisy-clean parallel speech corpus. The DAE shows a high denoising performance in various noisy environments.

In this paper, we focus on an example (or corpus)-based (or inventory-style) approach [16–25]. As with a DAE, an example-based approach directly estimates the underlying clean speech component in a given noisy input using a large-scale noisy-clean parallel speech corpus. However, it focuses strongly on exploiting raw and precise data, i.e. examples, included in the speech corpus. The example-based approach originally proposed in [16, 17] can be outlined as follows (see Section 2 for details). It prepares a clean speech corpus and a corresponding artificially contaminated noisy speech corpus. Given a noisy speech input for testing, it first uses a noisy speech GMM to search the noisy speech corpus for noisy speech examples (segments) that best match the input. Then, it concatenates the corresponding clean speech examples included in the clean speech corpus to obtain an estimate of the underlying clean speech component in the input. Finally, it uses this clean speech estimate to denoise the input. The GMM-based example search described above is performed based on the longest matching criterion. This criterion is important since longer speech examples can be identified more accurately in noisy environments than shorter examples because of their more distinct and richer spectral-temporal pattern information. As a result, the example-based approach exhibits higher enhancement performance than the other approaches especially in highly nonstationary noisy environments.

However, in previous studies of the example-based approach [16–20], the GMM-based example search was not always performed accurately enough. Although it is desirable that the noisy speech corpus encompass all the noisy environments that we encounter at the testing stage, in reality, this is infeasible. Moreover, since the example search is performed by evaluating the similarity between an input and a noisy example both of which are represented by mel-frequency cepstral coefficient features (MFCCs), which are sensitive to noise, the search process can be greatly affected by noise. Therefore, a mismatch between an input and the noisy speech corpus is inevitable. This mismatch makes the GMM-based example search less accurate and therefore degrades the quality of the enhanced speech.

To mitigate this mismatch, instead of using MFCCs, in [20] we proposed using bottleneck features (BNFs) [26–28] as a representation of a noisy speech input and a noisy speech example. BNFs are extracted from a DNN acoustic model (DNN-AM) built for automatic speech recognition (ASR). A DNN-AM can perform a robust prediction of hidden Markov model (HMM)-states for input noisy speech features thanks to nonlinear feature transformations through its stacked hidden layers (as with a DAE). By using noise robust (invariant) BNFs extracted from a DNN-AM, the accuracy of the GMM-based example search was improved and, as a result, the quality of the enhanced speech was also improved compared with when using MFCCs in a GMM-based example search.

However, in the above example search using a GMM with BNFs, the noise robustness of a DNN-AM may not be fully exploited due to the intermediate Gaussian mixture modeling. In

http://dx.doi.org/10.21437/Interspeech.2017-543

this paper, we propose the direct use of a DNN-AM in the example search to further improve its noise robustness by fully exploiting the high noise robustness of the DNN-AM (Section 3). Experimental results on the Aurora4 corpus [29, 30] show that the DNN-AM-based example search steadily improves the enhanced speech quality compared with the GMM-based example search using BNFs (Section 4).

2. Example-based speech enhancement

We begin by using Fig. 1 to outline the basic framework of the example-based approach, which was originally proposed in [16, 17]. Then, we elaborate the main process of the approach, i.e. the example search [19], and its problems.

2.1. Basic framework

In the training stage (top dotted box in Fig. 1), a clean speech corpus is first prepared. It is artificially contaminated with various types of noise to form a multi-condition parallel speech corpus. MFCC extraction is performed for all of the speech corpora (as for the clean corpus, magnitude spectra are also extracted). The extracted MFCCs are then used to train GMMs that represent each of the corpora. To represent the precise spectral patterns of a speech, the dimensionality of the MFCCs and the number of Gaussian components in the GMMs are set at large values (e.g. 80 and 4096). These GMMs are used to obtain example models for each of the corpora.

Then, given a noisy speech input for testing, we first extract its MFCC, magnitude spectrum and phase spectrum sequences. Using the example models and an example evaluation function [19], we find the longest example sequence that matches the input with the posterior probabilities of the matched examples.

We use the found matching example sequence and the example posterior probabilities to resynthesize a clean magnitude spectrum sequence by concatenating the corresponding clean speech magnitude spectra. Finally, we perform Wiener filtering using the resynthesized clean magnitude spectrum sequence and the magnitude and phase spectrum sequences extracted from the noisy speech input to obtain the final enhanced speech.

2.2. Example search

Hereafter, for simplicity, we assume that a single speech corpus is used for the example search (we actually use a single noisy speech corpus for the example search in the experiments described in Section 4) and all the utterances in the corpus are concatenated into one long utterance. We employ the notations used in [16, 17].

We start with the training stage. Let $\mathbf{x} = \{x_i : i = 1, 2, \ldots, I\}$ be the whole MFCC sequence in the speech corpus, x_i be the MFCCs at time frame i, and I be the total number of frames in the corpus. Using $\mathbf{x}$, a GMM G is trained as

$$G = \{g(x|m), w(m) : m = 1, 2, \ldots, M\}, \quad (1)$$

where $g(x|m)$ is the mth Gaussian component, $w(m)$ is its weight ($0 < w(m) < 1$), and M is the total number of Gaussian components in G. For each time frame i in the corpus $\mathbf{x}$, we find the Gaussian component $g(x|m)$ in G that maximizes the likelihood of the MFCCs x_i. This results in the following time sequence of maximum-likelihood Gaussian component indices

$$\mathbf{m} = \{m_i : i = 1, 2, \ldots, I\}, \quad (2)$$

where m_i is the index that addresses $g(x|m_i)$, i.e. the Gaussian component at the ith time frame. $g(x|m_i)$ represents the class of the precise short-time speech spectra, and thus, $\mathbf{m}$ can be used as a model that represents precise spectral-temporal patterns included in $\mathbf{x}$. We refer to this model $\mathbf{m}$ as an example model of the corpus $\mathbf{x}$.

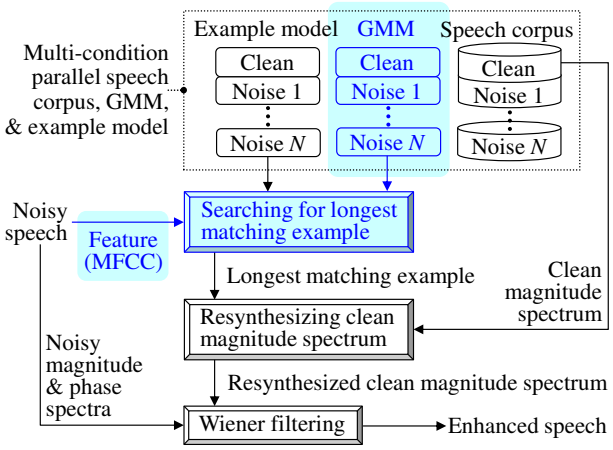

Figure 1: *Basic framework of example-based speech enhancement.*

In the following, we describe the example search, or more exactly an example evaluation function. Note that the example evaluation function described below is the one we proposed in [19] as a modified version of the one used in [16, 17].

Let $\mathbf{y} = \{y_t : t = 1, 2, \ldots, T\}$ be the T frame MFCC sequence of a noisy speech input for testing, y_t be the MFCCs at time frame t, and $\mathbf{y}_{t:t+\tau} = \{y_\epsilon : \epsilon = t, t+1, \ldots, t+\tau\}$ be a $\tau + 1$ frame MFCC segment taken from the time frames t to $t + \tau$ of $\mathbf{y}$. We calculate the likelihood of $\mathbf{y}_{t:t+\tau}$ by matching it with the example model $\mathbf{m}$ and the GMM G. Hereafter, again for simplicity, we add a constraint that only allows one-to-one frame basis linear matching (i.e. it does not allow the dynamic time warping) during the matching between $\mathbf{y}_{t:t+\tau}$ and $\mathbf{m}$ or G. We divide $\mathbf{y}_{t:t+\tau}$ into two sub-segments. The likelihood of the first sub-segment $\mathbf{y}_{t:t+\nu}$ ($0 \le \nu \le \tau$) is calculated with $\mathbf{m}_{u:u+\nu}$, i.e. the $\nu + 1$ frame example taken from the time frames u to $u + \nu$ of $\mathbf{m}$, and the likelihood of the second sub-segment $\mathbf{y}_{t+\nu+1:t+\tau}$ is calculated with G (denoted as $\phi_{u+\nu+1:u+\tau}$). As a result, the likelihood of $\mathbf{y}_{t:t+\tau}$, i.e. the evaluation function of the example $\mathbf{m}_{u:u+\nu}$, is written as

$$\begin{aligned} &p(\mathbf{y}_{t:t+\tau}|\mathbf{m}_{u:u+\nu}, \phi_{u+\nu+1:u+\tau}) \\ &= p(\mathbf{y}_{t:t+\nu}|\mathbf{m}_{u:u+\nu}) p(\mathbf{y}_{t+\nu+1:t+\tau}|\phi_{u+\nu+1:u+\tau}), \\ &= \prod_{\epsilon=0}^{\nu} g(y_{t+\epsilon}|m_{u+\epsilon}) \prod_{\epsilon=\nu+1}^{\tau} \left[\sum_{m=1}^{M} w(m) g(y_{t+\epsilon}|m) \right]. \end{aligned} \quad (3)$$

We can change the length of $\mathbf{m}_{u:u+\nu}$ by changing the value of ν. However, we always evaluate $\mathbf{m}_{u:u+\nu}$ with the length of $\mathbf{y}_{t:t+\tau}$, i.e. $\tau + 1$, by assuming there is no particular Gaussian component sequence that matches $\mathbf{y}_{t+\nu+1:t+\tau}$ and by assigning the accumulated GMM likelihood to $\mathbf{y}_{t+\nu+1:t+\tau}$ as the smoothed likelihood. This example evaluation method is similar to the hypothesis evaluation method employed in the A* search for ASR [31, 32].

Using this example evaluation function, at each time frame t in $\mathbf{y}$, we can find the MFCC segment $\mathbf{y}_{t:t+\nu_{\max}}$ and the corresponding matching example $\mathbf{m}^t_{\hat{u}:\hat{u}+\nu_{\max}}$ by maximizing the posterior probability as

$$\begin{aligned} &\mathbf{m}^t_{\hat{u}:\hat{u}+\nu_{\max}} \\ &= \arg\max_{\nu} \max_{\mathbf{m}_{u:u+\nu}} P(\mathbf{m}_{u:u+\nu}, \phi_{u+\nu+1:u+\tau}|\mathbf{y}_{t:t+\tau}), \end{aligned} \quad (4)$$

$$\begin{aligned} &P(\mathbf{m}_{u:u+\nu}, \phi_{u+\nu+1:u+\tau}|\mathbf{y}_{t:t+\tau}) \\ &= \frac{p(\mathbf{y}_{t:t+\tau}|\mathbf{m}_{u:u+\nu}, \phi_{u+\nu+1:u+\tau})}{\sum_{u'} \sum_{\nu'} p(\mathbf{y}_{t:t+\tau}|\mathbf{m}_{u':u'+\nu'}, \phi_{u'+\nu'+1:u'+\tau})}, \end{aligned} \quad (5)$$

where the denominator of Eq. (5) is the sum of the likelihoods of the MFCC segment $\mathbf{y}_{t:t+\tau}$ given all the possible example locations u' and all the possible example lengths $\nu'+1$ (i.e. all the possible MFCC segment division boundaries ν'). An efficient implementation method of Eq. (4) is detailed in [19].

The posterior probability of Eq. (5) has the longest matching property. Its proof is simple as follows. We compare the two posterior probabilities of the MFCC segment $\mathbf{y}_{t:t+\tau}$. One is evaluated with the example $\mathbf{m}_{u:u+\nu}$ and GMM $\phi_{u+\nu+1:u+\tau}$, and the other is evaluated with $\mathbf{m}_{u:u+\nu-1}$ and $\phi_{u+\nu:u+\tau}$. The denominator is common to both probabilities and their ratio is equal to the likelihood ratio as

$$\frac{p(\mathbf{y}_{t:t+\tau}|\mathbf{m}_{u:u+\nu},\phi_{u+\nu+1:u+\tau})}{p(\mathbf{y}_{t:t+\tau}|\mathbf{m}_{u:u+\nu-1},\phi_{u+\nu:u+\tau})}=\frac{g(y_{t+\nu}|m_{u+\nu})}{\sum_{m=1}^{M}w(m)g(y_{t+\nu}|m)}. \tag{6}$$

Here, we assume that the whole acoustic space is equally covered by each of the Gaussian components in G and the MFCC $y_{t+\nu}$ is well-matched to the Gaussian component $m_{u+\nu}$. With these two assumptions, the denominator on the right-hand side of Eq. (6) can be approximated as $w(m_{u+\nu})g(y_{t+\nu}|m_{u+\nu})$, and thus, Eq. (6) becomes equal to $1/w(m_{u+\nu}) \geq 1$. This means that, as long as there is a Gaussian component $m_{u+\nu}$ that matches the MFCC $y_{t+\nu}$, the matching example $\mathbf{m}_{u:u+\nu}$ becomes long. The importance of this longest matching property was described in Section 1.

2.3. Problems of example search

As we have already described in Section 1, it is impossible to prepare a noisy speech corpus that covers all types of noisy environments. Moreover, the example search is performed using noise sensitive MFCCs. Therefore, a mismatch between a noisy speech input and the noisy speech corpus is inevitable. This mismatch can degrade the accuracy of the example search and, as a result, the quality of the enhanced speech.

To mitigate this mismatch, instead of using MFCCs, in [20] we proposed using BNFs [26–28], which are extracted from a DNN-AM built for ASR, in the GMM-based example search. However, with this method, the noise robustness of a DNN-AM may not be fully exploited due to the intermediate Gaussian mixture modeling.

3. Proposed method: Example search using DNN-AM

We propose the direct use a DNN-AM, which is built for ASR, in the example search to further improve its noise robustness by fully exploiting the high noise robustness of the DNN-AM. There are many types of DNN-AMs for ASR [33–35]. However, they all usually have full connections between the last hidden layer and the output layer to predict the posterior probabilities of the HMM states for the input speech features by using the softmax activation function. One node in the output layer corresponds to one HMM state. Input features are usually MFCCs or log-mel filterbank coefficients (FBANKs), which are spliced within a left and right context window across several frames.

Using a DNN-AM, we can calculate the likelihood for one frame of input speech features x given an HMM state s as

$$p(x|s)=\exp(z_s^L(x)-\log P(s)), \tag{7}$$

where $z_s^L(x)$ is the excitation value at the sth node of the output layer (the Lth layer in the DNN-AM) given x, and $P(s)$ is the prior probability of the HMM state s. $z_s^L(x)$ is obtained as

$$z_s^L(x)=\sum_{r=1}^{R}w_{s,r}^L f(z_r^{L-1}(x))+b_s^L, \tag{8}$$

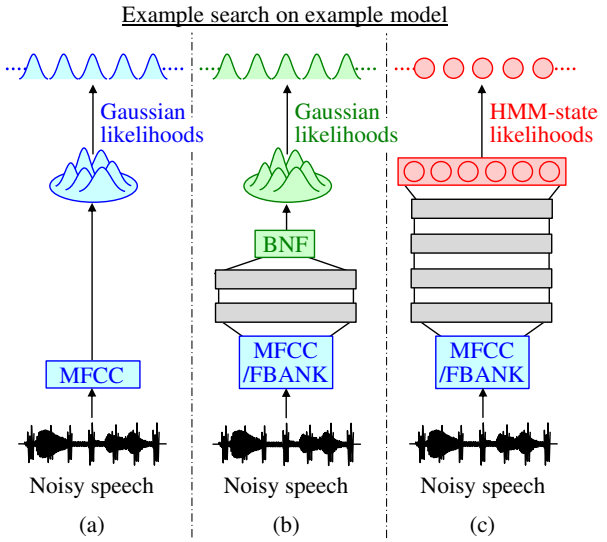

Figure 2: *Three types of example search. (a) GMM-based example search using MFCCs, (b) GMM-based example search using BNFs, and (c) DNN-AM-based example search.*

where $w_{s,r}^L$ is the weight value between the rth node of the last hidden layer (the $(L-1)$th layer in the DNN-AM) and the sth node of the output layer, b_s^L is the bias value of the sth node of the output layer, $z_r^{L-1}(x)$ is the excitation value at the rth node of the last hidden layer, $f(\cdot)$ is an activation function, typically the sigmoid function, and R is the number of nodes in the last hidden layer. We also define a pseudo GMM G_s using $p(x|s)$ and $P(s)$ as follows

$$G_s=\{p(x|s),P(s):s=1,2,\ldots,S\}, \tag{9}$$

where S is the number of HMM states (nodes of the output layer). As shown in Eq. (2), using a GMM G, we can obtain an example model $\mathbf{m}$ for the whole feature sequence $\mathbf{x}$ of a speech corpus. In the same way, using the pseudo GMM G_s, we can obtain an example model, i.e. a time sequence of maximum-likelihood HMM state (output layer node) indices, for $\mathbf{x}$ as

$$\mathbf{s}=\{s_i:i=1,2,\ldots,I\}, \tag{10}$$

where s_i is the index that addresses the HMM state (the node of the output layer) at the ith time frame.

Using $p(x|s)$, $P(s)$, G_s and $\mathbf{s}$ instead of $g(x|m)$, $w(m)$, G and $\mathbf{m}$, and following Eqs. (3) to (5), we can perform the DNN-AM-based example search in the same way as the GMM-based example search while satisfying the longest matching property as proved in Eq. (6). The likelihood $p(x|s)$ is also used in the decoding of ASR and thus the DNN-AM-based example search can be understood as a kind of ASR decoding constrained by the example model (without using a language model). Figure 2 shows the three types of example search, i.e. (a) the GMM-based example search using MFCCs [16, 17, 19], (b) the GMM-based example search using BNFs [20], and (c) the DNN-AM-based example search. This figure clearly illustrates that the proposed DNN-AM-based example search directly exploits the advantage of a DNN-AM, namely its high noise robustness. However, it also has a high speaker normalization ability [26], and thus it may be difficult to use speaker information included in input speech in the example search. As a result, the enhanced speech may lose the original speaker characteristics (actually, we do not need to be concerned with this as described in Section 4.2). It also should be noted that any types of NN-based AMs can be used in the proposed DNN-AM-based example search.

4. Experiments

We conducted experiments to evaluate the example-based approach based on the proposed DNN-AM-based example search (hereafter, referred to as ExB-DNN-AM) in comparison with that based on the GMM-based example search using MFCCs (ExB-GMM-MFCC), that based on the GMM-based example search using BNFs (ExB-GMM-BNF), and a DAE.

4.1. Experimental settings

The Aurora4 multi-condition speech corpus [29, 30] was used in the experiments. The corpus is derived from the Wall Street Journal (WSJ0) 5k-word closed vocabulary ASR task. The sampling frequency, frame length, and frame shift were 16 kHz, 20 ms, and 10 ms, respectively. The clean training set consists of 7138 utterances spoken by 83 speakers (about 14 hours in total). The noisy training set is a noisy version of the clean training set including a clean and six different types of noise conditions (car, babble, restaurant, street traffic, airport, train station) with 10 to 20 dB signal-to-noise ratios (SNRs). These two training sets form a noisy-clean parallel speech corpus. In the example-based approaches, the noisy set was used for the example search and the clean set was used to provide clean speech examples.

We used a subset of the original evaluation set. The subset consisted of 112 utterances obtained from eight speakers and all the seven noisy (including a clean) conditions included in the original evaluation set to keep the variety of the original set. The eight speakers were different from those of the training set. The seven noisy conditions were the same as those of the training set but with different SNRs, i.e. 5 to 15 dB SNRs.

A DAE, i.e. a strong DNN-based competitor, was trained using the noisy-clean parallel speech corpus. The DAE was based on a fully connected feedforward DNN that had four 2048-node hidden layers with the sigmoid activation function. Both the input and output features were 40-dimensional log-power spectra. This DAE is detailed in [15].

For ExB-GMM-MFCC (ExB-GMM-BNF), we extracted 80-dimensional MFCCs (BNFs) for the noisy training set and trained a GMM with 4096 Gaussian components. The BNFs were extracted using a fully connected feedforward DNN-AM trained using the noisy training set. The DNN-AM had seven hidden layers with the sigmoid activation function. The sixth layer was an 80-node bottleneck layer and the other layers had 2048 nodes. ExB-GMM-BNF is detailed in [20].

We trained a convolutional neural network-based acoustic model (CNN-AM) using the noisy training set and used it for ExB-DNN-AM. This is because a CNN-AM shows high ASR performance for the Aurora4 task [33]. The input features consisted of 1320-dimensional vector obtained by splicing 40-dimensional FBANKs plus their Δ and $\Delta\Delta$ coefficients within an 11-frame context window. These features were input into the CNN-AM, which consisted of two convolution plus pooling layers followed by three fully connected 2048-node layers with the sigmoid activation function (the operations in the CNN-AM were basically the same as those described in [33]). Then, posterior probabilities of the HMM states were predicted at the output layer. The number of HMM states (nodes of the output layer) was set at 4096, which was equal to the number of Gaussian components in the GMMs of ExB-GMM-MFCC and ExB-GMM-BNF.

4.2. Experimental results

Figure 3 shows the objective evaluation results measured with the frequency weighted segmental SNR (FWSegSNR) [36] and the short-time objective intelligibility measure (STOI) [37, 38]. We can confirm that ExB-DNN-AM steadily improves the values of both the evaluation measures compared with the other

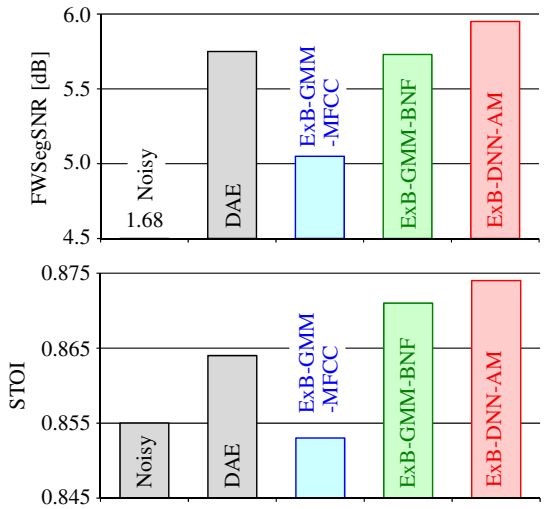

Figure 3: *Objective evaluation results averaged over seven noisy (including a clean) conditions of Aurora4 corpus.*

methods. These are the effects of directly using (i.e. fully exploiting the noise robustness of) the DNN-AM (CNN-AM) in the example search as expected in Sections 1 and 3.

Results of an informal listening test indicate that the audible quality of the enhanced speech obtained by ExB-DNN-AM is also steadily improved compared with those obtained by the DAE and ExB-GMM-MFCC, and it is slightly better than that obtained by ExB-GMM-BNF. We had a concern that the original speaker characteristics may be lost in the enhanced speech obtained by ExB-DNN-AM due to the high speaker normalization ability of a DNN-AM (Section 3). However, it maintains the characteristics well thanks to Wiener filtering using the original input speech at the last stage of the enhancement (Section 2.1). We expect that the quality of the enhanced speech may be further improved if we develop a DNN-AM that can jointly perform HMM state prediction and speaker identification using multitask learning [39].

5. Relation to previous work

The steps in our development of the example-based approach, i.e. from the original ExB-GMM-MFCC [16, 17, 19] to ExB-GMM-BNF [20] and then ExB-DNN-AM proposed in this paper, are similar to those of the ASR systems, i.e. from conventional GMM-HMM systems to Tandem systems [26–28] and then state-of-the-art DNN-HMM hybrid systems [33–35].

We have attempted in this paper to improve speech enhancement performance by exploiting ASR technology. Similar attempts have been made in some other studies [14, 24, 25] and we can refer to them to further improve our method.

6. Conclusion and future work

We have proposed an improved example-based speech enhancement approach that directly uses a DNN-AM for a noise robust example search. Our experimental results confirm the effectiveness of the proposed DNN-AM-based example search.

Future work will include (1) the use of DNN-AMs other than the CNN-AMs, e.g. fully connected feedforward DNN-AMs and (bidirectional) (long short-term memory) recurrent NN-based AMs [33–35], (2) the development of a DNN-AM that is more suitable for the proposed example search, i.e. that jointly performs HMM state prediction and speaker identification (i.e. a multitask learning [39]), and (3) more tight integration with the ASR technology [14, 24, 25] for complementarily improving the performance of speech enhancement and ASR.

7. References

[1] P. Loizou, *Speech Enhancement: Theory and Practice, Second Edition*. CRC Press, 2013.

[2] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, December 1984.

[3] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, July 2001.

[4] I. Cohen, "Noise estimation by minima controlled recursive averaging for robust speech enhancement," *IEEE Signal Processing Letters*, vol. 9, no. 1, pp. 12–15, January 2002.

[5] ——, "Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 5, pp. 466–475, September 2003.

[6] P. Moreno, B. Raj, and R. Stern, "A vector Taylor series approach for environment-independent speech recognition," in *Proc. ICASSP*. IEEE, 1996, pp. 733–736.

[7] M. Fujimoto, S. Watanabe, and T. Nakatani, "Noise suppression with unsupervised joint speaker adaptation and noise mixture model estimation," in *Proc. ICASSP*. IEEE, 2012, pp. 4713–4716.

[8] K. Wilson, B. Raj, P. Smaragdis, and A. Divakaran, "Speech denoising using nonnegative matrix factorization with priors," in *Proc. ICASSP*. IEEE, 2008, pp. 4029–4032.

[9] N. Mohammadiha, P. Smaragdis, and A. Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2140–2151, October 2013.

[10] A. Maas, Q. Le, T. O'Neil, O. Vinyals, P. Nguyen, and A. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Proc. Interspeech*. ISCA, 2012.

[11] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, January 2014.

[12] K. Han, Y. Wang, D. Wang, W. Woods, I. Merks, and T. Zhang, "Learning spectral mapping for speech dereverberation and denoising," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 6, pp. 982–992, June 2015.

[13] F. Weninger, S. Watanabe, Y. Tachioka, and B. Schuller, "Deep recurrent de-noising auto-encoder and blind de-reverberation for reverberated speech recognition," in *Proc. ICASSP*. IEEE, 2014, pp. 4623–4627.

[14] H. Erdogan, J. Hershey, S. Watanabe, and J. L. Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. ICASSP*. IEEE, 2015, pp. 708–712.

[15] K. Kinoshita, M. Delcroix, A. Ogawa, and T. Nakatani, "Text-informed speech enhancement with deep neural networks," in *Proc. Interspeech*. ISCA, 2015, pp. 1760–1764.

[16] J. Ming, R. Srinivasan, and D. Crookes, "A corpus-based approach to speech enhancement from nonstationary noise," in *Proc. Interspeech*. ISCA, 2010, pp. 1097–1100.

[17] ——, "A corpus-based approach to speech enhancement from nonstationary noise," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 822–836, May 2011.

[18] K. Kinoshita, M. Souden, M. Delcroix, and T. Nakatani, "Single channel dereverberation using example-based speech enhencement with uncertainty decoding technique," in *Proc. Interspeech*. ISCA, 2011, pp. 197–200.

[19] A. Ogawa, K. Kinoshita, T. Hori, T. Nakatani, and A. Nakamura, "Fast segment search for corpus-based speech enhancement based on speech recognition technology," in *Proc. ICASSP*. IEEE, 2014, pp. 1576–1580.

[20] A. Ogawa, S. Seki, K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, and K. Takeda, "Robust example search using bottleneck features for example-based speech enhancement," in *Proc. Interspeech*. ISCA, 2016, pp. 3733–3737.

[21] X. Xiao and R. Nickel, "Speech enhancement with inventory style speech resynthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1243–1257, August 2010.

[22] R. Nickel, R. Astudillo, D. Kolossa, and R. Martin, "Corpus-based speech enhancement with uncertainty modeling and cepstral smoothing," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 983–997, May 2013.

[23] R. Nickel and R. Martin, "Memory and complexity reduction for inventory-style speech enhancement systems," in *Proc. EUSIPCO*. EURASIP, 2011, pp. 196–200.

[24] J. Ming and D. Crookes, "Wide matching - an approach to improving noise robustness for speech enhancement," in *Proc. ICASSP*. IEEE, 2016, pp. 5910–5914.

[25] ——, "Speech enhancement based on full-sentence correlation and clean speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 3, pp. 531–543, March 2017.

[26] D. Yu and L. Deng, *Automatic speech recognition: a deep learning approach*. Springer-Verlag London, 2015.

[27] F. Grézl and P. Fousek, "Optimizing bottle-neck features for LVCSR," in *Proc. ICASSP*. IEEE, 2008, pp. 4279–4732.

[28] T. Yoshioka and M. Gales, "Environmentally robust ASR front-end for deep neural network acoustic models," *Computer Speech and Language*, vol. 31, no. 1, pp. 65–86, May 2015.

[29] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*. IEEE, 2013, pp. 7398–7402.

[30] G. Hirsch, "Experimental framework for the performance evaluation of speech recognition front-ends on a large vocabulary task, version 2.0, AU/417/02." ETSI STQ-Aurora DSR Working Group, October 2002.

[31] D. Paul, "New developments in the Lincoln stack-decoder based large-vocabulary CSR system," in *Proc. ICASSP*. IEEE, 1995, pp. 45–48.

[32] P. Gopalakrishnan, L. Bahl, and R. Mercer, "A tree search strategy for large-vocabulary continuous speech recognition," in *Proc. ICASSP*. IEEE, 1995, pp. 572–575.

[33] T. Yoshioka, K. Ohnishi, F. Fang, and T. Nakatani, "Noise robust speech recognition using recent developments in neural networks for computer vision," in *Proc. ICASSP*. IEEE, 2016, pp. 5730–5733.

[34] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. ICASSP*. IEEE, 2013, pp. 6645–6649.

[35] T. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. ICASSP*. IEEE, 2015, pp. 4580–4584.

[36] Y. Hu and P. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, January 2008.

[37] C. Taal, R. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time-frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, September 2011.

[38] [Online]. Available: http://www.ceestaal.nl/matlab-code/.

[39] R. Caruana, "Multitask learning: a knowledge-based source of inductive bias," in *Proc. ICML*. IMLS, 1993, pp. 41–48.