A Speech Processing Front-End with Eigenspace Normalization for Robust Speech Recognition in Noisy Automobile Environments

Kaisheng Yao*, Erik Visser, Oh-Wook Kwon, and Te-Won Lee

Institute for Neural Computation, University of California at San Diego
9500 Neil Garson Drive, La Jolla, CA, 92093-0523
*kyao@ucsd.edu, visser@salk.edu, owkwon@ucsd.edu, tewon@ucsd.edu

Abstract

A new front-end processing scheme for robust speech recognition is proposed and evaluated on the multi-lingual Aurora 3 database. The front-end processing scheme consists of Mel-scaled spectral subtraction, speech segmentation, cepstral coefficient extraction, utterance-level frame dropping and eigenspace normalization module. We also investigated performance on all language databases by post-processing features extracted by the ETSI advanced front-end with an additional eigenspace normalization module. This step consists in linear PCA matrix feature transformation followed by mean and variance normalization of the transformed cepstral coefficients. In speech recognition experiments, our proposed front-end yielded better than 16 percent relative error rate reduction over the ETSI front-end on the Finnish language database. Also, more than 6% in average relative error reduction was observed over all languages with the ETSI front-end augmented by eigenspace normalization.

1. Introduction

With the increased demand of hands-free speech communication, including mobile devices, the need for automatic speech recognition in hands-free environments has increased. In these settings, speech signals are often distorted by background noise, acoustic reverberation, and linear channel distortion before being processed by a speech recognition system. Recent progress in the speech recognition research in hands-free communications has resulted in many promising approaches. These approaches include speech enhancement [1], feature enhancement (or robust feature extraction) [2][3][4], and model-based back-end processing [5][6], which vary in algorithm complexity and latency.

As part of an effort to standardize the front-end feature analysis algorithm, the ETSI Aurora group has released several proposals [1]. A database, Aurora 3 [7], including speech utterances collected in cars through close-talking and hands-free microphones has been assembled for evaluation of algorithms as compared to the baseline performance by ETSI.

In this paper, we demonstrate the effectiveness of a front-end by evaluating its performance on the Aurora 3 database. The front-end consists of several processing algorithms that are designed with an eye to simplicity and efficiency. They are Mel-warped spectral subtraction for speech enhancement, speech segmentation, eigenspace normalization for feature normalization and utterance-level frame dropping. Eigenspace normalization and utterance-level frame dropping work independently from speech enhancement, thus they are fit for post-processing of extracted features. Notice that the hidden Markov model parameters are not adapted or compensated in this work, for the consideration of achieving low-latency and simple algorithms for robust speech recognition.

Our results show that these algorithms can achieve improved performance compared to the ETSI advanced front-end [1]. In particular, post-processing of speech features extracted from the ETSI advanced front-end by eigenspace normalization can improve the ETSI baseline performance. In the following Section 3.1, we describe the Mel-scaled spectral subtraction algorithm for speech enhancement. Section 3.2 presents a speech segmentation method based on tracking the evolution of the variance of linear spectra. In Section 3.3, an eigenspace normalization algorithm is proposed. Section 3.4 presents non-parametric utterance-level frame dropping. Experiments are shown in Section 4, and discussed in Section 5. Section 6 concludes the paper.

2. Database and the System

The Aurora 3 database has the following configuration. Digits strings of four European languages, Finnish, Spanish, German, and Danish, were recorded using close-talking and hands-free microphones in three driving conditions: quiet, low noise, and high noise. Quiet refers to the condition that a car is stopped or idling; Low noise refers to the condition where the car was driven at speeds of 40-60 km/h; High noise refers to the condition that the car was driven at speeds of 100-120 km/h. Training and testing sets were designed to include these driving conditions, and were divided into well-matched (WM), medium-matched (MM), and high-mismatched (HM) situations.

The structure of the back-end model has been specified to a simple left-to-right 16-state 3-mixture per state whole word hidden Markov model (HMM). The silence model is a 1-state 6-mixture HMM and the short-pause model is 1-state 6-mixture. The baseline for comparison provided by ETSI is its advanced front-end consisting of noise reduction by a two-stage Wiener filter, signal-to-noise ratio dependent waveform processing, cepstral calculation, blind equalization, coding/decoding, and frame dropping. Compared to a standard front end without noise reduction [7], this advanced front-end has achieved robustness to background noise and linear channel distortion [1].

3. Proposed Front-End

A diagram of the proposed front-end is shown in Figure 1. For speech signals sampled at 8kHz, a 256 points FFT transforms the 20 ms waveform signals into the linear frequency domain.
Time-shift of the FFT is 10ms. For each frame, pre-emphasis of the waveform with a pre-emphasis coefficient of 0.97 and smoothing using a Hamming window are applied. Signal-to-noise ratio (SNR) is improved by Mel-scaled spectral subtraction. Voice activity is decided for segmentation of speech in the input utterances. Mel-scale cepstral coefficients (MFCCs including C0) and their first- and second-order derivatives with a regression length of 5 are obtained for each frame. Those frames which continuously have a distance (defined in Section 3.4) smaller than a threshold are dropped. The feature vectors are normalized in an eigenspace in the module Eigenspace Normalization. Detailed descriptions of these modules are provided in later sections.

3.1. Mel-Scaled Spectral Subtraction

To improve the SNR of the signal, modified spectral subtraction is applied to the power spectrum of the signal. The noise spectrum is assumed to be additive to the spectrum of the clean speech. Estimates of noise power spectrum are conducted in the linear frequency domain by a first order recursion in noise frames assigned from a voice activity decision (VAD). An instantaneous filter is estimated at each frame by

$$[H_t(k)]^2 = \max \{ \frac{|Y_t(k)|^2 - \gamma(k)|\hat{N}_t(k)|^2}{|Y_t(k)|^2} \beta \}$$  

(1)

where $t$ and $k$ each denote time- and frequency-index. $|\hat{N}_t(k)|^2$ and $|Y_t(k)|^2$ are the estimated noise power spectrum and the power spectrum of the observed signal, respectively. $\beta$ works as spectral flooring. $\gamma(k)$ is the noise overestimation factor decided similarly as in [8], which has the following equation,

$$\gamma(k) = \frac{G F_{\text{min}} - G F_{\text{max}}}{20.0} \frac{R_{\text{post}}(k) + G F_{\text{max}}}{S N R_{\text{post}}(k)}$$  

(2)

where $G F_{\text{min}}$ and $G F_{\text{max}}$ each are 1.0 and 4.125 in this work. The $S N R_{\text{post}}(k)$ is the posterior SNR \footnote{The definition of the posterior SNR is different from other work, e.g., [9].} at frame $k$, calculated as the decibel of the ratio of the enhanced power spectrum by a parallel spectral subtraction and the estimated noise power spectrum. This parallel spectral subtraction yields enhanced power spectra without considering noise overestimation.

The filter is smoothed in time to reduce variances by

$$|H_t(k)| = 0.95 \times |H_{t-1}(k)| + 0.05 \times |H_{t+1}(k)|$$

The filters are further smoothed in the frequency domain by warping them to Mel-scale. Spectral subtraction is conducted in the Mel-scale frequency domain.

3.2. Speech Segmentation

The speech segmentation is modified from ETSI proposal [1], and it works on the denoised signals. At each frame, the spectral variance in the linear frequency domain of the denoised signals is calculated. The acceleration of the spectral variance is also computed. A tracker of the spectral variance is updated when the acceleration is within a limit (it was set to 1.60 in this work), or the current variance is within a range decided by the previously calculated variance. If the currently calculated variance is larger than a certain threshold decided by the tracker, the current frame is considered to be a hypothesis of non-speech; otherwise, the current frame is a hypothesis of non-speech. The final decision of speech/non-speech is conducted by smoothing the hypothesis with incurring a hangover scheme which has a timer of 30 frames.

3.3. Eigenspace Normalization

In the literature, many normalization methods have been proposed, for example, histogram normalization [10], and mean/variance normalization plus time-smoothing [2], in order to compensate for the distortions in the feature coefficients due to additive and convolutive noise. This section proposes an algorithm for feature normalization in eigenspace.

Our algorithm is motivated by the following observations. For a $K$-dimensional feature vector, denoted by $\mathbf{x}_t$ with element $x_t^k$ at each index $k$, a continuous-valued latent representation can be constructed, i.e.,

$$Z_t \sim N_t(0, I)$$  

(3)

$$\mathbf{x}_t = \Lambda Z_t + \nu_t$$  

(4)

where superscript $c$ denotes the cepstral domain, and $N_t(0, I)$ denotes a standard normal distribution with dimension $L$. Usually, we assume that $L \approx K$. $\Lambda$ is the loading matrix mapping $Z_t \in R^L$ to $\mathbf{x}_t \in R^K$.

Furthermore, we assume that $\nu_t \approx 0$. In this situation, the above formulae give the principal component analysis (PCA) of $X^c(t)$. Note that this is a reasonable assumption, if $X^c(t)$ is not distorted.

In the situation of distortion in $\mathbf{x}_t$, given a fixed $\Lambda$, the distortion can be viewed in the space spanned by $Z_t \in R^L$. Since, when the $\Lambda$ is learned from $\mathbf{x}_t$ by PCA, the mean and variance in the space are assumed to be zero and one, respectively, the mean and variance of the transformed $Z_t$ from distorted $X^c(t)$ should be respectively forced to zero and one. Feature normalization can be achieved as a result. We outline the processing scheme in the following.

1. Learning PCA matrix $\Lambda$: A PCA matrix is learned from a training set. In this work, it is learned from the well-matched training set for each language, assuming that the cepstral feature vector (MFCCs and their first- and second derivatives) can be modeled by the latent presentation in Eq. (3) and Eq. (4). $\Lambda$ is obtained as $U V^{1/2}$, where $U V^{T} = E[X^c_t X^c_t^T]$. $U$ and $V$ each are the eigenvector matrix and eigenvalue matrix of the sample covariance matrix $E[X^c_t X^c_t^T]$.

2. Eigenspace normalization in each utterance:

   (a) Map $\mathbf{x}_t^c$ to $Z_t \in R^L$: $Z_t = \Lambda^{-T} X_t^c = V^{-1/2} U^{T} X_t^c$.  

Figure 1: The diagram of the proposed front-end.
3.4. Utterance-Level Frame Dropping

In order to drop some frames that are noise intervals, parametric methods assuming trained models for modeling statistics of silence and speech segments have been proposed in the literature. This section proposes a non-parametric method for frame dropping. The benefit of the approach is its simplicity and working without training.

A measure of quantile distance is defined here as

\[ d_t = Q^{\text{max}}_t - Q^{\text{min}}_t \]  

where \( Q^{\text{max}}_t \) and \( Q^{\text{min}}_t \) denote the largest and the smallest quantile of the cepstral coefficients at frame \( t \). A plot of the histogram of the distance for each utterance usually shows a strong bimodal distribution of the distance, where one mode corresponding to larger values belongs to speech segments and the other mode for small values corresponds to non-speech segments. An example of the histogram is plotted in Figure 3.

Based on the histogram of the distance, a threshold \( \alpha \) can be determined for each utterance, where those frames that continuously fall below the threshold for a certain length, say \( \tau \), are decided as non-speech segments. Note that this frame dropping method works directly in the cepstral domain.

4. Experiments

In this section, we conducted two sets of experiments. The first set compared the performance of the described front-end in the above section with that of the ETSI advanced front-end [1]. In the second set of experiments, extracted features from the advanced front-end by ETSI [1] were post-processed by eigenspace normalization in Section 3.3 and utterance-level frame dropping in Section 3.4.

4.1. Performance of the Proposed Front End on the Finnish Language Database

We chose the Finnish subset for evaluation of the proposed front-end, because it is by far the largest dataset and hence provides statistically reliable comparison of methods. The \( \alpha \) and \( \tau \) in the utterance-level frame dropping were respectively set to 0.3 and 20. The first and second column of Table 1 show the word error rate (WER) of the ETSI advanced front end [1] and those of our proposed front-end.

It is observed in Table 1 that our proposed front-end has better performance in MM and HM conditions. In particular, there are 44% and 38% relative improvement over the ETSI front-end in MM and HM, respectively. In WM condition, our front-end has worse performance than the ETSI advanced front-end. As a whole, the proposed front-end achieves 16.86% relative error rate reduction over the ETSI advanced front-end in the Finnish subset.

<table>
<thead>
<tr>
<th>Conditions</th>
<th>ETSI</th>
<th>Proposed</th>
<th>Relative Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>WM</td>
<td>13.51</td>
<td>8.75</td>
<td>-30.9%</td>
</tr>
<tr>
<td>MM</td>
<td>13.59</td>
<td>8.30</td>
<td>-43.4%</td>
</tr>
<tr>
<td>HM</td>
<td>13.39</td>
<td>8.20</td>
<td>-40.7%</td>
</tr>
<tr>
<td>WER reduction (%)</td>
<td></td>
<td>16.86</td>
<td></td>
</tr>
</tbody>
</table>
Table 4: Aurora 3 reference word error rate

<table>
<thead>
<tr>
<th>Well (x40%)</th>
<th>Finnish</th>
<th>Spanish</th>
<th>German</th>
<th>Danish</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.91%</td>
<td>3.36%</td>
<td>4.89%</td>
<td>6.63%</td>
<td>4.70%</td>
<td></td>
</tr>
<tr>
<td>Mid (x35%)</td>
<td>19.08%</td>
<td>6.08%</td>
<td>9.16%</td>
<td>18.51%</td>
<td>13.21%</td>
</tr>
<tr>
<td>High (x25%)</td>
<td>13.39%</td>
<td>8.45%</td>
<td>8.75%</td>
<td>20.41%</td>
<td>12.75%</td>
</tr>
<tr>
<td>Overall</td>
<td>11.59%</td>
<td>5.58%</td>
<td>7.35%</td>
<td>14.23%</td>
<td>9.65%</td>
</tr>
</tbody>
</table>

Table 5: Aurora 3 word error rate

<table>
<thead>
<tr>
<th>Well (x40%)</th>
<th>Finnish</th>
<th>Spanish</th>
<th>German</th>
<th>Danish</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.83%</td>
<td>3.41%</td>
<td>3.81%</td>
<td>6.52%</td>
<td>4.39%</td>
<td></td>
</tr>
<tr>
<td>Mid (x35%)</td>
<td>10.81%</td>
<td>5.52%</td>
<td>10.32%</td>
<td>20.51%</td>
<td>11.79%</td>
</tr>
<tr>
<td>High (x25%)</td>
<td>13.67%</td>
<td>8.24%</td>
<td>7.59%</td>
<td>18.27%</td>
<td>11.94%</td>
</tr>
<tr>
<td>Overall</td>
<td>8.73%</td>
<td>5.36%</td>
<td>7.03%</td>
<td>14.35%</td>
<td>8.87%</td>
</tr>
</tbody>
</table>

Figure 4: Aurora 3 word error rate results obtained with the augmented ETSI advanced front-end using post-processing by eigenspace normalization and utterance-level frame dropping.

4.2. Performance of Post-Processed ETSI Advanced Front-End on the Complete Aurora 3 Database

Eigenspace normalization and utterance-level frame dropping are applied to 39-dimensional MFCC features extracted by the advanced front-end [1]. The $\alpha$ and $\tau$ in the utterance-level frame dropping were set to 0.1 and 40, respectively. With this post-processing, the augmented system has an averaged relative performance improvement of 6.48%, over the baseline (see Figure 4). In particular, we observed significant improvement in the MM test on the Finnish subset, which reaches 43% relative gain by using the post-processing. Performance improvement in German subset is also significant, and it reaches 7%.

In fact, we found that the gain from utterance-level frame dropping is marginal. This is a different observation from the experiment carried out in the previous section. This might be due to sufficient speech segmentation performance by the ETSI advanced front-end.

Post-processing on Danish set of the database did not yield improved performance. Because the PCA matrix for feature transformation requires sufficient amount of data for training, the relatively small size of the Danish database presents a difficulty for learning reliable PCA matrix. This observation raises the issue of robust training of PCA matrix or incorporating proper prior information on PCA matrix.

5. Discussion

It is interesting to note the different performances for the HM case on the Finnish subset obtained by the proposed front-end and the augmented ETSI advanced front-end with post processing. The proposed front-end achieved 38% relative gain on the HM set, whereas the augmented ETSI advanced front-end had negative relative gain. This large performance discrepancy may be attributed to different front-end processing, and the underlying mechanism for good performance of the combination of eigenspace normalization with a front-end processing deserves further study.

Due to different implementation of the ETSI advanced front-end [1] by our team, we obtained slightly different performances from the reported baseline performances. However, we found that eigenspace normalization can provide consistent relative gains on the Finnish, Spanish, and German subsets of the Aurora 3 database, compared to baseline performance. Averaged relative error rate reductions by the eigenspace normalization are 19.88%, 8.34%, and 17.56% on the Finnish, Spanish, and German databases, respectively. We plan to conduct further research on this promising approach of eigenspace normalization.

6. Conclusions

The advanced ETSI front-end has been designed with speech enhancement, blind equalization, and frame dropping modules, which have achieved robustness to additive noise and linear channel distortion. Similar processing methods based on speech enhancement and blind equalization may yield performance close to the ETSI advanced front-end. However, our proposed front-end processing achieves improved performance over the ETSI advanced front-end in the Finnish subset.

The originality and effectiveness of the proposed front-end relies on two novel processing methods: one is an eigenspace normalization method, and the other is a non-parametric frame-dropping method. Eigenspace normalization is suitable for ETSI advanced front-end feature post-processing. In that case, we have observed significant gain in some subsets of the database. Fine-tuning of the proposed front-end may yield even better results. Alternatively, the ETSI advanced front-end can be combined with eigenspace normalization to achieve robustness to different environment related distortions.

7. References