Calibration of Microphone Arrays for Improved Speech Recognition

Michael L. Seltzer[1] and Bhiksha Raj[2]
1. Department of Electrical and Computer Engineering, Carnegie Mellon University
Pittsburgh, Pennsylvania 15213 USA
2. Compaq Computer Corporation
Cambridge, MA 02142 USA

Abstract

We present a new microphone array calibration algorithm specifically designed for speech recognition. Currently, microphone-array-based speech recognition is performed in two independent stages: array processing, and then recognition. Array processing algorithms designed for speech enhancement are used to process the waveforms before recognition. These systems make the assumption that the best array processing methods will result in the best recognition performance. However, recognition systems interpret a set of features extracted from the speech waveform, not the waveform itself. In our calibration method, the filter parameters of a filter-and-sum array processing scheme are optimized to maximize the likelihood of the recognition features extracted from the resulting output signal. By incorporating the speech recognition system into the design of the array processing algorithm, we are able to achieve improvements in word error rate of up to 37% over conventional array processing methods on both simulated and actual microphone array data.

1. Introduction

State-of-the-art speech recognition systems are known to perform reasonably well when the speech signals are captured in noise-free environments using close-talking microphones worn near the speaker's mouth. However, such ideal acoustic conditions are usually unrealistic. The real-world environment is often noisy, and the speaker is normally not wearing a close-talking microphone. In such environments, as the distance between the speaker and the microphone increases, the recorded signal becomes increasingly susceptible to background noise and reverberation effects that significantly degrade speech recognition performance. This is an especially vexing problem in situations where the location of the microphone and/or the user are dictated by physical constraints of the operating environment, as in meeting rooms or automobiles.

It has long been known that this problem can be greatly alleviated by the use of multiple microphones to capture the speech signal. Microphone arrays record the speech signal simultaneously over a number of spatially separated channels. Many techniques have been developed to combine the signals in the array to achieve a substantial improvement in the signal-to-noise ratio (SNR) of the final output signal.

The most common array processing method is delay-and-sum beamforming [1]. Signals from the various microphones are first time-aligned to adjust for the delays caused by path length differences between the speech source and each of the microphones, and then the aligned signals are averaged. Any interfering noise signals from sources that are not exactly coincident with the speech source remain misaligned and thus are attenuated by the averaging. It can be shown that if the noise signals corrupting each microphone channel are uncorrelated to each other and the target speech signal, delay-and-sum processing results in a 3 dB increase in the SNR of the output signal for every doubling of the number of microphones in the array [1].

Most other array-processing procedures are variations of this basic delay-and-sum scheme or its natural extension, filter-and-sum processing, where each microphone channel has an associated filter, and the captured signals are first filtered before they are combined. Nordholm *et al.* design adaptive filters for each of the microphones in the array based on stored calibration examples of speech and noise [2]. In [3], Marro *et al.* apply a post filter that filters the combined signal from the microphones in order to increase the SNR of the resulting signal. Several other similar microphone array processing methods have been proposed in the literature.

While these methods can effectively improve the SNR of the captured speech signal, they suffer from the drawback that they are all inherently speech *enhancement* schemes, aimed at improving the quality of the speech waveform as judged perceptually by human listeners or quantitatively by SNR. While this is certainly appropriate if the speech signal is to be interpreted by a human listener, it may not necessarily be the right criteria if the signal is to be interpreted by a speech recognition system. Speech recognition systems interpret not the waveform itself, but a set of *features* derived from the speech waveform through a series of transformations. By ignoring the manner in which the recognition system processes incoming signals, these speech enhancement algorithms are treating speech recognition systems as equivalent to human listeners, which is not the case.

As a result, while more complex array-processing algorithms can significantly outperform simple delay-and-sum processing from a speech enhancement point of view, many of these improvements do not translate into substantial gains in speech recognition performance.

In this paper we propose a new filter-and-sum microphone array processing scheme that *integrates the speech recognition system directly into the filter design process*. In our scheme, as in previous methods, the array calibration process involves the design of a set of finite impulse response (FIR) filters, one for each microphone in the array. However, unlike all previous methods, our algorithm calibrates these filters specifically for optimal speech recognition performance, without regard to SNR or perceived "listenability". More precisely, filter parameters are learned which maximize the likelihood of the recognition features derived from the final output signal, as measured

by the recognition system itself. Incorporating the speech recognition system into the filter design strategy ensures that the filters enhance those components of the speech signal that are important for recognition, without undue emphasis on the unimportant components. Experiments indicate that recognition accuracies obtained with signals derived using the proposed method are significantly higher than those obtained using conventional array processing techniques.

The remainder of this paper describes the proposed method and experimental results showing its efficacy. In Section 2 we review the filter-and-sum array processing scheme used in this work. In Section 3 the proposed filter optimization method is described in detail. In Section 4 we present experimental results using the proposed method, and finally in Section 5 we present our conclusions and proposals for future work.

2. Filter-and-sum array-processing

We employ traditional filter-and-sum processing to combine the signals captured by the array. In the first step the speech source is localized and the relative channel delays caused by path length differences to the source are resolved so that all waveforms captured by the individual microphones are aligned with respect to each other. Several algorithms have been proposed in the literature to do this, *e.g.* [4], and any of them can be applied here. In our work we have employed simple cross-correlation to determine the delays among the multiple channels.

Once the signals are time aligned, each of the signals is passed through an FIR filter whose parameters are determined by the calibration scheme described in the following section. The filtered signals are then added to obtain the final signal. This procedure can be mathematically represented as follows:

$$y[n] = \sum_{i=1}^{N} \sum_{k=0}^{K} h_i[k] x_i[n-k-\tau_i] \quad (1)$$

where $x_i[n]$ represents the n^{th} sample of the signal recorded by the i^{th} microphone, τ_i represents the delay introduced into the i^{th} channel to time align it with the other channels, $h_i[k]$ represents the k^{th} coefficient of the FIR filter applied to the signal captured by the i^{th} microphone, and $y[n]$ represents the n^{th} sample of the final output signal. K is the order of the FIR filters and N is the total number of microphones in the array.

Once $y[n]$ is obtained, it can be parameterized to derive a sequence of feature vectors to be used for recognition.

3. Filter Calibration

As stated in Section 1, we wish to choose the filter parameters $h_i[k]$ that will optimize speech recognition performance. One way to do this is to maximize the likelihood of the *correct* transcription for the utterance, thereby increasing the difference between its likelihood and that of other competing hypotheses. However, because the correct transcription of any utterance is unknown, we optimize the filters based on a single *calibration utterance* with a known transcription. Before using the speech recognition system, a user records a calibration utterance, and the filter parameters are optimized based on this utterance. All subsequent utterances are processed using the derived filters in the filter-and-sum scheme described in the previous section.

The sequence of recognition features derived from any utterance $y[n]$ is a function of the filter parameters $h_i[n]$ of all of the microphones, as in (1). In this paper recognition features are assumed to be mel-frequency cepstra; however, the filter optimization algorithm presented here should be applicable to any choice of recognition features with appropriate modification to the arithmetic. The sequence of mel-frequency cepstral coefficients is computed by segmenting the utterance into overlapping frames of speech and deriving a mel-frequency cepstral vector for each frame. If we let $\boldsymbol{h}$ represent the vector of all filter parameters $h_i[k]$ for all microphones, and $\boldsymbol{y}_j(\boldsymbol{h})$ the vector of observations of the j^{th} frame expressed as a function of these filter parameters, the mel-frequency cepstral vector for a frame of speech can be expressed as

$$\boldsymbol{z}_j = DCT(\log(\boldsymbol{M}\left|DFT(\boldsymbol{y}_j(\boldsymbol{h}))\right|^2)) \quad (2)$$

where $\boldsymbol{z}_j$ represents the mel-frequency cepstral vector for the j^{th} frame of speech and $\boldsymbol{M}$ represents the matrix of the weighting coefficients of the mel filters.

The likelihood of the correct transcription must be computed using the statistical models employed by the recognition system. In this paper we assume that the speech recognition system is a Hidden Markov Model (HMM) based system. We further assume, for simplicity, that the likelihood of the utterance is largely represented by the likelihood of the most likely state sequence through the HMMs. Under this assumption, the log-likelihood of the utterance can be represented as

$$L(\boldsymbol{Z}) = \sum_{j=1}^{T} \log(P(\boldsymbol{z}_j|s_j)) + \log(P(s_1, s_2, s_3, \ldots, s_T)) \quad (3)$$

where $\boldsymbol{Z}$ represents the set of all feature vectors $\{\boldsymbol{z}_j\}$ for the utterance, T is the total number of feature vectors (frames) in the utterance, s_j represents the j^{th} state in the most likely state sequence and $\log(P(\boldsymbol{z}_j|s_j))$ is the log likelihood of the observation vector $\boldsymbol{z}_j$ computed on the state distribution of s_j. The *a priori* log probability of the most likely state sequence, $\log(P(s_1, s_2, s_3, \ldots, s_T))$, is determined by the transition probabilities of the HMMs. In order to maximize the likelihood of the correct transcription, $L(\boldsymbol{Z})$ must be jointly optimized with respect to both the filter parameter vector $\boldsymbol{h}$ and the state sequence $s_1, s_2, s_3, \ldots, s_T$. This can be done by alternately optimizing the state sequence and $\boldsymbol{h}$.

For a given $\boldsymbol{h}$, the most likely state sequence can be easily determined using the Viterbi algorithm. However, for a given state sequence, in the most general case, $L(\boldsymbol{Z})$ cannot be directly maximized with respect to $\boldsymbol{h}$ for two reasons. First, the state distributions used in most HMMs are complex distributions, *i.e.* mixtures of Gaussians. Second, $L(\boldsymbol{Z})$ and $\boldsymbol{h}$ are related through many levels of indirection, as can be seen from (1), (2), and (3). As a result, iterative non-linear optimization methods must be used to solve for $\boldsymbol{h}$. Computationally, this can be highly expensive. In this paper we make a few additional approximations that reduce the complexity of the problem. We assume that the state distributions of the various states of the HMMs are modelled by single Gaussians. Furthermore, we assume that to maximize the likelihood of a vector on a Gaussian, it is sufficient to minimize the Euclidean distance between the observation vector and mean vector of the Gaussian. Thus, given the optimal state sequence, we can define an objective

function to be minimized with respect to $\boldsymbol{h}$ as follows:

$$Q(\boldsymbol{Z}) = \sum_{j=1}^{T} \left\| \boldsymbol{z}_j - \mu_{s_j} \right\|^2 \quad (4)$$

where μ_{s_j} is the mean vector of the Gaussian distribution of the state s_j. Because the dynamic range of mel-frequency cepstra diminishes with increasing cepstral order, the low order cepstral terms have a much more significant impact on the objective function in (4) than the higher ones. To avoid this potential problem, we define the objective function in the log Mel spectral domain, rather than the cepstral domain:

$$Q(\boldsymbol{Z}) = \sum_{j=1}^{T} \left\| IDCT(\boldsymbol{z}_j - \mu_{s_j}) \right\|^2 \quad (5)$$

Using (1), (2), and (5), the gradient of the objective function with respect to $\boldsymbol{h}$, $\nabla_{\boldsymbol{h}} Q(\boldsymbol{Z})$, can be determined. Using the objective function and its gradient, we can minimize (5) using the conjugate gradient method [5] to obtain the optimal filter parameters $\boldsymbol{h}$.

Thus, the entire algorithm for estimating the filter parameters for an array of *N* microphones using the calibration utterance is as follows:

1. Determine the array path length delays τ_i and time-align the signals from each the *N* microphones.
2. Initialize the filter parameters: $h_i[0] = 1/N$; $h_i[k]=0,\ k \neq 0$
3. Process the signals using (1) and derive recognition features
4. Determine the optimal state sequence from the obtained recognition features.
5. Use the obtained state sequence and (5) to estimate optimal filter parameters.
6. If the value of the objective function using the estimated filter parameters has not converged, go to Step 3.

An alternative to estimating the state sequence and filter parameters iteratively is to record the calibration utterance simultaneously through a close-talking microphone. The recognition features derived from this clean speech signal can either be a) used to determine the optimal state sequence, or b) used directly in (5) instead of the Gaussian mean vectors. However, even in the more realistic situation where no close-talking microphone is used, a single pass through Steps 1 through 6 is sufficient to estimate the filter parameters. The estimated filter parameters are then used to process all subsequent signals in the filter-and-sum manner described in Section 2.

4. Experimental results

Experiments were performed using two different databases to evaluate the proposed algorithm, one using simulated microphone array speech data and one with actual microphone array data.

A simulated microphone array test set, "WSJ_SIM", was designed using the test set of the Wall Street Journal (WSJ0) corpus [6]. Room simulation impulse response filters were designed for a room 4m x 5m x 3m with a reverberation time of 200ms. The microphone array configuration consisted of 8 microphones placed around an imaginary 0.5m x 0.3m flat panel display on one of the 4m walls. The speech source was placed 1 meter from the array at the same height as the center of the array, as if a user were addressing the display. A noise source was placed above, behind, and to the left of the speech source. A room impulse response filter was created for each source/microphone pair. To create a noise-corrupted microphone array test set, clean WSJ0 test data were passed through each of the 8 speech source room impulse response filters and white noise was passed through each of the 8 noise source filters. The filtered speech and noise signals for each microphone location were then added together. The test set consisted of 8 speakers with 80 utterances per speaker. Test sets were created with SNRs from 0-25 dB. The original WSJ0 test data served as a close-talking control test set.

The real microphone array data set, "CMU_TMS", was collected at CMU [7]. The array used in this data set was a horizontal linear array of 8 microphones spaced 7cm apart placed on a desk in a noisy speech lab approximately 5m x 5m x 3m. The talkers were seated directly in front of the array at a distance of 1 meter. There are 10 speakers each with 14 unique utterances comprised of alphanumeric strings and strings of command words. Each array recording has a close-talking microphone control recording for reference.

All experiments were performed using a single pass through Steps 1-6 in the calibration algorithm described in the previous section. In all experiments, the first utterance of each data set was used as the calibration utterance. After the microphone array filters were calibrated, all test utterances were processing using the filter-and-sum method described in Section 2. Speech recognition was performed using the SPHINX-III speech recognition system with context-dependent continuous HMMs (8 Gaussian/state) trained on clean speech using 7000 utterances from the WSJ0 training set.

In the first series of experiments, the calibration procedure was performed on the WSJ_SIM test set with an SNR of 5 dB and the CMU_TMS test set. In the first experiment, the close-talking recording of the utterance was used for calibration. The stream of target feature vectors was derived from the close-talking recording and used in Equation (5) to estimate a 50-point filter for each of the microphone channels.

In the second experiment, the HMM state segmentation derived from the close-talking calibration recording was used to estimate the filter parameters. The calibration recording used in the previous experiment was force-aligned to the known transcription to generate an HMM state segmentation. The mean vectors of 1 Gaussian/state HMMs in the state sequence were used to estimate a 50-point filter for each microphone channel.

Finally, we assumed that no close-talking recording of the calibration utterance was available. Delay-and-sum processing was performed on the time-aligned microphone channels and the resulting output was used with the known transcription to generate an estimated state segmentation. The Gaussian mean vectors of the HMMs in this estimated state sequence were extracted and used to estimate 50-point filters as in the previous experiment. The word error rates (WER) from all three experiments are shown in Table 1. The results using conventional delay-and-sum beamforming are shown for comparison. Large improvements over conventional beamforming schemes are seen in all cases. Having a close-talking recording of the calibration utterance is clearly beneficial, yet significant

Array Processing Method	WSJ_SIM	CMU_TMS
Close-talking mic (CLSTK)	16.52	19.36
Single mic array channel	93.84	62.32
Delay and Sum (DS)	64.48	39.36
Calibrate Optimal Filters w/ CLSTK Cepstra	33.37	35.0
Calibrate Optimal Filters w/ CLSTK State Segmentations	36.5	37.07
Calibrate Optimal Filters w/ DS State Segmentations	40.2	34.95

Table 1: Word error rate for the two microphone array test corpora, WSJ_SIM at 5 dB SNR, and CMU_TMS, using conventional delay and sum processing and the optimal filter calibration methods

improvements in word error rate can be seen even when no close-talking recording is used.

Figure 1 shows WER as a function of SNR for the WSJ_SIM data set, using the described calibration scheme and for comparison, conventional delay-and-sum processing. For all SNRs, no close-talking recordings were used. All target feature vector sequences were estimated from state segmentations generated from the delay-and-sum output of the array.

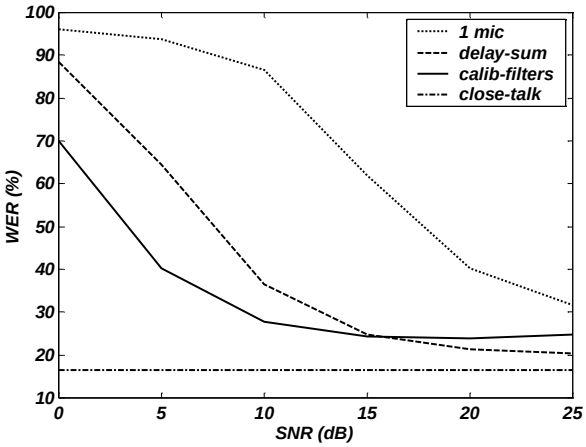

Figure 1. Word error rate vs. SNR for the WSJ_SIM test set using filters calibrated from delay-and-sum state segmentations.

Clearly, at low to moderate SNRs, there are significant gains over conventional delay-and-sum beamforming. However, at high SNRs, the performance of the calibration technique drops below that of delay-and-sum processing. We believe that this the result of using the mean vectors from the 1 Gaussian/state HMMs as the target feature vectors. In doing so, we are effectively quantizing our feature space, and forcing the data to fit single Gaussian HMMs rather than the Gaussian mixtures which are known to result in better recognition accuracy.

To demonstrate the advantage of estimating the filter parameters of each microphone channel jointly, rather than independently, a final experiment was conducted. The recognition performance using jointly optimized filters was compared to two other strategies: 1) performing delay-and-sum and then optimizing a single filter for the resulting output signal, and 2) optimizing the filters for each channel independently. These optimization variations were performed on the WSJ_SIM test set with an SNR of 10 dB. Again, 50-point filters were designed in all cases. The results are shown in Table 2.

It is clear that joint optimization of the filters is superior to either of the other two optimization methods.

Filter Optimization Method	WSJ_SIM
Delay and Sum	36.43
Optimize Single Filter for D & S output	36.29
Optimize Mic Array Filters Independently	48.19
Optimize Mic Array Filters Jointly	27.79

Table 2: Word error rate for the WSJ_SIM test set with an SNR of 10dB for delay-and-sum processing and three different filter optimization methods.

5. Summary

In this paper, we have presented a new calibration scheme for microphone arrays specifically targeted at speech recognition performance. By incorporating the speech recognition system itself into the calibration algorithm, we have been able to design an array processing strategy that ensures that signal components important for recognition are emphasized, without undue emphasis on less important signal components, SNR or other speech enhancement metrics. In doing so, we achieved relative improvements of up to 37% in WER over conventional delay-and-sum processing. Because of the relatively short filter lengths used in these experiments, it is apparent that the estimated calibration filters were performing noise reduction only, and not dereverberation. We plan to try to calibrate significantly longer filters in order to attenuate the effects of both noise and reverberation on the speech recognition feature vectors.

Acknowledgements

The authors thank Professor Michael Brandstein of Harvard University for providing us with the room simulation filters, and the members of the speech group at Compaq Cambridge Research Labs.

References

[1] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques*. New Jersey: Prentice Hall, 1993.

[2] S. Nordholm, I. Clasesson, and M. Dahl, "Adaptive microphone array employing calibration signals: an analytical evaluation," *IEEE Trans. on Speech and Audio Proc*., vol. 7, pp. 241-252, May 1999.

[3] C. Marro, Y. Mahieux, and K. U. Simmer, "Analysis of noise reduction and dereverberation techniques based on microphone arrays with post filtering," *IEEE Trans. on Speech and Audio Proc.*, vol. 6, pp. 240-259, May 1998.

[4] M. S. Brandstein and H. F. Silverman, "A practical methodology for speech source localization with microphone arrays," *Computer Speech and Language*, vol. 11, pp. 91-126, April 1997.

[5] E. Polak, *Computational methods in Optimization*, New York: Academic Press, 1971.

[6] D. Paul and J. Baker, "The design of the Wall Street Journal-based CSR corpus", *Proc. DARPA Speech and Natural Language Workshop*, Harriman, New York, pp. 357-362, Feb. 1992.

[7] T. M. Sullivan, *Multi-microphone correlation-based processing for robust automatic speech recognition*, Ph.D. dissertation, Carnegie Mellon University, August, 1996.