HMM ADAPTATION AND MICROPHONE ARRAY PROCESSING FOR DISTANT SPEECH RECOGNITION

Jim Kleban, Yifan Gong

CAIP, Center for Advanced Information Processing
Rutgers University, Piscataway, NJ
kleban@caip.rutgers.edu

Media Technologies Laboratory
Texas Instruments, Dallas, TX
Yifan.Gong@ti.com

ABSTRACT

Connected strings of seven digits from the TIDIGITS database were recorded in a reverberant office room for evaluation using microphone array processing and HMM, Hidden Markov Model, adaptation. A sixteen-channel linear microphone array records a distance speech database useful for further experimentation. The adaptation techniques of Parallel Model Combination (PMC) and Maximum Likelihood Linear Regression (MLLR) are evaluated and compared. The effect of the number of adaptation utterances and number of vectors per class for the regression tree in order to optimize MLLR results are studied. Results show, compared to no adaptation, 40% word error reduction (improvement to 4.2%) for PMC and 60% word error reduction (improvement to 3.0%) for MLLR.

1. INTRODUCTION

In a noisy, reverberant environment such as a conference room or an automobile, the performance of speech recognizers decreases sharply. Interfering noise sources and reflections off of walls, floors, ceilings, and objects degrade the quality of the incoming speech signal. A microphone array can be used to spatially select the desired speech source by rejecting interfering sources and suppressing diffuse noise [1].

Speech recognition accuracy can be further improved when the microphone array processing is combined with adaptation of the speech recognition models, as reported in previous works, e.g. [2][3]. The adaptation algorithms are based on maximum likelihood linear regression [5] or artificial neural network [3]. While these algorithms give good recognition improvement, a large amount of noisy speech is typically required to train the adaptation parameters. Such noisy speech may not be available in practice. This work supplements previous experiments involving digit recognition with microphone arrays and HMM adaptation. The basic assumption is that the remaining speech distortion after microphone array processing can be considered as an additive noise to the speech signal and can be compensated by parallel model combination (PMC) [4]. PMC produces a noisy speech model based on clean speech model and noise statistics, and thus does not require speech collected in the noisy environment. Besides, in the set of experiments reported here, a microphone array with a greater number of channels than previously used is implemented.

We also give an experimental comparison of PMC and MLLR on the microphone array data.

Figure 1: Microphone Array input to GMHMM for evaluation of hands-free digit recognition in reverberant environments

2. SYSTEM DESCRIPTION

Figure 1 shows the overall system used to collect, process, and recognize the speech. The system consists of two major parts: the microphone array processing and speech recognizer.

2.1. Microphone Array Processing

A linear microphone array of sixteen channels with three nested frequency bands collects data for beamforming. The array consists of microphones with spacings of 4 cm, 8 cm, and 16 cm. The maximum wavelength of this array, geared towards the speech recognition application, is 4 kHz. The total length of the array is 1.28 meters. FIR filters separate the appropriate microphone channels into three temporal frequency bands of 100-1000Hz, 1000-2000Hz, and 2000-4000Hz. The sixteen channels are sampled at a rate of 48 kHz. Delay and Sum beamforming, time-delay compensation (TDC), processes the incoming array signals. This performs optimally under a diffuse, non-coherent between microphones, noise field. The summed output over k channels of the microphone array will be:

\[ X(t) = \sum_k s(t - \tau_k) + n_k(t) \]  

where X(t) is the output, s(t) is the speech signal, \( \tau_k \) is time delay to microphone k, and \( n_k(t) \) represents the additional noise at
Assuming a non-coherent noise field, the array gain in dB increases with the square root of the number of microphones as:

\[
Array \ Gain(SNR) = 10 \cdot \log_{10} \left( K \sigma_i^2 / \sigma_n^2 \right) - 20 \cdot \log_{10}(K) \sqrt{dB} = 20 \cdot \log_{10}(\sqrt{K}) dB
\]

where \( K \) is the total number of microphones, and \( \sigma_i^2 \) and \( \sigma_n^2 \) represent the variance of the speech and noise signals.

Given the microphone spacings and total length of array, the pressure wave equation under far field assumptions is used to show beamformer patterns in figure 2. The figure shows that for the sub-banded array, relatively narrow beams are found at low and high frequencies. A hamming window shading is applied across the array.

Processed array output is converted into a feature vector for input into the GMHMM recognizer.

2.2. Speech Recognition and Model Adaptation

The speech recognizer used in this experiment is an HMM recognizer with Gaussian mixture probability density functions. The models are trained on TIGIDITS database, which contains a large number of clean speech data comprising both male and female speakers.

The input from the microphone array is blocked into 32ms frames with Hamming windowing, with 20ms between frames. Feature vectors of eight cepstral coefficients, with time-derivatives, are used as inputs into the speech recognizer. Pre-emphasis and mean normalization of the cepstral vector are also applied.

The speech recognizer utilizes a grammar which allows the recognizer to return output strings of length between one and seven digits.

2.2.1. PMC Adaptation

In PMC, clean speech HMMs are combined with statistics about the background noise to create 'noisy' HMM models. The resulting model accounts for the noisy environment. Mean vectors of the cepstral coefficients for the noise background are combined with the clean-speech trained HMM models in the linear spectrum domain in the following fashion:

\[
\bar{\mu}_i = g \cdot \mu_i + n
\]

The value of the gain, \( g \), is adjusted experimentally to determine the optimal weighting of the original set of clean model mean values.

We compensate both static and dynamic MFCC model parameters. The compensated dynamic MFCC is the continuous time derivative of compensated static parameters. This is an approximation to the discrete nature of dynamic parameters, with resulting expression extremely simple and easy to implement. Variances are not adapted in this study.

The Gaussian distribution of the noise is estimated in the MFCC domain with the first \( N \) frames of the test utterance. To deal with the channel distortion introduced by the distant recording line, the mean vector of some utterances recorded in the target environment is added to the clean model mean vectors, which are MFCC-mean-normalized, before applying PMC. During the recognition, the mean-normalization is de-activated.

2.2.2. MLLR Adaptation

MLLR, maximum likelihood linear regression, adapts the mean values of the gaussian distributions of the HMMs of the speech recognizer. A set of linear transformations is found using a set of adaptation speech data. These transformations are optimal in a maximum likelihood sense. The adaptation data is recorded in the distant talking environment that it is desired to adapt the clean-speech trained models to. The advantage of MLLR technique is that the amount of adaptation data is much smaller than the amount of data which would be otherwise required to retrain the recognizer in the distant talking environment. In our experiment, a total of 220 training utterances are available. As stated in the results section, the number of utterances used for
adaptation can be further reduced without incurring substantial recognition performance loss. The mean vector adaptation for class $c$ is given by:

$$\hat{\mu}_c = \mathbf{A}_c \mu_i + b_c \quad (4)$$

where $\mu_i$ are the model gaussian means, $\mathbf{A}_c$ is the transformation matrix per class, and $b_c$ is the linear bias. New $\mu$ are found by maximizing the likelihood of the adaptation data:

$$\max \ P(O|A_c, b_c) = \max \sum_{\theta \in \Theta} P(O, \theta|A_c, b_c) \quad (5)$$

A matrix $\mathbf{A}$ is computed for every class $c$ and a class is shared by a set of Gaussian distributions.

In order to achieve reliable parameter estimation for small amount of adaptation data and high resolution for large amount of data, the number of classes in the system is data-driven, and is based on the amount of data available for adaptation [6]. To do so, each class is associated with a number that specifies the minimum number of vectors required to train the class. The number is used to control the number of adaptation classes, which is adjusted to determine the optimal number of transformations to be generated. For a high number of vectors per class, a fewer number of matrices $\mathbf{A}_c$ are used to transform the Gaussian means. For a lower number of vectors per class, many $\mathbf{A}_c$ matrices will be used. The optimal number of vectors per class is found by a series of trials, and this uses the granularity best achieved by the collected training data.

3. RECORDINGS

Recordings were taken in a medium-sized office space. Digitized TIDIGITS speech files were played back through speakers and were recorded over all sixteen channels of the microphone array. A single channel from the center of the array was used in all comparisons. The microphones used in the recordings were Panasonic WM-54B omnidirectional electrets, which are inexpensive and provide a relatively flat frequency response until 11kHz. A single cone speaker, the Fostek 6301-BE, was used to play the speech samples. The capture system was a 200 MHz PC with a SCSI hard drive for fast disk writes. A Dakota PCI card and two Tango24 rack-mounted A/D modules converted and transferred the data to the PC. Cool Edit Pro software was used to write the incoming channels to file. An additional 40 dB gain before input into the Tango 24’s was provided by an analog pre-amp box designed at Rutgers.

The data was collected over a period of eight hours in one evening sitting. Speech data was taken at an angle of $0^\circ$ (boresight) of the microphone array at a distance of exactly 10.0 feet (3.05 meters.) The speaker was placed at a height matching that of the array, corresponding to the level of a person sitting at a table. The sound output level at the speaker was measured to range from 75-85 dBA SPL. The background level of the lab room, with noise due to CPU fans and other building noise, was measured to range from 65-70 dBA SPL.

A set of 1242 seven-digit strings, spoken by 56 male and 57 female speakers, was recorded to test the performance of the recognizer. A total of 220 training utterances, taken from the initial clean-speech training set, were also recorded for use with MLLR.

4. RESULTS

Results are compiled for single and array microphones, with and without adaptation. WER (word error rate) is the primary measure of recognizer accuracy. WER was defined as the number of incorrect digit substitutions plus the number of digit deletions and insertions divided by the total number of digits.

4.1. Without Adaptation

A WER of 0.21% was observed under clean, close talking speech for the set of test data. Single microphone, array, and adaptation results are compared to this baseline. It was found that the distance recording environment increases the single microphone WER to 15.14%, a large degradation from the baseline. Furthermore, while the close-talk clean speech recognition had a sentence error rate, SER, of only 1.37%, the single distance microphone resulted in a sentence error rate of 55.31%. That is, in the single microphone distance environment, over half of the seven digit strings are recognized incorrectly. The delay-and-sum beamforming microphone array alleviates some of the recognition error. The microphone array processing reduces the WER from 15.14% to 7.22%. Adaptation of the clean speech HMMS provides further enhancement on this result, as described below.

4.2. PMC and MLLR Results

The gain of the PMC model was varied in order to determine the minimum WER for this adaptation:

<table>
<thead>
<tr>
<th>Gain, g</th>
<th>Single Mic WER %</th>
<th>Array WER %</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>12.93</td>
<td>5.00</td>
</tr>
<tr>
<td>0.05</td>
<td>11.21</td>
<td>4.38</td>
</tr>
<tr>
<td>0.1</td>
<td>10.81</td>
<td>4.24</td>
</tr>
<tr>
<td>0.25</td>
<td>10.80</td>
<td>4.65</td>
</tr>
<tr>
<td>0.5</td>
<td>11.48</td>
<td>5.37</td>
</tr>
<tr>
<td>1.0</td>
<td>13.25</td>
<td>7.14</td>
</tr>
</tbody>
</table>

PMC yielded a minimum 10.8% WER for single microphone, and 4.24% WER for the microphone array.

For MLLR adaptation, several experiments are carried to determine the best adaptation parameters. The minimum number of vectors per class, the number of MLLR adaptation iterations, and the number of utterances used are varied.

To determine the optimal granularity of the adaptation, the number of training vectors per class ranged from 25 to 400:
The results show that the adaptation parameters should have enough degree of freedom to cover the distortion but too much freedom will cause the system to be specialized only on the adaptation data. The table below shows the WER as function of the number of MLLR iterations.

<table>
<thead>
<tr>
<th>MLLR Iterations</th>
<th>Iterations</th>
<th>Single Mic WER %</th>
<th>Array WER %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>8.30</td>
<td>3.22</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>7.05</td>
<td>2.98</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>6.81</td>
<td>2.99</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>6.84</td>
<td>2.92</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>7.12</td>
<td>2.97</td>
</tr>
</tbody>
</table>

From this data, it was determined that two iterations are sufficient for MLLR adaptation to converge on the adaptation data. Three iterations minimized single microphone WER.

In order to determine the effect of the number of the adaptation utterance on the performance, two seven digit strings were randomly selected per speaker for an equal ratio of male to female. This is to determine the minimum amount of training data that can be collected and still yield a low WER.

<table>
<thead>
<tr>
<th>MLLR Training Utterances</th>
<th># Training Utterances</th>
<th>Single Mic WER %</th>
<th>Array WER %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>220</td>
<td>7.05</td>
<td>2.98</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>7.76</td>
<td>3.01</td>
</tr>
<tr>
<td></td>
<td>75</td>
<td>8.82</td>
<td>3.40</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>9.19</td>
<td>4.04</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>10.01</td>
<td>4.78</td>
</tr>
</tbody>
</table>

These results show that the MLLR, for distant array data, can be successfully trained on as few as 100 utterances, and achieve a WER of 3.01%, similar to 2.98%.

### 5. CONCLUSIONS

The best WER results were found with MLLR adaptation and using the microphone array, at 2.98%. The following tables summarizes the best results for single and array microphones, with and without adaptation:

<table>
<thead>
<tr>
<th>Compiled Recognition Results</th>
<th>WER (%)</th>
<th>Single Mic</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>0.21%</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>No Adapt</td>
<td>15.14%</td>
<td>7.22%</td>
<td></td>
</tr>
<tr>
<td>PMC</td>
<td>10.81%</td>
<td>4.24%</td>
<td></td>
</tr>
<tr>
<td>MLLR</td>
<td>7.05%</td>
<td>2.98%</td>
<td></td>
</tr>
</tbody>
</table>

Delay-and-sum beamforming was found to provide a significant improvement of recognition accuracy over that of a single microphone. In addition, adapting the HMMs with MLLR and PMC provided further improvement.

Although it under-performed MLLR, PMC had the advantage that only statistics about the noise background are necessary, and not an entire set of adaptation data. The MLLR offered better performance. Variation over the MLLR showed that two iterations are sufficient for performance optimization. In addition, it was found that as few as 100 utterances was sufficient for effectively adapting the distant models. Using 100 vector per class minimum provides a set of transformations that is neither too general nor too specific.

The improvement trends using MLLR model adaptation with a microphone array match those found in previous work [2].

### REFERENCES


