WORD GRAPH BASED FEATURE ENHANCEMENT FOR NOISY SPEECH RECOGNITION

Zhi-Jie Yan[1] *Frank K. Soong*[2] *Ren-Hua Wang*[1]

[1]iFlytek Speech Lab, University of Science and Technology of China, Hefei, P. R. China, 230027
[2]Microsoft Research Asia, Beijing, P. R. China, 100080
yanzhijie@ustc.edu frankkps@microsoft.com rhw@ustc.edu.cn

ABSTRACT

This paper presents a word graph based feature enhancement method for robust speech recognition in noise. The approach uses signal processing based speech enhancement as a starting point, and then performs Wiener filtering to remove residual noise. During the process, a decoded word graph is used to directly guide the feature enhancement with respect to the HMM for recognition, so that the enhanced feature can match the clean speech model better in the acoustic space. The proposed word graph based feature enhancement method was tested on the Aurora 2 database. Experimental results show that an improved recognition performance can be obtained comparing with conventional signal processing based and GMM based feature enhancement methods. With signal processing based Weighted Noise Estimation and GMM based method, the relative error rate reductions are 35.44% and 42.58%, respectively. The proposed word graph based method improves the performance further, and a relative error rate reduction of 57.89% is obtained.

Index Terms— Speech recognition, Robustness, Speech enhancement

1. INTRODUCTION

It has been well-known that the performance of Automatic Speech Recognition (ASR) system degrades dramatically when there is a mismatch between training and testing environments. For the ASR systems deployed in real conditions, this mismatch is usually caused by additive noise and channel distortion. Consequently, robust speech recognition which has the ability to compensate various kinds of noise and channel effect is desired as one of the key techniques for real-world ASR applications.

Previous research which focused on minimizing environmental mismatch can be categorized into the following three groups: (a) *Signal processing based compensation*, which tries to find robust features (e.g., Mel-Frequency Cepstral Coefficients, MFCC) or to compensate noise and channel effect over the representation of speech (e.g., Spectral Subtraction [1], Cepstral Mean Normalization, CMN [2]); (b) *Model based compensation*, which tries to adapt model or to transform feature with respect to the model so that the speech variation in noisy environments can be better handled (e.g., Parallel Model Combination, PMC [3], and Feature-space Maximum Likelihood Linear Regression, fMLLR [4]); (c) *Combination of signal processing based and model based methods*, which tries to benefit from both approaches (e.g., model based compensation [5], and Model Based Wiener filter, MBW [6]).

Generally speaking, the methods in the first group are simple and efficient, while the methods in the second group can achieve a better recognition performance at the cost of much more computational load. Compared with the first two groups, the methods in the third group aim to take advantages of both signal processing based and model based approaches, and try to achieve a reasonable recognition performance while maintaining a relatively low computational cost.

In this paper, we present a word graph based feature enhancement method, which belongs to the third group of the compensation approaches. The proposed method is based upon Wiener filtering of the Mel-filter bank energy, given (a) the input noisy speech, (b) a signal processing based estimate of noise, and (c) a clean trained Hidden Markov Model (HMM) which is used for both feature enhancement and speech recognition. In our approach, the input noisy speech is first de-noised and channel-normalized via signal processing based method. The roughly processed signal is then decoded using the clean speech model, and a word graph is obtained to represent the hypothesis space. After that, both static and dynamic features of the model based clean speech are estimated, and the speech parameter sequence for Wiener filtering is synthesized in Maximum Likelihood (ML) sense. Finally, Wiener filtering is performed using the input noisy speech, the estimated noise, and the synthesized model based clean speech. The output of the filter is re-decoded in a word graph constrained second pass decoding, to get the final recognition results.

The main difference compared with previous research [5, 6] is that in our approach, a word graph is constructed to directly guide the feature enhancement process with respect to the clean speech model for recognition. As a result, a same HMM can be used for both enhancement and recognition, and the use of another Gaussian Mixture Model (GMM) in [5, 6] becomes unnecessary. The word graph based approach enables us to exploit the temporal resolution of the HMM, as well as to improve the estimate accuracy of the model based clean speech via imposing the explicit constraint between its static and dynamic features. Therefore, the enhanced speech feature after Wiener filtering can match the clean speech model better in the acoustic space, and thus leads to an improved recognition performance.

The rest of this paper is organized as follows: In Section 2, the word graph based feature enhancement method is first described from a global point of view, and then specified in the details of the processing steps. In Section 3, experimental results of the proposed method are presented and compared with conventional methods. Finally, we draw our conclusions and future work in Section 4.

2. WORD GRAPH BASED FEATURE ENHANCEMENT

2.1. System Overview

The flowchart of the word graph based feature enhancement method can be illustrated by Fig. 1. The input noisy speech, X, is first fed into the signal processing based speech enhancement block, in which Weighted Noise Estimation [7] is performed to get a rough es-

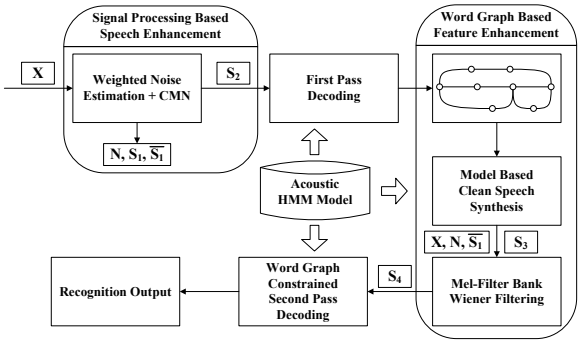

Fig. 1. Flowchart of the word graph based feature enhancement.

timate of the noise spectrum N, as well as the corresponding clean speech S_1. Then, S_1 is converted to MFCC coefficients, and the cepstral mean $\overline{S_1}$ is subtracted from it to normalize the channel effect. After that, the normalized speech, S_2, is decoded using the clean speech HMM, and a word graph representing the hypothesis space is constructed. By merging the kernel parameters of the clean speech model according to their posterior probabilities, a model based estimate of the clean speech, S_3, can be synthesized. S_3 is then transformed back to Mel-filter bank energy, and Wiener filtering is performed to get the final estimate of the clean speech, S_4. In the last step, S_4 is re-decoded in a constrained search space defined by the word graph, and the final recognition output is obtained.

In the following subsections, the processing steps of the method will be specified in details:

2.2. Signal Processing Based Speech Enhancement

The input noisy speech X is first fed into the speech enhancement block, where it is converted to the linear spectral domain, and signal processing based speech enhancement is performed. The purpose of this step is to get a rough estimate of the noise and clean speech with relatively low computational cost. Because the accuracy of the word graph decoding relies greatly on the Signal-to-Noise Ratio (SNR) of the input speech, it is then necessary to remove the noise effect via signal processing based enhancement before a clean trained speech model can be applied.

In our approach, Weighted Noise Estimation [7] is performed to estimate the noise spectrum. This method continuously updates the noise estimate N, using weighted noisy speech according to the estimated SNR. Consequently, the corresponding clean speech S_1 can be obtained by using conventional spectral subtraction.

Besides additive noise, channel effect should also be considered. In our approach, CMN is performed on S_1 to get the channel-normalized MFCC coefficients, S_2. Meanwhile, the cepstral mean $\overline{S_1}$ is also stored for latter process.

2.3. First Pass Decoding and Word Graph Construction

S_2 is decoded using the clean trained HMM to construct a word graph which compactly represents the hypothesis space. Even after signal processing based speech enhancement, S_2 may still have some residual noise which can lead to incorrect decoding. But the word graph based approach would have more chance that the correct hypotheses exist in the graph with relatively lower posterior probabilities (or likelihoods) than the incorrect first best hypothesis. Therefore, they can still be recovered in the latter Wiener filtering process with the help of the clean speech model.

Once the word graph has been decoded, kernel posterior probabilities for each Gaussian component of the model can be calculated. These posterior probabilities will serve as the weighting coefficients for synthesizing the model based clean speech for Wiener filtering. Using the word graph, the posterior probability of kernel k at time t, given the entire observation sequence $\boldsymbol{o}_1^T$ can be formulated as:

$$p([k;t] \mid \boldsymbol{o}_1^T) = \sum_{\substack{\forall [w;s,e] \\ s \le t \le e \\ j \in w, k \in j}} p([w;s,e] \mid \boldsymbol{o}_1^T) \cdot p([j;t] \mid w) \cdot p([k;t] \mid j) \tag{1}$$

in which $p([w;s,e] \mid \boldsymbol{o}_1^T)$ is the Word Posterior Probability (WPP) of word w in the word graph, starting at time s and ending at time e; $p([j;t] \mid w)$, the state occupancy probability of state j at time t, given w; $p([k;t] \mid j)$, the occupancy probability of kernel k in state j at time t.

In Eq. (1), $p([w;s,e] \mid \boldsymbol{o}_1^T)$ is calculated as the conventional WPP [8] defined as:

$$p([w;s,e] \mid \boldsymbol{o}_1^T) = \sum_{\substack{\forall M, [w;s,e]_1^M \\ \exists n, 1 \le n \le M \\ w = w_n, s = s_n, e = e_n}} \frac{\prod_{m=1}^{M} p(\boldsymbol{o}_{s_m}^{e_m} \mid w_m) \cdot p(w_m \mid w_1^M)}{p(\boldsymbol{o}_1^T)} \tag{2}$$

in which M is the number of words in a string hypothesis; $p(\boldsymbol{o}_{s_m}^{e_m} \mid w_m)$ and $p(w_m \mid w_1^M)$ are the scaled acoustic model likelihood and language model likelihood, respectively. Within the word graph, $p([w;s,e] \mid \boldsymbol{o}_1^T)$ can be calculated efficiently with the forward-backward algorithm.

In our approach, state occupancy probability $p([j;t] \mid w)$ is calculated using Viterbi approximation, so it equals one for the states of the best alignment path $J([w;s,e], \boldsymbol{o}_s^e)$ and zero otherwise:

$$p([j;t] \mid w) \overset{\text{Viterbi}}{\approx} \delta(j, J_t([w;s,e], \boldsymbol{o}_s^e)) \tag{3}$$

Finally, $p([k;t] \mid j)$ is calculated as the kernel output probability normalized by the state output probability:

$$p([k;t] \mid j) = \frac{c_{jk} \cdot p(\boldsymbol{o}_t \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{l=1}^{L} c_{jl} \cdot p(\boldsymbol{o}_t \mid \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)} \tag{4}$$

in which L is the number of Gaussian components of state j; c_{jk} is the component weight of kernel k; $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are the mean vector and covariance matrix of that kernel, respectively.

2.4. Model Based Clean Speech Synthesis

The model based clean speech estimate for Wiener filtering is constructed in two steps. In the first step, for each time frame t, the expected values of the mean and covariance of the clean speech feature are calculated using the kernel posterior probabilities along with the kernel parameters (diagonal covariance matrix is used in our experiments):

$$\begin{aligned} \hat{\boldsymbol{\mu}}(t) &= E\{\boldsymbol{\mu} \mid \boldsymbol{\mu}_k, p([k;t] \mid \boldsymbol{o}_1^T)\} \\ &= \textstyle\sum_{k=1}^{K} p([k;t] \mid \boldsymbol{o}_1^T) \cdot \boldsymbol{\mu}_k \end{aligned} \tag{5}$$

and

$$\hat{\boldsymbol{\Sigma}}(t) = E\{[\boldsymbol{\mu}-\hat{\boldsymbol{\mu}}][\boldsymbol{\mu}-\hat{\boldsymbol{\mu}}]^\top \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, p([k;t] \mid \boldsymbol{o}_1^T)\}$$
$$= \sum_{k=1}^{K} p([k;t] \mid \boldsymbol{o}_1^T) \cdot (\boldsymbol{\Sigma}_k + \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top) - \hat{\boldsymbol{\mu}}(t)\hat{\boldsymbol{\mu}}(t)^\top \quad (6)$$

In contrast to former approaches that only use static means of the speech feature, in Eqs. (5) and (6), we calculate the expected values using both static and dynamic (first- and second-order delta) features of the kernels. As a result, in the second step, the model based estimate of the clean speech S_3 can be synthesized in ML sense by imposing the explicit constraint between its static and dynamic features. Following this way, the accuracy of the model based clean speech for Wiener filtering can be improved by considering not only its static "level", but also its dynamic "trend".

It can be seen from [9] that, the ML solution of S_3 can be obtained by solving the weighted normal equation:

$$\mathbf{W}^\top\mathbf{U}^{-1}\mathbf{W}\boldsymbol{C} = \mathbf{W}^\top\mathbf{U}^{-1}\boldsymbol{M} \quad (7)$$

where $\mathbf{W}$ is the weighting matrix for computing the dynamic features, via $\mathbf{W}$ which imposes the static-dynamic constraint [9], and

$$\boldsymbol{C} = [\boldsymbol{c}(1)^\top, \boldsymbol{c}(2)^\top, \ldots, \boldsymbol{c}(T)^\top]^\top \quad (8)$$

is the synthesized clean speech S_3 in terms of its MFCC parameter sequence;

$$\mathbf{U}^{-1} = \mathrm{diag}[\hat{\boldsymbol{\Sigma}}^{-1}(1), \hat{\boldsymbol{\Sigma}}^{-1}(2), \ldots, \hat{\boldsymbol{\Sigma}}^{-1}(T)]$$
$$\boldsymbol{M} = [\hat{\boldsymbol{\mu}}(1)^\top, \hat{\boldsymbol{\mu}}(2)^\top, \ldots, \hat{\boldsymbol{\mu}}(T)^\top]^\top \quad (9)$$

are the matrices composed by the expected mean and covariance values calculated by Eqs. (5) and (6). Because of the band diagonal structure of $\mathbf{W}^\top\mathbf{U}^{-1}\mathbf{W}$, Eq. (7) can be solved efficiently in a time-recursive manner by the QR decomposition.

2.5. Wiener Filtering and Constrained Second Pass Decoding

Wiener filtering of the Mel-filter bank energy is performed in the linear spectral domain, so the estimated cepstral mean $\overline{S_1}$ is first added back to S_3 (i.e., $\boldsymbol{c}(t)$ in the cepstral domain), and an Inverse Discrete Cosine Transform (IDCT) followed by an exponential transform is performed:

$$S_3^{\mathrm{FBE}}(t) = \exp\left\{\mathrm{IDCT}[\boldsymbol{c}(t) + \overline{S_1}]\right\} \quad (10)$$

where the superscript "FBE" stands for Mel-filter bank energy.

Meanwhile, the input noisy speech X and the estimated noise N can also be converted to Mel-filter bank energies. Wiener filtering can then be performed to get the final estimate of the clean speech, S_4:

$$S_4^{\mathrm{FBE}}(t) = \frac{S_3^{\mathrm{FBE}}(t)}{S_3^{\mathrm{FBE}}(t) + N^{\mathrm{FBE}}(t)} \cdot X^{\mathrm{FBE}}(t) \quad (11)$$

In the last step, S_4^{FBE} is converted to the cepstral domain, where its cepstral mean is removed so that a second pass decoding can be performed. As we have already got a word graph after the first pass decoding, it is then possible to re-score the word graph or to re-decode S_4 within the constrained search space defined by the word graph. Because in most cases, the Word Graph Error Rate (GER, computed by determining the sentence through the word graph that best matches the reference in terms of word errors) is considerably lower than the Word Error Rate (WER) of the first best hypothesis, the word graph constrained second pass decoding can achieve a reasonably low WER while significantly reducing the computational cost of another pass of completely free decoding.

Aurora 2 Reference Word Error Rate				
Testing Set	Set A	Set B	Set C	Overall
WER (%)	38.42	42.59	30.57	38.52

Table 1. Aurora 2 reference Word Error Rate using MFCC_0DA.

Signal Processing Based Speech Enhancement				
Testing Set	Set A	Set B	Set C	Overall
WER (%)	25.27	25.47	22.87	24.87
Relative	34.23%	40.21%	25.18%	35.44%
GER (%)	5.59	6.07	4.58	5.58

Table 2. Performance of signal processing based speech enhancement (absolute, relative to the reference, and graph error rate).

3. EXPERIMENTS

3.1. Experimental Setup

The word graph based feature enhancement method has been tested on the Aurora 2 database. Because transforms between spectral domain and cepstral domain are needed, a 39-dimensional MFCC feature vector, including c_0 to c_{12} and their first and second order dynamic coefficients, was used in our system. An HMM used for both enhancement and recognition was trained with ETSI provided scripts [10] using HTK. As a result, 11 whole word digit models were trained, each with 16 emitting states and 3 Gaussian components per state. A three-state silence model was also constructed with 6 Gaussian components per state, while a one-state short pause model, tied with the central state of the silence model, was used.

Because our approach uses the same clean trained HMM for both enhancement and recognition, the Aurora 2 clean-condition training scenario is just suitable to evaluate the performance of our algorithm. Besides the baseline system, three feature enhancement methods have been compared in our experiments: (a) Signal processing based speech enhancement using Weighted Noise Estimation and CMN; (b) The GMM based feature enhancement method similar to that of in [5, 6], and (c) The proposed word graph based feature enhancement method. Note that method (a) actually serves as a starting point for the latter two feature enhancement methods (b) and (c) (reference to Fig. 1).

3.2. Signal Processing Based Speech Enhancement

The reference word error rate of Mel-cepstrum on the Aurora 2 database is given in Table 1 (slightly better than [10] because we use c_0 instead of log-energy). After signal processing based speech enhancement, the word error rate and word graph error rate are shown in Table 2. It is shown from the table that, signal processing based feature enhancement consistently improves the recognition performance, and the overall relative error rate reduction is 35.44%. Moreover, the GER of the decoded word graph is significantly lower than the WER of the first best hypothesis (only about $1/4 \sim 1/5$). So the word graph constructed in the first pass decoding can be used not only to guide the feature enhancement process, but also to narrow the search space in the second pass decoding.

3.3. GMM Based Feature Enhancement

Conventional GMM based feature enhancement method was performed in our experiments for comparison purpose, and its perfor-

GMM Based Feature Enhancement				
Testing Set	Set A	Set B	Set C	Overall
WER (%)	22.85	21.45	21.99	22.12
Relative	40.52%	49.64%	28.05%	42.58%

Table 3. Performance of the GMM based feature enhancement.

Word Graph Based Feature Enhancement				
Testing Set	Set A	Set B	Set C	Overall
WER (%)	16.79	15.92	15.68	16.22
Relative	56.31%	62.62%	48.70%	57.89%
Word Graph Based Feature Enhancement (UD)				
Testing Set	Set A	Set B	Set C	Overall
WER (%)	16.58	15.87	15.70	16.12
Relative	56.85%	62.73%	48.66%	58.15%

Table 4. Performance of the word graph based feature enhancement (UD = unconstrained second pass decoding).

mance is given in Table 3. In this case, the acoustic model in Fig. 1 is replaced by a GMM with 128 Gaussian components, and the word graph is a single path of the GMM states. As shown in the table, GMM based feature enhancement reduces the word error rate further, and the overall relative error rate reduction is improved to 42.58%.

3.4. Word Graph Based Feature Enhancement

The performance of the proposed word graph based feature enhancement method is shown in Table 4. The results show that the performance is further improved over the GMM based method, and an overall relative error rate reduction of 57.89% is obtained. As we were using word graph constrained second pass decoding, this result is obtained with a minor increase of the computational cost. We also compared the WER if we perform an unconstrained free decoding in the second pass, and the result is given in the "UD" part of Table 4. The experimental results suggest that the difference between the two decoding scenarios is minimal, and the costly unconstrained second pass decoding is not necessary. This is true especially when the GER of the word graph is sufficiently low.

Fig. 2 shows the recognition results of the three enhancement methods as a function of SNR. The word graph based feature enhancement method consistently achieves the best performance at different SNRs. It outperforms the other two methods especially when SNR is low.

4. CONCLUSIONS

In this paper we presented a word graph based feature enhancement method for robust speech recognition in noise. This method performs signal processing based speech enhancement as a foundation, and then using it to construct the word graph. After that, a maximum likelihood estimate of the model based clean speech is synthesized using the word graph and clean trained speech model, so Wiener filtering can be carried out to get the output speech feature for recognition. The word graph based method enables us to directly guide the feature enhancement process with respect to the model for recognition, and the temporal resolution as well as the dynamic feature of the HMM can also be exploited. Experimental results suggest that the word graph based feature enhancement method outperforms conventional signal processing based and GMM based methods under different SNRs. In future work, we are planning to adapt the method to achieve an improved performance when not only the clean speech, but also the noise statistical information, can be observed.

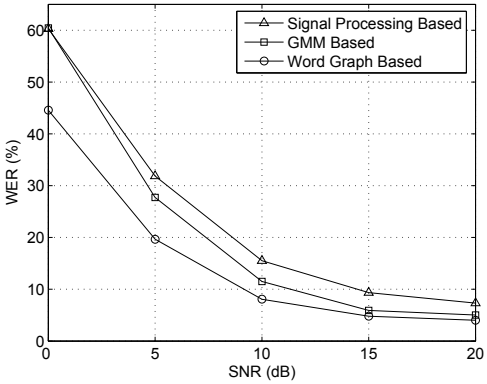

Fig. 2. Performance of the enhancement methods against SNR.

5. REFERENCES

[1] P. Lockwood and J. Boudy, "Experiments with a Nonlinear Spectral Subtractor (NSS), Hidden Markov Models and the projection, for robust speech recognition in cars," *Speech Communication*, vol. 11, no. 2-3, pp. 215–228, 1992.

[2] F. Liu, R. Stern, X. Huang, and A. Acero, "Efficient cepstral normalization for robust speech recognition," in *Proc. ARPA Human Language Technology Workshop*, 1993, pp. 69–73.

[3] M. J. F. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination," *IEEE Trans. on Speech and Audio Processing*, vol. 4, pp. 352–359, 1996.

[4] M. J. F. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," Tech. Rep., CUED/F-INFENG/TR 291, Cambridge University, 1997.

[5] J. C. Segura, A. de la Torre, M. C. Benitez, and A. M. Peinado, "Model-based compensation of the additive noise for continuous speech recognition. Experiments using the AURORA II database and tasks," in *Proc. Eurospeech2001*, 2001, vol. 1, pp. 217–220.

[6] T. Arakawa, M. Tsujikawa, and R. Isotani, "Model-based Wiener filter for noise robust speech recognition," in *Proc. ICASSP2006*, 2006, vol. 1, pp. 537–540.

[7] M. Kato, A. Sugiyama, and M. Serizawa, "Noise suppression with high speech quality based on weighted noise estimation and MMSE STSA," *Electronics and Communications in Japan*, vol. 89, no. 2, pp. 43–53, 2006.

[8] W. K. Lo and F. K. Soong, "Generalized posterior probability for minimum error verification of recognized sentences," in *Proc. ICASSP2005*, 2005, pp. 85–88.

[9] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP2000*, 2000, pp. 1315–1318.

[10] H. G. Hirsch and D. Pearce, "The Aurora experimental framework for the performance evaluation of speech recognition under noisy conditions," in *Proc. ISCA ITRW ASR2000*, 2000, pp. 181–188.