Speech-Recognizer-Based Filter Optimization for Microphone Array Processing

Michael L. Seltzer, *Student Member, IEEE,* and Bhiksha Raj

***Abstract*—Conventional microphone-array processing schemes used for speech recognition enhance the output waveform using optimization criteria that are independent of the recognition system. We present a new filter-and-sum array processing algorithm in which the filter parameters are calibrated to maximize recognizer likelihoods. The proposed method provides significant improvement in recognition accuracy over conventional methods.**

***Index Terms*—Beamforming, microphone array processing, robust speech recognition.**

I. Introduction

MICROPHONE-ARRAY-BASED signal processing schemes provide an effective means of compensating for the effects of noise and reverberation on the performance of speech recognition systems deployed in hands-free environments [1]. These algorithms process and combine the multiple signals captured by the array to derive an enhanced output signal that is then used for recognition, e.g., [2]. Typically, such methods are speech *enhancement* algorithms that aim to improve the signal-to-noise ratio (SNR) or perceptual quality of the output waveform [3]. As such, the optimization criteria used by these algorithms bear no direct relation to those used by speech recognition systems to determine the words spoken in an utterance.

In this letter, we present a new filter-and-sum array processing algorithm that integrates the speech recognition system into the filter design process. Filter parameters are chosen to maximize the likelihood of the processed signal as measured by the recognizer, rather than its SNR or perceptual quality. This ensures that the designed filters enhance signal components that are important for recognition, without undue emphasis on unimportant components. We describe the proposed algorithm in Section II of this letter. Experiments reported in Section III show that the proposed method results in significantly better recognition performance than that achieved with conventional delay-and-sum processing.

Manuscript received January 28, 2002; revised August 26, 2002. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Steven Kay.

M. L. Seltzer is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: mseltzer@cs.cmu.edu).

B. Raj is with the Mitsubishi Electric Research Labs, Cambridge, MA 02139 USA (e-mail: bhiksha@merl.com).

Digital Object Identifier 10.1109/LSP.2002.807877

II. Filter Calibration

Filter-and-sum microphone-array processing can be notationally expressed as

$$y[n] = \sum_{i=1}^{N} h_i[n] \otimes x_i[n - \tau_i] \tag{1}$$

where $x_i[n]$ represents the signal recorded by the ith microphone; τ_i represents the delay introduced into the ith channel to time-align it with the other channels; $h_i[n]$ represents the finite impulse response (FIR) filter applied to the signal captured by the ith microphone; $\otimes$ is the convolution operator; and $y[n]$ represents the output signal. N is the total number of microphones in the array.

The goal of the proposed algorithm is to estimate the filter parameters $h_i[n]$ that optimize the speech recognition performance obtained with $y[n]$. We can do this by maximizing the likelihood of the *correct* transcription for the utterance, computed using the statistical models of the recognizer. However, since the correct transcriptions used by utterances to be recognized are unknown, we optimize the filters based on a *calibration utterance* with known transcription, which the user records prior to using the system.

We pose the optimization problem in the context of hidden Markov model (HMM)-based speech recognition systems that operate on frame-based parameterizations of the speech signal. In this letter, we assume that each frame of speech is parameterized as a vector of Mel-frequency cepstral coefficients (MFCC); however, the approach taken is equally applicable to any other type of feature vector. Let $\boldsymbol{h}$ represent a vector composed of all filter parameters for all microphones. Let $\boldsymbol{y}_j(\boldsymbol{h})$ represent the signal $y[n]$ in the jth frame of the calibration utterance, expressed as a function of $\boldsymbol{h}$. The MFCC vector for the frame, $\boldsymbol{z}_j(\boldsymbol{h})$, is computed as

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

where $\boldsymbol{M}$ represents the matrix of weighting coefficients of the Mel filters. The entire utterance is parameterized into the sequence of vectors $\boldsymbol{z}_1(\boldsymbol{h}), \boldsymbol{z}_2(\boldsymbol{h}), \ldots, \boldsymbol{z}_T(\boldsymbol{h})$ that we represent as $\boldsymbol{Z}(\boldsymbol{h})$.

In an HMM-based system, the likelihood of any data sequence is largely represented by the likelihood of the most likely state sequence through the HMMs. The log-likelihood of $\boldsymbol{Z}(\boldsymbol{h})$ can therefore be approximated as

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

where $s_1, s_2, s_3, \ldots, s_T$ represents the most likely state sequence. $\boldsymbol{P}(\boldsymbol{z}_j(\boldsymbol{h})|s_j)$ represents the probability of $\boldsymbol{z}_j(\boldsymbol{h})$ computed on the distribution of the jth state s_j in this sequence. $\boldsymbol{P}(s_1, s_2, s_3, \ldots, s_T)$ is determined by the state transition probabilities of the HMM.

Optimization of $\boldsymbol{L}(\boldsymbol{Z}(\boldsymbol{h}))$ requires joint estimation of both $\boldsymbol{h}$ and the most likely state sequence $s_1, s_2, s_3, \ldots, s_T$. This can be performed by iteratively estimating the optimal state sequence for a given $\boldsymbol{h}$ using the Viterbi algorithm, and optimizing $\sum_j \log(\boldsymbol{P}(\boldsymbol{z}_j(\boldsymbol{h})|s_j))$ with respect to $\boldsymbol{h}$ for that state sequence. However, $\sum_j \log(\boldsymbol{P}(\boldsymbol{z}_j(\boldsymbol{h})|s_j))$ cannot be directly optimized, and computationally expensive hill-climbing methods must be used to solve for $\boldsymbol{h}$. To reduce computational effort, we model state output distributions as Gaussians, and we assume that to maximize $\boldsymbol{P}(\boldsymbol{z}_j(\boldsymbol{h})|s_j))$ it is sufficient to minimize the weighted distance $(\boldsymbol{z}_j(\boldsymbol{h})-\mu_{s_j})^T\boldsymbol{W}(\boldsymbol{z}_j(\boldsymbol{h})-\mu_{s_j})$ between $\boldsymbol{z}_j(\boldsymbol{h})$ and μ_{s_j}, the mean of the output distribution of s_j. Specifically, we assume that $\boldsymbol{W} = (\mathrm{IDCT})^T(\mathrm{IDCT})$, where IDCT is the inverse discrete cosine transform matrix. This effectively transforms the maximization of $\boldsymbol{P}(\boldsymbol{z}_j(\boldsymbol{h})|s_j)$ into the minimization of the Euclidean distance between two log-spectral vectors. Under these assumptions, maximization of $\sum_j \log(\boldsymbol{P}(\boldsymbol{z}_j(\boldsymbol{h})|s_j))$ is equivalent to minimization of the objective function

$$Q(\boldsymbol{h}) = \sum_{j=1}^{T} ||\mathrm{IDCT}(\boldsymbol{z}_j(\boldsymbol{h}) - \mu_{s_j})||^2. \tag{4}$$

$Q(\boldsymbol{h})$ can be optimized with respect to $\boldsymbol{h}$ using hill-climbing methods such as the conjugate gradient method [4].

The entire algorithm for optimizing $\boldsymbol{h}$ from a calibration utterance is thus as follows.

1) Time-align the signals from the N microphones.
2) Initialize $\boldsymbol{h}$ as $h_i[0] = 1/N$; $h_i[k] = 0, k \neq 0$.
3) Process signals using $\boldsymbol{h}$ and derive recognition features.
4) Determine optimal state sequence from derived recognition features.
5) Use optimal state sequence and (4) to estimate $\boldsymbol{h}$.
6) If $Q(\boldsymbol{h})$ has not converged, go to step 3).

The estimated $\boldsymbol{h}$ is used to process all future utterances during recognition. If the calibration utterance is recorded simultaneously over a close-talking microphone, features derived from this cleaner signal can be used either to determine the optimal state sequence in step 4), or directly in (4) instead of the Gaussian mean vectors.

III. Experiment Results

Experiments were conducted on two databases. The first database (CMU_TMS) was recorded in a noisy laboratory using a linear array of eight microphones spaced 7 cm apart. Talkers were seated 1 m from the center of the array. Ten speakers recorded 14 utterances each, consisting of alphanumeric and command-word strings. A close-talking microphone recording of each utterance was captured simultaneously for reference.

The second database (WSJ_SIM) is a simulated microphone array test set derived from the Wall Street Journal (WSJ0) test corpus. A 4 m × 5 m × 3 m room with eight microphones around a 0.5 m × 0.3 m flat-panel display on one of the walls was simulated using the image method [5]. The speech source was located 1 m from the center of the array, and a white noise source was placed above, behind, and to the left of the speech source. Recordings were created at a range of SNRs. Both the CMU_TMS data and the WSJ_SIM data were digitized at a sampling rate of 16 kHz.

TABLE I
Percentage WER on the CMU_TMS and the 5-dB WSJ_SIM Data, Using Conventional Delay-and-Sum Processing and the Proposed Filter Calibration Methods. The WERs on Close-Talking Microphone Recordings of the Test Data and on Signals Recorded by One of the Microphones are Also Shown

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 filters using CLSTK cepstra instead of Gaussian means in (4)	33.37	35.0
Calibrate filters using CLSTK cepstra to derive state segmentations	36.5	37.07
Calibrate filters using only multimic adaptation data	40.2	34.95

The SPHINX-III continuous-density HMM-based speech recognition system was used in all experiments. Five thousand tied states, each modeled by a single Gaussian, were trained with 7000 utterances from the WSJ0 training set. Cepstral mean normalization was performed on all training utterances. In addition, a second mean vector was computed for each state from unnormalized cepstra. These unnormalized mean vectors were used for filter optimization, since the objective function used in the filter design does not account for mean normalization.

In all experiments, 50-point FIR filters were optimized for each microphone using a single calibration utterance and applied to the entire test set. Only one iteration of steps 1)–6) of the algorithm was performed. Filter calibration using features derived from the close-talking microphone recording of the calibration utterance was also evaluated.

Table I shows recognition word error rates (WERs) for the CMU_TMS and 5-dB SNR WSJ_SIM data. WERs with conventional delay-and-sum processing and the proposed algorithm are both shown. For reference, WERs on signals from a close-talking microphone and a single microphone from the array are also shown. Fig. 1 shows WERs obtained on the WSJ_SIM corpus as a function of the SNR of the test data. For this experiment, the close-talking recording was not used for calibration. Table I and Fig. 1 show that the proposed algorithm outperforms conventional delay-and-sum processing significantly in most situations. At high SNRs (above 15 dB), delay-and-sum processing is somewhat better than the proposed method, possibly due to the various approximations introduced in the algorithm for computational efficiency.

The proposed algorithm jointly optimizes all filters in the array. Obvious alternatives are to optimize all filters independently, or to optimize a single filter that operates on the output

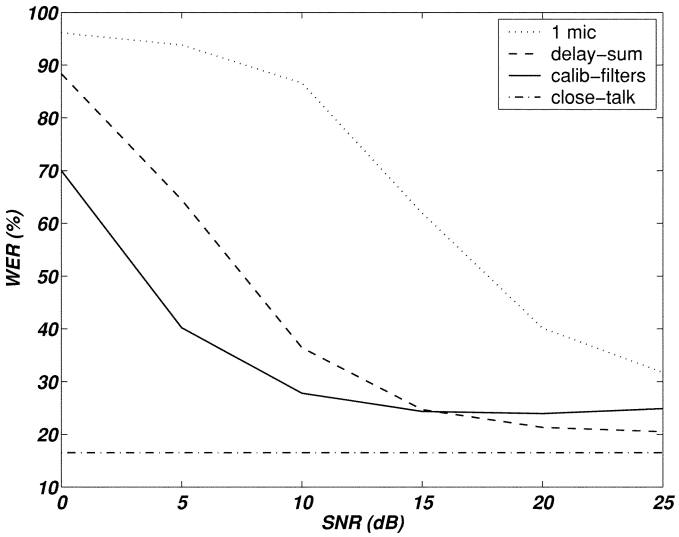

Fig. 1. Word error rate versus SNR for the WSJ_SIM test set using filters calibrated with the proposed algorithm.

of a conventional delay-and-sum beamformer. Table II, which compares these alternatives, shows that joint optimization results in the best performance. Finally, it should be noted that only a single pass through the calibration algorithm was used in these experiments. Further iterations did not improve recognition performance.

IV. Discussion

In this letter, we have described a new microphone array processing calibration algorithm that jointly optimizes the parameters of a filter-and-sum array processor based on information from a speech recognizer. Speech signals processed using the proposed algorithm are more accurately recognized by speech recognition systems than those processed with conventional delay-and-sum beamforming. The algorithm also outperforms delay-and-sum beamforming followed by postfiltering, showing that the optimized filters produce both spatial and spectral filtering.

The filters can, in principle, also improve the alignment of the microphone recordings by appropriate selection of filter taps. However, this is not the primary effect of the filters. In the experiments reported on the WSJ_SIM data, the signals were hand-aligned using the perfectly known array geometry and talker location. Nevertheless, the proposed algorithm resulted in significantly better performance than delay-and-sum processing. On this dataset, the results obtained with hand-aligned and automatically aligned signals were very similar.

TABLE II
WER for the WSJ_SIM Test set With an SNR of 10 dB for Delay-and-Sum Processing and Three Different Filter Optimization Methods

Filter Optimization Method	WER(%)
Delay and sum (D & S)	36.43
Optimize single filter applied to D & S output	36.29
Optimize mic array filters independently	48.19
Optimize mic array filters jointly	27.79

In our experiments, we used only a single utterance to calibrate the array. Since the calibration data were limited, only relatively short filters were optimized. We expect that as the amount of calibration data is increased, longer filters can be trained to further improve performance, especially in highly reverberant conditions. However, minor variations in filter order were not observed to change performance significantly.

A final point is that feature mean normalization, a standard preprocessing procedure used in recognition systems, was not performed during filter optimization, although it was used during recognition. Since mean normalization is known to result in better recognition, it is likely that incorporating it into the filter design may improve the performance of the proposed algorithm further.

References

[1] M. Omologo, "Hands-free speech recognition: current activities and future trends," in *Proc. Int. Workshop Hands-Free Speech Communication*, Kyoto, Japan, Apr. 2001.

[2] T. B. Hughes, S. S. Kim, J. H. DiBiase, and H. F. Silverman, "Performance of an HMM speech recognizer using a real-time tracking microphone array as input," *IEEE Trans. Speech Audio Processing*, vol. 7, pp. 346–349, May 1999.

[3] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[4] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York: Springer-Verlag, 1999.

[5] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, pp. 943–950, 1979.