SPARSE OVERCOMPLETE DECOMPOSITION FOR SINGLE CHANNEL SPEAKER SEPARATION

Madhusudana V. S. Shashanka

Boston University Hearing Research Center
677 Beacon St, Boston MA 02215

Bhiksha Raj, Paris Smaragdis

Mitsubishi Electric Research Labs (MERL)
201 Broadway, Cambridge MA 02139

ABSTRACT

We present an algorithm for separating multiple speakers from a mixed single channel recording. The algorithm is based on a model proposed by Raj and Smaragdis [6]. The idea is to extract certain characteristic spectro-temporal *basis functions* from training data for individual speakers and decompose the mixed signals as linear combinations of these learned bases. In other words, their model extracts a *compact code* of basis functions that can explain the space spanned by *spectral vectors* of a speaker. In our model, we generate a *sparse-distributed code* where we have more basis functions than the dimensionality of the space. We propose a probabilistic framework to achieve sparsity. Experiments show that the resulting sparse code better captures the structure in data and hence leads to better separation.

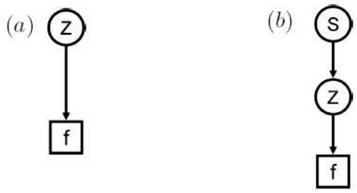
Index Terms— Separation, Speech enhancement, Minimum entropy methods, MAP estimation

1. INTRODUCTION

The problem of separating speakers from a mixed single microphone recording is an important but hard problem and is an active area of research. Research in this area can be categorized into two major approaches. In the first, the aim is to identify the time-frequency components of a mixed-signal that are dominated by a target speaker (eg [7]). Signals are reconstructed from the resulting incomplete time-frequency representations. The alternate approach attempts to construct entire spectra for each of the speakers. Characteristic spectro-temporal structures, or *basis functions*, are learned for each speaker and mixed signals are decomposed into linear combinations of these bases. Signals are reconstructed by recombining the learned bases with appropriate weights that are estimated from the mixed-signal.

In this paper, we propose a new algorithm following the latter approach that is based on a model proposed by Raj and Smaragdis [6] (henceforth referred to as the RS model). We propose to utilize *sparse, overcomplete* representations within the RS model to enable the learning of basis functions which better represent the spectral structure present in speech, thereby leading to significantly improved speaker separation.

The paper is organized as follows. We first provide some background with a review of the RS model and a discussion of the concepts of sparsity and overcompleteness in the following sections 1.1 and 1.2 respectively. We describe our model and algorithms in section 2. We present results of experimental evaluation of the model in section 3 and end with conclusions in section 4.



$$(a)\quad P_t(f)=\sum_z P_t(z)P(f|z)$$

$$(b)\quad P_t(f)=\sum_s P_t(s)\sum_{z\in\{\mathbf{z_s}\}} P_t(z|s)P_s(f|z)$$

Fig. 1. Let $P_t(f)$ represent the probability of observing frequency f in analysis frame t. (a) Graphical model and equation for the underlying process in the RS model for a single speaker. $P(f|z)$ is the probability of observing frequency f given hidden variable z, and $P_t(z)$ is the *a priori* probability of z in analysis frame t. (b) Graphical model and equation for the underlying process in the RS model for a mixture of speakers. $P_t(s)$ is the *a priori* probability of the s-th speaker and $\{\mathbf{z}_s\}$ represents the set of values that z can take for that speaker.

1.1. Latent Variable Model

Raj and Smaragdis [6] proposed an algorithm for single channel speaker separation by latent variable decomposition of the speech spectrogram. Each magnitude spectral vector in the short-time Fourier transform of a speech signal is modeled as the outcome of a discrete random process that generates frequency bin indices. In other words, the magnitude spectrum of each frame represents a scaled histogram of multiple draws from this random process. The discrete random process is modeled as a mixture of multinomial distributions over frequencies. The mixture weights of the component multinomials vary from analysis frame to analysis frame. The component multinomials are assumed to be speaker specific and are learned from training signals for each speaker. Figure 1 provides graphical models and mathematical descriptions of the underlying random process.

We can formalize the model by associating a latent variable z with every component multinomial. The conditional probabilities for frequency f, $P(f|z)$, are assumed to be constant for a given speaker while the *a priori* probability of z at analysis frame t, $P_t(z)$, is assumed to vary with t. f takes the values of the discrete frequencies of the FFT for the frame, while z takes on as many values as there are component multinomials. One can intuitively think of the component multinomials $P(f|z)$ as *basis functions* and the latent

1-4244-0728-1/07/$20.00 ©2007 IEEE

variable probabilities $P_t(z)$ as the corresponding *mixture weights* that together explain the t-th frame of the spectrogram $P_t(f)$.

The spectrum of a mixed signal is modeled as the outcome of repeated draws from a two-level random process. Within each draw, the process first draws a speaker from the mixture (represented by latent variable s in figure 1(b)), then a specific multinomial for the speaker (latent variable z), and finally a frequency index from the multinomial. To separate the spectrum for each speaker within an analysis frame, we obtain estimates of the mixture weights for each speaker given the speaker-specific multinomial distributions that were learned from training data. The separated spectrum for the speaker within the frame is finally obtained as the expected value of the number of draws of each frequency index from the mixture multinomial distribution for the speaker.

1.2. Sparsity and Overcompleteness

The idea of sparsity originated from attempts at understanding the general information processing strategy employed by biological sensory systems (eg [4]). The assumption is that the goal of sensory coding is to transform the input in such a manner that reduces the redundancy present among the elements of the input stream. Typically, the input space has *some* structure (is not completely random) and the idea is to take advantage of this redundancy to produce more efficient representations of the environment.

There are two approaches with which one can utilize the redundancy of the inputs. In a *compact code*, the goal is to represent all the likely inputs with a relatively small number of vectors with minimal loss in the description of the input (eg. minimal mean-squared error). In such a code, the dimensionality of the representation is reduced. Principal Components Analysis (PCA) is an example of that approach.

In a *sparse-distributed code*, the number of elements in the code is at least as large as the dimensionality of the input. In other words, the code is *overcomplete*. However, the number of elements of the code needed to describe a particular instance of the input is minimized. The goal is to obtain a code where only a few elements are required to explain a given input.

1.3. Sparsity and Overcompleteness for Speaker Separation

The application of the concepts of sparsity and overcompleteness to speaker separation is best explained through the following hypothetical example: assume that the complete set of speech spectra that a talker can generate is known for two talkers. All spectral vectors in any mixed recording are obtained by linearly combining one spectrum each from the respective sets of the two talkers.

The set of all possible spectral vectors for any talker may be viewed as the set of bases for that talker. The set must, by nature, be overcomplete since there are many more vectors than there are dimensions in the vectors. A spectral vector from a mixed signal may be explained as a linear combination of the bases from the two talkers. However, such an explanation would be uninformative since the vector may be equally well explained by several combinations of bases. However, if it is also required that the explanation be sparse, i.e. that the vector be explained using the smallest number of bases, the set of possible solutions reduces – in the best case it will only include the solution that describes the vector as a combination of the two bases vectors (one from each talker) that actually combined to form it, thereby permitting exact separation of the two talkers from the mixed spectrum. Such a solution which would not be possible using compact basis sets.

More generally, a finite overcomplete basis set may estimated for a talker by stipulating the all spectral vectors in a training set be explained through the sparsest combinations of bases from the set. It may be expected that decomposition of spectra from mixed speech signals using such bases sets will result in better separation than may be possible using compact basis sets.

2. SPARSE OVERCOMPLETE DECOMPOSITION MODEL

We first introduce some matrix notations that will help with the exposition. For a given speaker, let the two-dimensional distribution corresponding to the spectrogram be represented by the $F \times T$ matrix $\mathbf{V}$. Let all the K component multinomial distributions be represented by the $F \times K$ matrix $\mathbf{W}$ with the k-th column corresponding to the k-th multinomial distribution i.e. $W_{fk} = P(f|z = k)$. Let the probabilities of the latent variables at all frames by represented by the $K \times T$ matrix $\mathbf{H}$ where $H_{kt} = P_t(z = k)$. Every column of $\mathbf{H}$, just like every column of $\mathbf{W}$, corresponds to a probability distribution and hence sums to unity. With this notation, the model given by equation (a) in figure 1 can be equivalently written as $\mathbf{V} = \mathbf{WH}$. Finally, we shall refer to the columns of $\mathbf{W}$ as *basis functions* and the t-th column of $\mathbf{H}$ as *mixture weights* or just *weights* for frame t.

There are two stages in speaker separation algorithms: learning the parameters of each speaker from training data, and using the learned parameters to separate speakers from mixed signals.

Consider the training stage. The RS model is given by equation (a) of figure1. Our model is described by the same equation. The crucial difference lies in *how* we estimate parameters. The RS model produces a *compact code*. The goal of RS model in the training stage is to find matrices $\mathbf{W}$ and $\mathbf{H}$ given a $F \times T$ spectrogram $\mathbf{N}$. Implicit in the model is the assumption that there is a reduction in dimensionality i.e. $K < F$. In the examples given in [6], $F = 513$ and $K = 100$.

In contrast, the goal of our model is to produce a *sparse-distributed code*. The first requirement is that the basis set be overcomplete i.e. $K \geq F$. An overcomplete basis set by itself can result in more than one feasible solution. We will have to explicitly enforce sparsity so that we get unique solutions. Below, we show how we enforce sparsity during estimation to produce a sparse-distributed code of basis functions. This forms the crux of our contribution in this paper.

We use the concept of *entropic prior* introduced in [1] to enforce sparsity. Given a probability distribution $\boldsymbol{\theta}$, entropic prior is defined as

$$P_e(\boldsymbol{\theta}) = e^{-\mathcal{H}(\boldsymbol{\theta})} \tag{1}$$

where $\mathcal{H}(\boldsymbol{\theta}) = -\sum_i \theta_i \log \theta_i$ is the entropy of the distribution. A sparse representation, by definition, has few "active" elements which means that the representation has low entropy. Hence, imposing this prior during *maximum a posteriori* estimation is a way to minimize entropy during estimation which will result in a sparse $\boldsymbol{\theta}$ distribution. If we want a sparse distributed set of basis functions, we need to impose sparsity on the distributions over the latent variable z for all frames i.e. on every column of $\mathbf{H}$.

We use the Expectation-Maximization algorithm to derive the update equations. Let Λ represent the parameters of the model and super-script (i) denote the i-th iteration. Let $\mathbf{N}$ be the spectrogram available as training data. $P(\Lambda)$ represents the prior knowledge we have about the parameters i.e.

$$\log P(\Lambda) = \beta \sum_t \sum_z P_t(z) \log P_t(z)$$

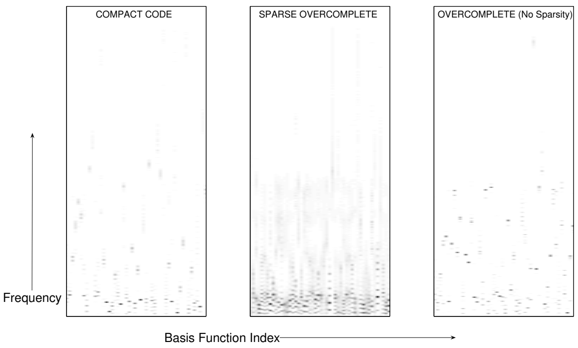


Fig. 2. Examples of a subset (fifty) of basis functions learned for a male speaker. The left panel corresponds to the compact code where 100 bases were learned in total. The mid-panel corresponds to a sparse-distributed code where 1000 basis functions were learned by imposing sparsity. The rightmost panel shows a subset of 1000 basis functions learned without the imposition of sparsity. The sparse code shows harmonic structure present in speech while the overcomplete code without sparsity resembles a set of impulse distributions.

where β is a parameter indicating the degree of sparsity desired. We can write the E-step as

$$P_t(z|f) = \frac{P_t(z)P(f|z)}{\sum_{z'} P_t(z')P(f|z')}, \tag{2}$$

and the M-step as

$$P(f|z) = \frac{\sum_t N_{ft} P_t(z|f)}{\sum_{f'} \sum_t N_{f't} P_t(z|f')}, \tag{3}$$

$$\frac{\alpha \sum_f N_{ft} P_t(z|f)}{P_t(z)} + \beta + \beta \log P_t(z) + \lambda = 0 \tag{4}$$

where λ is a Lagrange multiplier and α is an unknown scaling factor (so that all entries of $\alpha\mathbf{N}$ are integers). For simplicity, we write it as

$$\frac{\omega_z}{\theta_z} + \beta + \beta \log \theta_z + \lambda = 0 \tag{5}$$

where $\omega_z = \alpha \sum_f N_{ft} P_t(z|f)$ and $\theta_z = \log P_t(z)$. We need to solve for θ_z for all t. [1] proposes a method to solve the above system of simultaneous transcendental equations for θ_z using the Lambert $\mathcal{W}$ function [3], an inverse mapping satisfying $\log \mathcal{W}(u) + \mathcal{W}(u) = \log u$. Rearranging the terms in (5), one can derive

$$\hat{\theta}_z = \frac{-\omega_z/\beta}{\mathcal{W}(-\omega_z e^{1+\lambda/\beta}/\beta)} \tag{6}$$

Equations (5) and (6) form a set of fixed-point iterations that typically converge in 2-5 iterations [1]. Details on computing the Lambert $\mathcal{W}$ function can be found in [2].

The final update equations are given by equations (2), (3) and the fixed-point equations (5) and (6). Factors α and β weight the relative contributions of data and prior respectively. They have to be chosen empirically based on the application domain and the particular problem being solved.

Once we have trained basis functions for each speaker, we can use them to separate speakers from a mixed single channel recording. The algorithm is identical to the separation stage of the RS model and one can refer to [6] for equations.

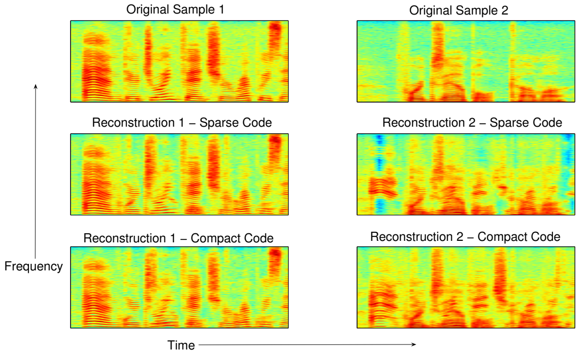


Fig. 3. Examples spectrograms of reconstructions from a mixture with female and male speakers. Sample 1 corresponds to the female speaker and sample 2 corresponds to the male speaker. One can notice that reconstructions with the sparse code are better than reconstructions with the compact code, especially for the male speaker in this example.

3. EXPERIMENTAL EVALUATION

Experiments were conducted to evaluate the speaker separation performance of the proposed algorithm on synthetic mixtures. We evaluated on six pairs of speaker combinations: two pairs were female/male, two were male/male and two were female/female.

A set of about 135 utterances comprising approximately 16 minutes of speech was used as training data for each speaker. Signals were sampled at 16 kHz and short-term Fourier transforms were generated with an FFT point size of 1024, hop size of 256 between frames, and a hanning window. The dimensionality of each spectral vector was 513 ($F = 513$) and they were modeled by a mixture of 1000 multinomial distributions ($K = 1000$). Factors α and β of equation (5) were empirically chosen to be 1 and 0.7 respectively. Thus, a set of 1000 multinomial distributions were learned from the training data for each speaker (*sparse-distributed code*).

We also evaluated the RS model on the same data. As in the original paper, we used a set of utterances of approximately 30 seconds as training data for each speaker. A set of 100 multinomial distributions were learned from the data for each speaker (*compact code*). Figure 2 shows examples of basis functions learned for a male speaker. The left and mid panels correspond to compact and sparse-distributed codes respectively. Also shown are basis functions that result when sparsity is not imposed in the overcomplete case.

For a given pair of speakers, mixed signals were obtained by digitally adding test signals for both speakers. The length of the mixed signal was set to the shorter of the two signals. The component signals were all normalized to 0 mean and unit variance prior to addition, resulting in 0 dB SNR for each speaker. A set of five mixed recordings were obtained for every pair of speakers considered. Mixed signals were separated using both the sparse-distributed code and the compact code according to the separation stage of the RS model. Figure 3 show example spectrograms of reconstructions for a mixture with male and female talkers. .

The quality of speech separation is hard to evaluate reliably. We provide two measures that have been used in the literature. Let $\mathbf{O}$ and $\mathbf{R}$ represent the magnitude spectrograms of the original test signal and the reconstructed signal of the i-th speaker in the mixture.

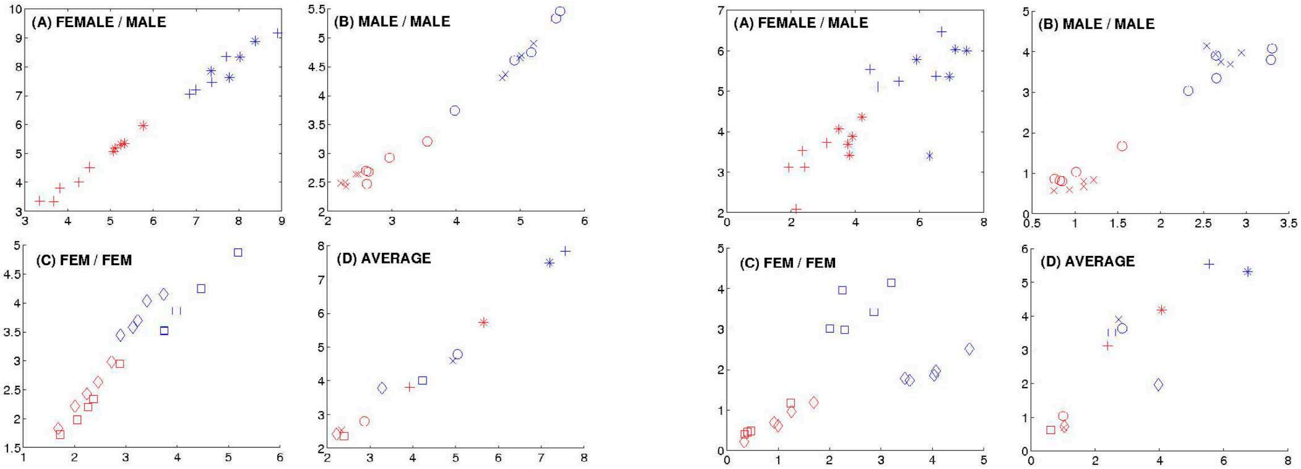


Fig. 4. Evaluation results in terms of SNR improvements (in dB, left panels) and SER (in dB, right panels) for the sparse distributed code (in BLUE) and for the compact code (in RED). The Y-axis corresponds to SNR_1 (SER_1 in the right panels) in dB and X-axis corresponds to SNR_2 (SER_2 in the right panels). Each point corresponds to a particular experiment. Different symbols used for the points represent different speaker combinations in the mixture. Each point in Panel (D) is the average of the corresponding points in the first three panels.

Let $\mathbf{N}$ and $\mathbf{\Phi}$ represent the magnitude and phase of the mixture spectrogram. Define a function

$$g(\mathbf{X}) = 10\log_{10}\left(\frac{\sum_{f,t} O_{ft}^2}{\sum_{f,t} |O_{ft}e^{j\Phi_{ft}} - X_{ft}e^{j\Phi_{ft}}|^2}\right). \quad (7)$$

We define the *SNR improvement* for the i-th speaker [6] as

$$SNR_i = g(\mathbf{R}) - g(\mathbf{N}) \quad (8)$$

The second metric, *Speaker Energy Ratio (SER)* [5] is based on correlations between reconstructed and original signals. It is given by

$$SER_i = 10\log_{10}\left(\frac{c_{ii}}{\sum_{\forall j \neq i} c_{ij}}\right) \quad (9)$$

where c_{ij} is the correlation between the reconstructed time signal for the i-th speaker and the original signal for the j-th speaker.

Results of our experiment are summarized in figure 4. For every speaker pair, separation was evaluated using both the sparse code and the compact code for five different mixtures. Every point in the figures corresponds to the result of one experiment. In the panels on the left, we plot the SNR improvements of the two reconstructed signals against each other while in the panels to the right, we plot the speaker energy ratios. Points in blue correspond to results with the sparse code while points in red correspond results from the compact code. All the results for a given speaker pair are represented by the same symbol and different symbols have been used for different speaker pairs.

Results show that the sparse code performs significantly and consistently better than the compact code on the basis of both metrics. Perceptual tests confirm this. We also performed an experiment where we used 1000 basis functions trained without the imposition of sparsity. The resulting basis functions resemble more like impulse distributions across frequencies instead of capturing the harmonic structure present in speech. As expected, the separation performance was poor but we don't report the results here due to lack of space. A few examples of separated signals can be obtained at http://cns.bu.edu/∼mvss/courses/speechseg/.

4. CONCLUSIONS

In this paper, we have proposed an algorithm to train sparse-overcomplete bases in the framework of a mixture multinomial model for speaker separation. It is demonstrated that that the quality of speaker separation obtained with sparse-overcomplete basis functions is superior than what one can obtain with a compact set of bases.

5. REFERENCES

[1] ME Brand. Pattern discovery via entropy minimization. In *Uncertainty 99: AISTATS 99*, 1999.

[2] ME Brand. Structure learning in conditional probability models via an entropic prior and parameter extinction. *Neural Computation*, 11(5):1155–1182, 1999.

[3] RM Corless, GH Gonnet, DEG Hare, DJ Jeffrey, and DE Knuth. On the lambert w function. *Advances in Computational mathematics*, 5:329–359, 1996.

[4] DJ Field. What is the goal of sensory coding? *Neural Computation*, 6:559–601, 1994.

[5] Smaragdis P. Convolutive speech bases and their application to supervised speech separation. *IEEE Trans on Audio, Speech and Language Processing*, 2007 (to appear).

[6] B Raj and P Smaragdis. Latent variable decomposition of spectrograms for single channel speaker separation. In *IEEE WASPAA*, 2005.

[7] ST Roweis. Factorial models and re-filtering for speech separation and denoising. In *EUROSPEECH*, volume 7, pages 1009–1012, 2003.