MULTI-CHANNEL SOURCE SEPARATION BY FACTORIAL HMMS

Manuel J. Reyes-Gomez[1]*, Bhiksha Raj*[2]*, Daniel P. W. Ellis*[1]

[1]Dept. of Electrical Engineering, Columbia University
[2]Mitsubishi Electric Research Laboratories

ABSTRACT

In this paper we present a new speaker-separation algorithm for separating signals with known statistical characteristics from mixed multi-channel recordings. Speaker separation has conventionally been treated as a problem of Blind Source Separation (BSS). This approach does not utilize any knowledge of the statistical characteristics of the signals to be separated, relying mainly on the independence between the various signals to separate them. The algorithm presented in this paper, on the other hand, utilizes detailed statistical information about the signals to be separated, represented in the form of hidden Markov models (HMM). We treat the signal separation problem as one of beam-forming, where each signal is extracted using a filter-and-sum array. The filters are estimated to maximize the likelihood of the summed output, measured on the HMM for the desired signal. This is done by iteratively estimating the best state sequence through the HMM from a factorial HMM (FHMM), that is the cross-product of the HMMs for the multiple signals, using the current output of the array, and estimating the filters to maximize the likelihood of that state sequence. Experiments show that the proposed method can cleanly extract a background speaker who is 20dB below the foreground speaker in a two-speaker mixture, when the HMMs for the signals are constructed from knowledge of the utterance transcriptions.

1. INTRODUCTION

There are several situations where two or more speakers speak simultaneously, and it is necessary to be able to separate the speech from the individual speakers from recordings of the simultaneous speech. Conventionally, this is referred to as the *speaker-separation* or *source-separation* problem. One approach to this problem is through the use of a time-varying filter on single-channel recordings of speech simultaneously spoken by two or more speakers [1, 2]. This approach uses extensive prior information about the statistical nature of speech from the different speakers, usually represented by dynamic models like the hidden Markov model (HMM), to compute the time-varying filters. A second, more popular approach to speaker separation is through the use of signals recorded using multiple microphones. The algorithms involved typically require at least as many microphones as the number of signal sources. The problem of speaker separation is then treated as one of *Blind* Source Separation (BSS), which is performed using standard techniques like Independent Component Analysis (ICA). In this approach, no *a priori* knowledge of the signals is assumed. Instead, the component signals are estimated as a weighted combination of current and past samples from the multiple recordings of the mixed signals. The weights are estimated to optimize an objective function that measures the independence of the estimated component signals [3].

Both of these approaches, however, have drawbacks. The time-varying filter approach, that uses *a priori* signal statistics, is based on single-channel recordings of the mixed signals. The amount of information present in a single recording is usually insufficient to do effective speaker separation. The blind multiple-microphone based approach, on the other, hand ignores all *a priori* information about the speakers and consequently fails in many situations, such as when the signals are recorded in a reverberant environment.

In this paper we propose a new speaker separation algorithm that does not have the drawbacks associated with either of the conventional approaches. Rather, it combines the best features of both. In the algorithm proposed, recordings from multiple microphones are combined to extract the component speech signals using the filter-and-sum method [4], described in Section 2. Statistical information about the speech from the multiple speakers is used to optimize the filters. The algorithm is thus not blind, rather, it can be viewed as beamforming that is performed using statistical information from the signals as encoded by a statistical model such as an HMM. A similar algorithm has been used earlier for speech enhancement by Seltzer *et. al.* [5]. We describe our filter estimation algorithm in Sections 3 and 4. Experiments reported in Section 5 show that the proposed algorithm is very effective at speaker separation even when the signal level of the desired speaker is very low.

In the specific implementation of the algorithm, we assume a large amount of information about the signals. Specifically, we assume that transcriptions are available for each of the speakers, and that this can be used to extract their audio signal. While this is an interesting problem in itself, the underlying algorithm is equally applicable to the more generic cases, as we explain in our conclusions in Section 6.

2. FILTER-AND-SUM MICROPHONE PROCESSING

In this section we will describe the filter-and-sum array processing to be used for developing the current algorithm for speaker separation. The only assumption we make in this context is that the number of speakers is known. For each of the speakers, a separate filter-and-sum array is designed. The signal from each microphone is filtered by a microphone-specific filter. The various filtered signals are summed to obtain the final processed signal. Thus, the output signal for speaker i, $y_i[n]$, is obtained as:

$$y_i[n] = \sum_{j=1}^{L} h_{ij}[n] * x_j[n] \qquad (1)$$

where L is the number of microphones in the array, $x_j[n]$ is the signal at the j^{th} microphone and $h_{iv}[n]$ is the filter applied to the

j^{th} filter for speaker i. The filter impulse responses $h_{ij}[n]$ must be optimized such that the resultant output $y_i[n]$ is the separated signal from the i^{th} speaker.

3. OPTIMIZING THE FILTERS FOR A SPEAKER

In the algorithm proposed, the filters for any speaker are optimized using the available information about their speech. The information used is based on the assumption that the correct transcription of the speech from the speaker whose signal is to be extracted, is known. The goal of the current implementation of the algorithm is thus transcription-based speaker separation. We further assume that we have access to a speaker-independent hidden Markov model (HMM) based speech recognition system that has been trained on a 40-dimensional Mel-spectral representation of the speech signal. The recognition system includes HMMs for the various sound units that the language comprises. From these, and the known transcription for the speaker's utterance, we first construct an HMM for the utterance. Following this, the filters for the speaker are estimated to maximize the likelihood of the sequence of 40-dimensional Mel-spectral vectors computed from the output of the filter-and-sum processed signal, on the utterance HMM.

For the purpose of optimization, we must express the Mel-spectral vectors as a function of the filter parameters as follows: We concatenate the filter parameters for the i^{th} speaker, for all channels, into a single vector $\mathbf{h}_i$. Let Z_i represent the sequence of Mel-spectral vectors computed from the output of the array for the i^{th} speaker. Let z_{it} be the t^{th} spectral vector in Z_i. z_{it} is related to $\mathbf{h}_i$ by the following equation:

$$z_{it} = log(\mathbf{M}|DFT(\mathbf{y}_{it})|^2) = log(\mathbf{M}(diag(\mathbf{F}\mathbf{X}_t\mathbf{h}_i\mathbf{h}_i^T\mathbf{X}_t^T\mathbf{F}^H))) \quad (2)$$

where $\mathbf{y}_{it}$ is a vector representing the sequence of samples from $y_i[n]$ that are used to compute z_it, $\mathbf{M}$ is the matrix of the weighting coefficients for the Mel filters, $\mathbf{F}$ is the Fourier transform matrix and $\mathbf{X}_t$ is a supermatrix formed by the channel inputs and their shifted versions.

Let Λ_i represent the set of parameters for the HMM for the utterance from the i^{th} speaker. In order to optimize the filters for the i^{th} speaker, we maximize $L_i(Z_i) = log(P(Z_i|\Lambda_i))$, the log-likelihood of Z_i on the HMM for that speaker. $L_i(Z_i)$ must be computed over all possible state sequences through the utterance HMM. However, in order to simplify the optimization, we assume that the overall likelihood of Z_i is largely represented by the likelihood of the most likely state sequence through the HMM, *i.e.*, $P(Z_i|\Lambda_i) \approx P(Z_i, \mathbf{S}_i|\Lambda_i)$, where $\mathbf{S}_i$ represents the most likely state sequence through the HMM. Under this assumption, we get

$$L_i(Z_i) = \sum_{t=1}^{T} log(P(z_{it} \mid \mathbf{s}_{it})) + log(P(\mathbf{s}_{i1}, \mathbf{s}_{i2}, .., \mathbf{s}_{iT})) \quad (3)$$

where T represents the total number of vectors in Z_i, and $\mathbf{s}_{it}$ represents the state at time t in the most likely state sequence for the i^{th} speaker.

$log(P(\mathbf{s}_{i1}, \mathbf{s}_{i2}, .., \mathbf{s}_{iT}))$ does not depend on z_{it} or the filter parameters, and therefore does not affect the optimization, hence maximizing equation 3 is the same as maximizing $\sum log(P(z_{it} \mid \mathbf{s}_{it}))$. We make the simplifying assumption that this is equivalent to minimizing the distance between Z_i and the most likely sequence of vectors for the state sequence $\mathbf{S}_i$. When state output distributions in the HMM are modeled by a single Gaussian, the most likely sequence of vectors is simply the sequence of means for the states in the most likely state sequence. In the rest of this paper we will refer to this sequence of means as the *target* sequence for the speaker. We can now define the objective function to be optimized for the filter parameters as:

$$Q_i = \sum_{t=1}^{T}((z_{it} - m^i_{\mathbf{s}_{it}})^T(z_{it} - m^i_{\mathbf{s}_{it}})) \quad (4)$$

where the t^{th} vector in the target sequence, $m^i_{\mathbf{s}_{it}}$ is the mean of $\mathbf{s}_{it}$, the t^{th} state, in the most likely state sequence $\mathbf{S}_i$.

It is clear from equations 2 and 4 that Q_i is a function of $\mathbf{h}_i$. Direct optimization of Q_i with respect to $\mathbf{h}_i$ is, however, not possible due to the highly non-linear relationship between the two. We therefore optimize Q using the method of conjugate gradient descent.

The filter optimization algorithm works as follows:

1. Initialize filter parameters to h_i[0] = 1/N, and h_i[k] = 0 for $k \neq 0$.
2. Process the signals for each speaker using Equation 1 and derive the feature vectors.
3. Determine optimal state sequence, and the corresponding target sequence for the optimization.
4. Estimate optimal filter parameters through conjugate gradient descent to optimize Equation 4.
5. Process signals with the new set of filters. If the new objective function has not converged go back to step 3.

Since the algorithm aims to minimize the distance between the output of the array and the target, the choice of a good target becomes critical to its performance. The next section deals with the determination of the target sequences for the various speakers.

4. TARGET ESTIMATION

The ideal target would be a sequence of Mel-spectral vectors obtained from clean uncorrupted recordings of the speaker. All other targets must be considered approximations to the ideal target. In this work we attempt to derive the target from the HMM for that speaker's utterance. This is done by determining the best state sequence through the HMM from the current estimate of that speaker's signal. A direct approach to obtaining the state sequence would be to directly find the most likely state sequence for the sequence of Mel-spectral vectors for the signal. Unfortunately, in the early iterations of the algorithm, when the filters have not yet been fully optimized, the output of the filter-and-sum array for any speaker contains a significant fraction of the signal from other speakers as well. As a result, naive alignment of the output to the HMM results in poor estimates of the target.

Instead, we also take into consideration the fact that the array output is a mixture of signals from all the speakers. The HMM that represents this signal is a *factorial* HMM (FHMM) that is the cross-product of the individual HMMs for the various speakers. In an FHMM each state is a composition of one state from the HMMs for each of the speakers, reflecting the fact that the individual speakers may have been in any of their respective states, and the final output is a combination of the output from these states. Figure 1 illustrates the dynamics of an FHMM for two speakers.

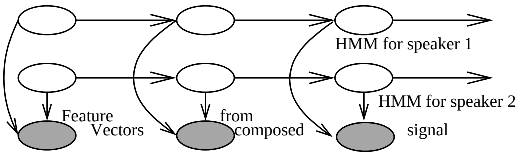

Fig. 1. Factorial HMM for two speakers (two chains).

For simplicity, we focus on the two-speaker case. Extension to more speakers is straightforward. Let S_i^k represent the i^{th} state of the HMM for the k^{th} speaker (where $k \in \{1, 2\}$). Let S_{ij}^{kl} represent the factorial state obtained when the HMM for the k^{th} speaker is in state *i* and that for the l^{th} speaker is in state *j*. The output density of S_{ij}^{kl} is a function of the output densities of its component states:

$$P(X|S_{ij}^{kl}) = f(P(X|S_i^k), P(X|S_j^l)) \tag{5}$$

The precise nature of the function $f()$ depends on the proportions to which the signals from the speakers are mixed in the current estimate of the desired speaker's signal. This in turn depends on several factors including the original signal levels of the various speakers, and the degree of separation of the desired speaker effected by the current set of filters. Since these are difficult to determine in an unsupervised manner, $f()$ cannot be precisely determined.

We do not attempt to estimate $f()$. Instead, the HMMs for the individual speakers are constructed to have simple Gaussian state output densities. We assume that the state output density for any state of the FHMM is also a Gaussian whose mean is a linear combination of the means of the state output densities of the component states. We define m_{ij}^{kl}, the mean of the Gaussian state output density of S_{ij}^{kl} as:

$$m_{ij}^{kl} = \mathbf{A}^k m_i^k + \mathbf{A}^l m_j^l \tag{6}$$

where m_i^k represents the D dimensional mean vector for S_i^k and $\mathbf{A}^k$ is a $D \times D$ weighting matrix. We consider three options for the covariance of a factorial state S_{ij}^{kl}.

- All factorial states have a common diagonal covariance matrix C.
- The covariance of S_{ij}^{kl} is given by $C_{ij}^{kl} = \mathbf{B}(C_i^k + C_j^l)$ where C_l^k is the covariance matrix for S_l^k and **B** is a diagonal matrix.
- C_{ij}^{kl} is given by $C_{ij}^{kl} = \mathbf{B}^k C_i^k + \mathbf{B}^l C_j^l$ where $\mathbf{B}^k$ is a diagonal matrix, $\mathbf{B}^k$ = diag(b^k).

We refer to the first approach as the *global covariance* approach and the latter two as the *composed covariance* approaches.The state output density of the factorial state S_{ij}^{kl} is now given by:

$$P(Z_t|S_{ij}^{kl}) = |C_{ij}^{kl}|^{-1/2}(2\pi)^{-D/2}e^{-\frac{1}{2}(Z_t - m_{ij}^{kl})'(C_{ij}^{kl})^{-1}(Z_t - m_{ij}^{kl})} \tag{7}$$

The various $\mathbf{A}^k$ values and the $C/\mathbf{B}/\mathbf{B}^k$ values are unknown and must be estimated from the current estimate of the speaker's signal. The estimation is performed using the expectation maximization (EM) algorithm. In the expectation (E) step of the algorithm, the *a posteriori* probabilities of the various factorial states, and thereby the *a posteriori* probabilities of the states of the HMMs for the speakers, are found. The factorial HMM has as many states as the product of the number of states in its component HMMs and direct computation of the E step is prohibitive. We therefore take the variational approach proposed by Ghahramani *et. al.* [6] for the computation. In the maximization (M) step of the algorithm the computed *a posteriori* probabilities are used to estimate the $\mathbf{A}^k$ as

$$\mathbf{A} = \sum_{i=1}^{N_k}\sum_{j=1}^{N_l}\sum_t (Z_t P_{ij}(t)'\mathbf{M}')(\mathbf{M}\sum_t (P_{ij}(t)P_{ij}(t)')\mathbf{M}')^{-1} \tag{8}$$

where $\mathbf{A} = [\mathbf{A}^1, \mathbf{A}^2]$, $P_{ij}(t)$ is a vector whose i^{th} and $(N_k + j)^{th}$ values equal $P(Z_t|S_i^k)$, and $P(Z_t|S_j^l)$, and **M** is a block matrix in which blocks are formed by matrices composed by the means of the individual state output distributions. The diagonal component of $\mathbf{B}^k$, b^k, is estimated in the n^{th} iteration of the EM algorithm as:

$$b_n^k = \sum_{t,i,j=1}^{T,N_k,N_l} (Z_t - m_{ij}^{kl})'(I + (B_{n-1}^k C_i^k)^{-1} B_{n-1}^l C_j^l)^{-1}(Z_t - m_{ij}^{kl}) p_{ij}(t) \tag{9}$$

where $p_{ij}(t) = P(Z_t|S_{ij}^{kl})$. The common covariance for the global covariance approach, and **B** for the first composed covariance approach can be similarly computed.

Once the EM algorithm converges and the $\mathbf{A}^k$s, the $C/\mathbf{B}/\mathbf{B}^k$ terms are computed, the best state sequence for the desired speaker can also be obtained from the FHMM, also using the variational approximation.

The overall system to determine the target for a speaker now works as follows: Using the feature vectors from the unprocessed signal and the HMMs found using the transcriptions, parameters **A** and $C/\mathbf{B}/\mathbf{B}^k$ are iteratively updated using equations 8 and 9 until the total log-likelihood converges. Thereafter, the most likely state sequence through the desired speaker's HMM is found. Once the target is obtained, the filters are optimized, and the output of the filter-and-sum array is used to reestimate the target. The system is said to have converged when the target does not change on successive iterations. The final set of filters obtained are used to separate the speaker's signal. A schematic of the overall system is shown in figure 2.

5. EXPERIMENTAL EVALUATION

Experiments were run to evaluate the proposed speaker separation algorithm. Simulated mixed-speaker recordings were generated using utterances from the test set of the Wall Street Journal(WSJ0) corpus [7]. Room simulation impulse response filters were designed for a room 4m × 5m × 3m with a reverberation time of 200msec. The microphone array configuration consisted of 8 microphones placed around an imaginary 0.5m × 0.3m flat panel display on one of the walls. Two speech sources were placed in dif-

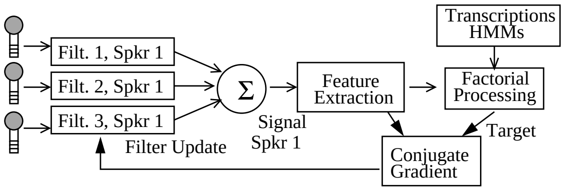

Fig. 2. Complete signal separation system for speaker 1.

ferent locations in the room. A room impulse response filter was created for each source/microphone pair. The clean speech signals for both sources were passed through each of the 8 speech source room impulse response filters and then added together.

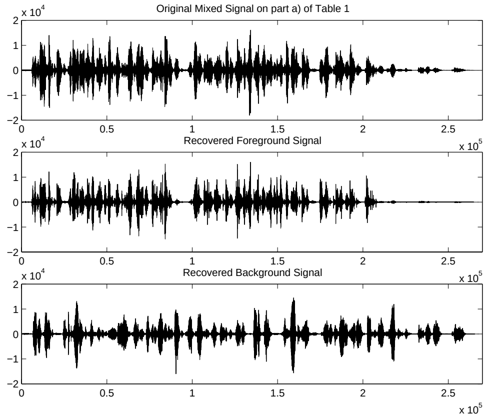

Fig. 3. Signals for signal A in table 1.

Table 1 shows the results obtained using the algorithm on two examples of two-speaker mixtures. Signal A represents a mixture where the signal levels of the two speakers are very different, *i.e.,* there is a clear foreground speaker and a background speaker. Signal B represents a mixture where the signals levels of the two speakers are comparable. The table gives the ratio of the energy of the signal from the desired speaker to that from the competing speaker, measured in decibels, in the separated signals. We refer to this measurement as the “speaker-to-speaker ratio”, or the SSR. The higher this value, the higher the degree of separation obtained for the desired speaker. The table also shows, in parentheses, the number of iterations of the algorithm required for the filters to converge. Results using various approaches are reported. The first column of the table shows the SSR obtained using simple delay-and-sum processing [4]. Here the signals are simply aligned to cancel out the delays from the desired speaker to the microphone (computed here with full prior knowledge of speaker and microphone positions) and added. This may be considered the default comparator that shows the kind of SSRs to be obtained when no further processing is performed. The second column shows the SSRs obtained when *ideal* targets have been used to optimize the filters. The ideal targets in this case are the sequences of Mel-spectral vectors derived from close-talking recordings of the same utterances that have been recorded through the microphones. The subsequent columns show the results obtained with the three methods of modelling the variances of the factorial states in the FHMM.

From table 1, it is evident that the proposed methods are all highly effective at separating the speakers. In the case where the signal levels of the two speakers are comparable, the algorithms are able to improve the SSRs by 20dB over simple delay-and-sum. For the case where the signal levels of the speakers are different, the results are more dramatic—the SSR of the background speaker in table 1, signal A, is improved by 38dB. Figure 3 shows one of the mixed signals and the two separated signals obtained on this recording. The signal separation obtained with the FHMM-based methods is, in most cases, is comparable to that obtained with ideal-targets for the filter optimization. However, the composed-variance FHMM methods converge to the final filters in fewer iterations than the method that uses a global covariance for all FHMM states.

Model Type	Delay & Sum	Clean Speech	Mean & Glob.Var	Mean & Com.Var1	Mean & Com.Var2
Signal A, Speaker 1 in background and Speaker 2 on the foreground					
SSR		Filters tuned to speaker 1			
Spkr1/Spkr2	-11dB	+36dB	+38dB (7)	+37dB (4)	+34dB (5)
SSR		Filters tuned to speaker 2			
Spkr2/Spkr1	+12dB	+24dB	+23dB (3)	+16dB (2)	+20dB (2)
Signal B, Speakers with similar loudness					
SSR		Filters tuned to speaker 1			
Spkr1/Spkr2	+1dB	+35dB	+21dB (12)	+5dB (3)	+20dB (4)
SSR		Filters tuned to speaker 2			
Spkr2/Spkr1	+1dB	+19dB	+18dB (6)	+17dB (3)	+16dB (4)

Table 1. SSRs obtained for different filters for two different signals

6. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a new multi-channel speaker separation algorithm that utilizes the known statistical characteristics of the signals from the speakers to separate them. While the algorithm is highly effective at separating signals, it is, in its current format, highly computationally intensive. In addition, as the number of speakers increases, the complexity of the FHMM computation increases. Future work with address these issues. In the specific instances of the algorithm reported in this paper, we assume fairly detailed information about the component signals is available, namely the transcription of the utterances by the multiple speakers. However, the proposed algorithm is generalizable to more “unsupervised” situations where only the speaker identity is available, or when only generic linguistic constraints about possible utterances are available. In future work we will report on these problems as well.

7. REFERENCES

[1] S. Roweis, “One Microphone Source Separation.,” *Neural Information Processing Systems* 2000.

[2] J. Hershey and M. Casey “Audio Visual Sound Separation Via Hidden Markov Models,” *Neural Information Processing Systems* 2001.

[3] A. Hyvärinen, “Survey on Independent Component Analysis,” *Neural Computing Surveys* 1999.

[4] D.H. Johnson and D.E. Dudgeon “Array signal Processing,” *Signal Processing Series, Prentice Hall* 1992.

[5] M. Seltzer, B. Raj and R.M Stern, “Speech Recognizer-Based Microphone Array Processing For Robust Hands-Free Speech Recognition” *Proc. ICASSP*, 2002.

[6] Z. Ghahramani and M.I. Jordan, “Factorial Hidden Markov Models,” *Machine Learning, Kluwer Academic Publishers*, Boston 1997.

[7] 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, p.p. 357-362, Feb. 1992.