DICTIONARY AND SPARSE DECOMPOSITION METHOD SELECTION FOR UNDERDETERMINED BLIND SOURCE SEPARATION

B. Vikrham Gowreesunker and Ahmed H. Tewfik

Dept. of Electrical and Computer Engineering, University of Minnesota
Minneapolis, MN 55455
phone: + (1)612-625-6024, fax: +(1)612-625-4583

ABSTRACT

In underdetermined BSS problems, it is common practice to exploit the underlying sparsity of the sources. In this work, we propose two approaches to improve the quality and robustness of current algorithms that rely on source sparsity. First, we highlight the benefits of using a matched dictionary as opposed to a standard overcomplete dictionary for separation. Second, we investigate the problem of additive noise for geometric separation methods such as the Hough Transform, and propose using a BESS decomposition algorithm as a robust method for estimating the mixing matrix in the presence of noise. We find that current sparse decomposition methods fail to take advantage of optimal dictionary design and suggest pursuing representations that are less sparse for signal mixtures.

1. INTRODUCTION

In the blind source separation problem, we have mixtures of several source signals and the goal is to separate them with as little prior information as possible, hence the term blind. In this work, we study the instantaneous underdetermined BSS case, where we have more sources than sensors/mixtures. We are concerned with separating mixtures of speech signals when the mixing matrix and number of underlying sources are unknown, and where additive white noise is also present at the sensors. This problem is intrinsically ill-defined and its solution requires some additional assumptions compared to its overdetermined counterpart.

The difficulty of the underdetermined setup can be somewhat alleviated if there exists a representation where all the sources are rarely simultaneously active, which entails finding a representation where the sources are sparse. Some authors have shown that speech signals are sparser in the time-frequency than in the time domain, and that there exists several other representations such as wavelets packets, where different degrees of sparsity can be obtained [11]. It has been shown that better separation can indeed be achieved by exploiting this improvement in sparsity [5],[10],[6]. In this paper, we further investigate how to get the best possible results based on this sparsity assumption. We compare the sparsity property of speech signals for a standard overcomplete representations, and training based dictionaries [2] and advocate the benefits of using trained dictionaries. One type of approach to using sparsity is a geometric method such as the Hough Transform. This method was previously used by the authors of [9] in the time domain for sources whose joint probability distribution have long tails and by [6] who combined it with a multichannel Matching Pursuit algorithm to address the problem of underdetermined BSS in stereo mixtures. In both of these above mentioned cases, we find that the Hough Transform is very sensitive to the presence of additive white noise at the sensors. We propose a method to make the estimation through the Hough Transform robust to the presence of noise. We propose a separation algorithm which estimates the mixing matrix using the Hough Transform and a Bounded Error Subset Selection(BESS) decomposition [4], and performs the separation by a nearest neighbor assignment.

The remainder of the paper is organized as follows, in section 2 we give a mathematical description of the problem and a detailed explanation of how sparsity is used in source separation, in section 3 we show the importance of choosing the proper dictionary to best exploit the underlying sparsity. In section 4 we illustrate how current sparse decomposition algorithm fails to distribute some coefficients to the proper source. We illustrate the shortcomings of the Hough method in the presence of additive sensor noise for the MDCT, Matching Pursuit(MP) and Multichannel MP algorithms. We propose a solution using the BESS algorithm and demonstrate its usefullness in estimating the mixing matrix.

2. MIXTURE MODEL FOR AN ARBITRARY DICTIONARY

For a problem where we have M mixtures of N sound sources and $M < N$, our goal is to separate the sources into individual tracks. We are concerned with the underdetermined linear instantaneous mixture model, which can be formulated mathematically as follows,

$$x(t) = As(t) + q(t) \tag{1}$$

where $s(t)$ is an unknown $Nx1$ vector containing the source data, $x(t)$ is a known $Mx1$ observation vector, $q(t)$ is the $Mx1$ additive noise vector, t is the sample index and A is an unknown MxN mixing matrix. Over T time samples, we have the following expression,

$$X = AS + Q \tag{2}$$

where $X = [x[1]x[2]\dots x[T]]$, $S = [s[1]s[2]\dots s[T]]$, and $Q = [q[1]q[2]\dots q[T]]$.

One approach to solving this problem is to assume that the sources are sufficiently sparse in a given representation. To solve the underdetermined BSS problem using sparsity, one can decompose the signal into a dictionary where the source signals are known to be sparse or use a Sparse Decomposition (SD) algorithm to find a sparse representation

for an overcomplete dictionary. In the rest of this section, we illustrate how sparsity can lead to separation, then show how sparsity in alternative representations can be exploited.

To simplify the discussion, let us assume without loss of generality that M = 2 and N = 3. Assuming there is no additive noise for illustrative purposes, we can expand Eq. 2 as follows

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \times \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} \quad (3)$$

By looking at the ratio $\frac{x_1}{x_2}$ for the case when only the j^{th} source is active, we get

$$\frac{x_1}{x_2} = \frac{a_{1j}}{a_{2j}} \quad (4)$$

Hence a scatter plot of x_1 v/s x_2 for the case where the sources never overlapped would reveal 3 distinct lines such that the j^{th} source corresponds to the line with gradient $\frac{a_{1j}}{a_{2j}}$. Separation can be easily achieved for Eq. 4. The general thinking is that the sparser the sources are, the less likely they are to be active at the same time, resulting in better separation. The same argument can be applied when the data is represented in a different dictionary. The dictionary can be a basis matrix or an overcomplete matrix and each column is referred to as a dictionary atom. We can express signal,$\{s_j\}_{j=1}^{N}$, in terms of the dictionary, D and coefficient matrix, C such that

$$S^T = DC \quad (5)$$

Substituting in Eq. 2, We get

$$X = AC^T D^T \quad (6)$$

where C is a $K \times M$ coefficient matrix and D is a $T \times K$ matrix. Thus, it is clear that the sparsity of the source is inherently limited by the dictionary used. In the following section we illustrate how the quest for a sparse representation entails finding a good dictionary.

3. EFFECT OF DICTIONARY ON SIGNAL SPARSITY

For source separation purposes, speech signals tend to display poor sparsity in the time domain. Fortunately other representations have been shown to be better for separation purposes [11]. In this section, we are interested in exploiting sparsity resulting from overcomplete dictionaries. There are two stages required to evaluate the performance of overcomplete dictionaries. First, the dictionaries need to be designed, and second, a proper decomposition algorithm is needed to find a sparse representation of the signal vector. In this section, we compare the sparsity of speech signals for a matched and unmatched overcomplete dictionary. The matched dictionary was designed using the KSVD method [2] and the unmatched dictionary chosen was a cosine packet(CP) dictionary. They were compared using the Orthogonal Matching Pursuit(OMP) [7] decomposition algorithm.

3.1 Dictionary Design

3.1.1 Unmatched Dictionary

The CP dictionary was designed using the Atomizer and Wavelab Matlab toolbox[1], with dimensions 128×896.

Sources	speaker 1	speaker 2	speaker 3
CP Dictionary	5014	4123	4718
Dictionary1	2317	6291	6192
Dictionary12	2164	1347	5299
Dictionary123	1981	1709	2742

Table 1: Number of Nonzero coefficients when using OMP

Sources	(s1,s2)	(s1,s3)	(s2,s3)	(s1,s2,s3)
CP Dictionary	58	173	138	7
Dictionary1	24	55	235	1
Dictionary12	3	28	24	0
Dictionary123	1	8	25	0

Table 2: Overlap of coefficients for the 3 sources. Each column on the right shows overlap for a set of sources. E.g (s1,s2) is source1 and source2

3.1.2 KSVD Dictionary

The matched dictionary was designed using a KSVD design method. This method takes after the Vector Quantization technique used in codebook design and tends to promote a sparse structure. The details of this algorithm is given in [2]. The speech data used for training the dictionary had to be first formatted into frames of length, T, with a standard windowing technique. The dictionary was initialized with a CP dictionary and the reconstruction error threshold used was 0.01. The dimension of the trained dictionary was the same as the CP, 128×896.

3.2 Method

To compare the sparsity improvement of the matched and unmatched dictionaries, we need a proper sparse decomposition(SD) algorithm. The goal of a SD algorithm is to find a sparse coefficient vector c of size $K \times 1$, such that

$$\|s - Dc\|_2 \leq \varepsilon, \quad (7)$$

where ε is the approximation error, s is a given signal vector of size $T \times 1$ and D is an overcomplete dictionary of size $T \times K$, with $K > T$.

Two different speech signals were decomposed into the matched and unmatched dictionary and the number of nonzero coefficients was used to compare the sparsity of the representation. Furthermore, to verify the idea that sparser representation results in lesser coefficient overlap, we also evaluate how much overlap is achieved when the two signals are represented in the different dictionaries.

3.3 Results

We ran four different experiments. Using 3 speech sources, which we call source1, source2 and source3, we trained 3 different dictionaries with the following properties.

- Dictionary1 is trained using source1 only
- Dictionary12 is trained using source1 and source2
- Dictionary123 is trained using source1, source2 and source3

For each experiment, the 3 speech sources were individually decomposed using OMP onto first a CP dictionary and then the KSVD designed dictionaries 1, 12, and 123. The number of nonzero coefficients were then recorded and compared in table 1. In all 3 experiments, source1 is clearly much sparser with the KSVD dictionary than the CP. Looking at source2 and source3 for dictionary1, and source3 for dictionary 12, we see that when sources are decomposed into dictionaries that are not trained for them, they exhibits decreased sparsity, i.e. they more nonzero coefficients than in the CP case. In the experiment with dictionary 123, we see how a dictionary trained using all speakers, induces a very high sparsity improvement for all the sources.

Another analysis performed on these data was to compare the number of overlapping coefficients of each source when they were represented in the same dictionary. In table 2, we see that the trained dictionaries which induces higher sparsity also results in very few overlapping coefficients. Hence, we can see that having optimal dictionary definitely offers an advantage in the source separation process.

4. CHOOSING A SPARSE DECOMPOSITION METHOD

Our results for matched dictionary confirmed the idea that improved sparsity in the source should improve separability of the signals. However, in practice we have only mixtures available for separation and there is an infinite number of ways that they can be represented in an overdetermined dictionary. There is no evidence to suggest that the most compact representation of the mixtures will result in the best separability. In fact, the results that we present below suggest that the sparsest representations of the mixture do not correspond to the weighted sums of the sparsest representation of the sources.

In this section we compare the performance of a few single channel pursuit algorithms such as the MP, OMP, BESS, and a multichannel MP method for an overdetermined dictionary, with an Modified Discrete Cosine Transform (MDCT). We first explain the Hough Transform method that we use for separation, then proceed to present the difference between the different decomposition techniques. In the last subsection, we propose using one of the decomposition method, the BESS method, to improve the separation in the MDCT domain when using the Hough method, and additive noise is present.

4.1 Separation using The Hough Transform Method

The Hough Transform is a method frequently used in the image processing community to find lines in images. As discussed in section 2, if the signals are sufficiently sparse, the coefficients of the mixtures signals should conglomerate around lines corresponding to the columns of the mixing matrix. Points closest to each line can be clustered and used to construct an estimate of the signals of interest. The Hough method is a straight forward method for finding the lines on the scatter plot. Every point on the plot, can be parameterized into an angle and an intercept, which is zero in this case. To estimate the mixing matrix A, we find the angle,

$$\theta = atan(\frac{x_1}{x_2}), \tag{8}$$

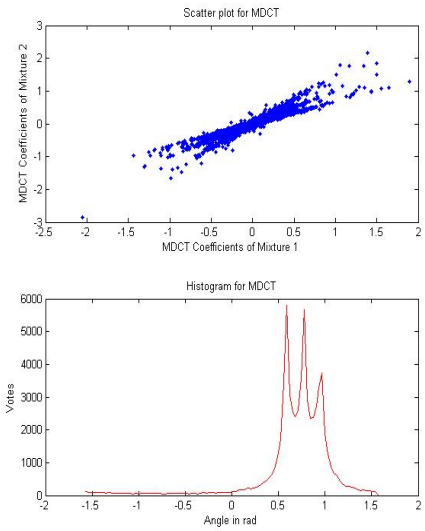

Figure 1: (a) At the top we have the scatter plot for coefficient of the MDCT of mixture 1 v/s mixture 2 (b) At the bottom, the histogram of the MDCT coefficient is shown

corresponding to each point from the scatter plot and plot a histogram of their occurrence over a quantized angular scale. Peaks on the histogram corresponds to potential lines. A peak detection algorithm can be used to find the angle corresponding to these lines, and one can subsequently estimate the columns of the mixing matrix up to a permutation and scale. This method does not require apriori knowledge of number of sources.

4.2 Hough Transform for MDCT

In the top portion of Fig 1, we plotted a scatter plot of the coefficients of an MDCT transform of two mixtures of three speech signals. We can clearly see three straights lines that were confirmed to lie long the columns of the mixing matrix. The goal is to find the angle corresponding to these lines. A histogram plot of the data is shown in the bottom portion of Fig. 1 and reveals three clear peaks corresponding to 3 lines/sources, indicating that the MDCT domain is quite conducive to exploit separation due to sparsity.

4.3 Sparse Decomposition for Overcomplete Dictionaries

For projection onto overcomplete dictionaries, we need to use sparse decompositions algorithms before any separation. We tried two types of algorithms, the single channel decomposition techniques such as MP, OMP and BESS, and the multichannel pursuit method as explained in [6]. In Fig. 2, we show the scatter plot for coefficients from performing single channel BESS independently on the each mixture signal. There are two interesting observations about this plot. First, there are a set of coefficients that appear on the axes, and second, the lines corresponding to mixing matrix columns are clearly visible. The coefficients on the axis are dictionary elements that appear in one mixture but not in the other, and belong to more than one source. The coefficients along the matrix orientation share the same dictionary atoms

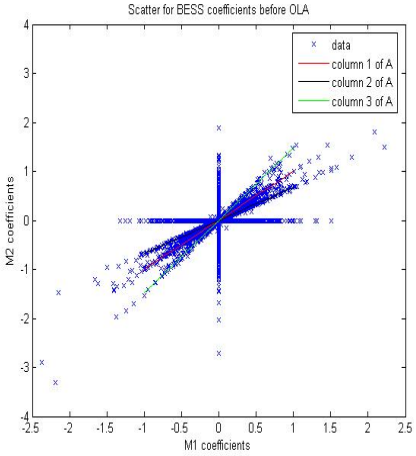

Figure 2: Scatter plot for BESS without noise

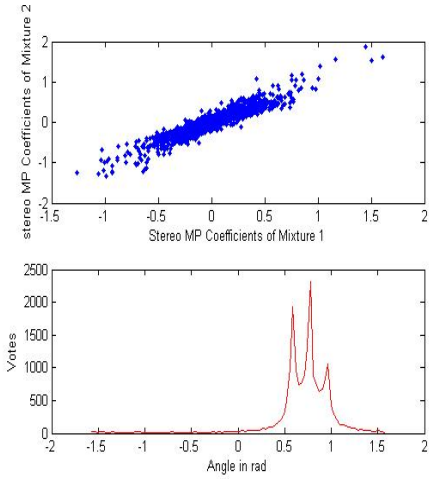

Figure 3: Scatter plot and Histogram for multichannel MP without noise

for both mixtures and belong to a single source. We find that the single channel MP and OMP also exhibit the same behavior. Figures were omitted here for space consideration.

An alternative to the single channel decomposition, is a multichannel counterpart such as the multichannel MP developed in [6], where the coefficients of the channels are constrained to share the same dictionary atoms. A scatter plot and the corresponding histogram of the multichannel MP is shown in Fig.3. Although this method makes separation possible for overcomplete dictionaries, it still is not optimal. We still have a large number of coefficient that do not align along the mixing matrix.

4.4 A Robust approach to estimate the mixing matrix in the presence of additive noise

As explained earlier, for the Hough Transform to work, one needs an accurate estimation of the peaks. Unfortunately, we find that in the presence of additive white noise, the histogram can get smudged and it is not possible to locate the peaks. Below, we investigat the effect of noise on the Hough

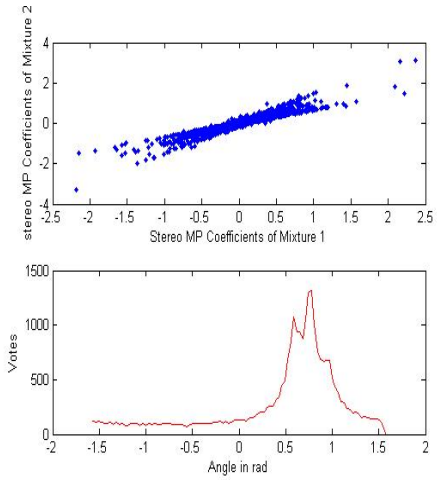

Figure 4: Histogram for multichannel MP in the presence of noise

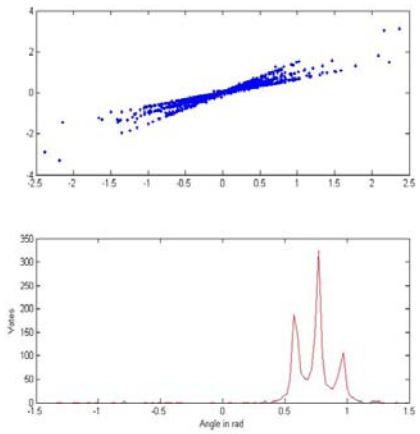

Figure 5: Histogram for BESS in the presence of noise

Method.

4.4.1 *Procedure*

We generated a mixture of three noncontinuous speech tracks of 4 seconds length each, used that as an input to our algorithms. White noise was added to find the behavior in the presence of noise. We used data frame of length, N = 512 and a CP dictionary of size 512×4608, generated using the Atomizer package. Results for matched dictionary are not shown here, but were very similar. We now present some results which compares the performance of the MDCT, BESS, MP, OMP, and Multichannel MP sparse decomposition algorithms in the presence of noise.

4.4.2 *Results*

In Fig.4, we show the scatter plot and histogram of for the Multichannel MP in the presence of additive noise. The peaks are no longer as clear as in Fig.3 and we cannot use the previous method to estimate the mixing matrix anymore. This problem is also seen in the MDCT case, the OMP and the multichannel MP decomposition. In Fig. 5, we show the

scatter plot and histogram for the BESS algorithm, as outlined in section 4.2. The peaks are still clearly detectable and do indeed give an accurate estimate of the mixing matrix. From these results, it is clear that BESS provides a robust method for estimating the mixing matrix and that reliable peak estimation for the other methods is not possible.

4.4.3 Algorithms

We propose the following procedure for mixing matrix estimation,

- Decompose the mixtures using the BESS algorithm
- Find the coefficients that share the same dictionary atoms
- Calculate their Hough Transform and plot the histogram for these data points only
- Find the location of the peak with a peak detection algorithm
- Calculate mixing matrix column from the peak estimates

With the new mixing matrix estimation method, the complete robust procedure for separating sources using the Hough Method is,

- For each data frame, normalize to unit norm, window with a 50% overlap and decompose using BESS
- Estimate mixing matrix using the procedure outlined above
- Find the MDCT decomposition of the original data frame and compute its Hough Transform
- Use the estimated mixing matrix from the BESS to separate the data in the MDCT domain
- Find the inverse MDCT transform and do the proper overlap and add to recover sources

We performed subjective listening test for the above mentioned algorithm, and found that the separation quality was just as good as in [5] but with higher levels of artifacts. The relative higher level of artifact is due primarily to the simple nearest neighbor assignment we employed for coeffients far from the matrix orientation. This is an area we hope to improve on as we develop better decomposition tools.

5. CONCLUSION

We discussed the implications of sparsity on source separation and explored different avenues to maximize separation for given signals. We looked at both the dictionary selection problem and selection of decomposition algorithms, and found out that the BESS algorithms provides a robust estimate of the mixing matrix in the presence of noise. In this work, we clearly illustrated how the choice of the proper dictionary makes a difference in the sparsity of the coefficients. However, we find that currently available sparse decomposition algorithm fail to take proper advantage of even well matched dictionary. Improvement in this area will have to first address how to deal with the extra coefficients that appeared on the axes when single channel decomposition is applied. One avenue that we are currently pursuing is representations for mixtures of signals that are less compact.

REFERENCES

[1] http://www-stat.stanford.edu/∼wavelab/

[2] M. Aharon, M. Elad, and A. Bruckstein, "The K-SVD: An Algorithm for Designing of Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. on Signal Processing*, vol. 54, pp. 4311–4322, November. 2006.

[3] M. Alghoniemy and A. H. Tewfik, "A sparse solution to the bounded subset selection problem: a network flow model approach," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, vol. 5, pp. 89–92, May 2004.

[4] M. Alghoniemy and A. H. Tewfik, "Reduced Complexity Bounded Error Subset Selection," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pp. 725–728, March. 2005.

[5] C.Fevotte and S. Godsill, "A Bayesian approach for blind separation of sparse sources," *IEEE Trans. on Speech and Audio Processing*, vol. 14, pp. 2174–2188, November. 2006.

[6] R. Gribonval, "Sparse decomposition of stereo signals with Matching Pursuit and application to blind separation of more than two sources from a stereo mixture," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 3057–3060, May 2002.

[7] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, 2nd edition ed., 1999.

[8] S. Mallat and Z.Zhang, "Matching Pursuit with Time-frequency Dictionaries," *IEEE Trans. on Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[9] H. Shindo and Y. Hirai, "Blind Source Separation by a Geometrical Method," in *Proceedings of the 2002 International Joint Conference on Neural Networks (IJNN)*, pp. 1109–1114, May 2002.

[10] V. Y. Tan and C. Fevotte, "A study of the effect of source sparsity for various transforms on blind audio source separation performance," in *Workshop on Signal Processing with Adaptative Sparse Structured Representations (SPARS'05)*, Rennes, France, Nov 2005.

[11] M. Zibulevsky, B. A. Pearlmutter, P. Bofill, and P. Kisilev., "Blind Source Separation by Sparse Decomposition," chapter in the book: S. J. Roberts, and R.M. Everson eds., Independent Component Analysis: Principles and Practice, Cambridge, 2001.