interspeech 2008 incorporating SST 08

Enhancement of Noisy Speech Recordings via Blind Source Separation

Jiri Malek, Zbynek Koldovsky, Jindrich Zdansky, and Jan Nouza

Technical University of Liberec, Studentská 2, 461 17, Liberec, Czech Republic

{jiri.malek,zbynek.koldovsky,jindrich.zdansky,jan.nouza}@tul.cz

Abstract

We propose an improved time-domain Blind Source Separation method and apply it to speech signal enhancement using multiple microphone recordings. The improvement consists in utilization of fuzzy clustering instead of a hard one, which is verified by experiments where real-world mixtures of two audio signals are separated from two microphones. Performance of the method is demonstrated by recognizing mixed and separated utterances from the Czech part of the European broadcast news database using our Czech LVCSR system. The separation allows significantly better recognition, e.g., by 32% when the jammer signal is a Gaussian noise and the input signal-to-noise ratio is 10dB.

Index Terms: Blind Source Separation, Independent Component Analysis, Speech Enhancement, Cluster Analysis

1. Introduction

Automatic speech recognition (ASR) aims at conversion of spoken language into a written text. Since the speech is usually recorded in a noisy environment, the efficiency of transcription can be deteriorated. Therefore, a preprocessing technique could be used to restore the speech signal and improve thus the recognition rate. In this paper, we consider the case where the speech is interfered by other acoustical sources in an ordinary room, which is recorded by $m > 1$ microphones. This situation is commonly known as the cocktail-party problem recently solved by Blind Source Separation (BSS). Here, we propose an improvement of the audio BSS method from [1], and we apply it to recover the speech.

The general goal of BSS is to separate d original signals from their mixtures obtained by the microphones. Doing this, the desired speech is thus obtained as one of the separated signals. A practical assumption that is used to make the blind separation possible is that the original signals, i.e. the speech and other interferences, are independent. Then, Independent Component Analysis (ICA) [2, 3, 4] can be used as the core method for BSS.

The convolutive model describing propagation and mixing of sources in a natural environment is given by

$$x_i(n) = \sum_{j=1}^{d} \sum_{\tau=0}^{M_{ij}} h_{ij}(\tau) s_j(n-\tau), \quad i = 1, \ldots, m, \qquad (1)$$

where $x_1(n), \ldots, x_m(n)$ are the observed signals on microphones, $s_1(n), \ldots, s_d(n)$ are the unknown audio signals, and h_{ij} are source-sensor impulse responses of length M_{ij}.

Since most of ICA methods were originally developed to separate instantaneous mixtures, i.e. when $M_{ij} = 0, \forall i, j$, they cannot be directly applied to the convolutive mixtures in (1). To transform the model to the instantaneous one, two main approaches exist. First, the signals can be transformed into the frequency-domain [6, 7], which gives a set of instantaneous mixtures, each of which correspond to one frequency bin. The other class of approaches works in the time-domain [8, 9] as well as the method considered in this paper.

The objective of BSS is to retrieve the original signals in the form

$$\widehat{s}_i(n) = \sum_{j=1}^{m} \sum_{\tau=0}^{L-1} w_{ij}(\tau) x_j(n-\tau), \qquad (2)$$

where w_{ij} are the separation filters to be estimated, and L is their length. In ICA, this means to estimate w_{ij}s such that the separated signals $\widehat{s}_1(n), \ldots, \widehat{s}_d(n)$ are as much independent as possible. However, there is ambiguity in the solution of the problem, because $\widehat{s}_i(n)$ may be arbitrarily filtered version of $s_i(n)$. Therefore, we aim at estimating responses of the original sources at microphones. For the kth original source and the ith microphone, the response is

$$s_i^k(n) = \sum_{\tau=0}^{M_{ij}} h_{ij}(\tau) s_k(n-\tau). \qquad (3)$$

2. Time-domain Audio BSS

In this section, we briefly describe the audio BSS method from [1] that is subject to our further improvement. The algorithm consists of three main steps.

In the first step, it applies an appropriate ICA method to the signal subspace spanned by

$$\mathbf{x}(n) = [x_1(n), x_1(n-1), \ldots, x_1(n-L+1), \\ x_2(n), \ldots \quad \ldots, x_m(n-L+1)]^T \qquad (4)$$

giving the de-mixing matrix $\mathbf{W}$. The rows of $\mathbf{W}$ can be seen as $m \cdot L$ different MISO filters of length L [6], whose outputs $\mathbf{c}(n) \stackrel{\text{def.}}{=} \mathbf{W}\mathbf{x}(n)$ are independent.

Thanks to the independence, the signals $\mathbf{c}(n)$, from here referred to as *(independent) components*, should contain contributions of one original source only. Therefore, in the second step, components originating from the same original source are grouped together, which is done via clustering them subject to a similarity measure. After the clustering, reconstruction weights are determined and applied in the third (reconstruction) step. The clustering and the choice of the weights are subject to our novel proposal, which will be discussed in details in the following section.

In the reconstruction step, components of each cluster are transformed into the responses (3) in the following way: The signals $\mathbf{x}(n)$ are reconstructed using reconstruction weights λ determined for the kth cluster as

$$\mathbf{x}^k(n) = \mathbf{W}^{-1} \mathrm{diag}[\lambda_{k1}, \ldots, \lambda_{k(mL)}] \, \mathbf{W}\mathbf{x}(n). \qquad (5)$$

Then, the response of the kth source on the ith microphone is obtained as

$$\widehat{s}_i^k(n) = \sum_{p=1}^{L} x^k_{(i-1)L+p}(n+p-1), \tag{6}$$

where $x_j^k(n)$ denotes the jth element of $\mathbf{x}^k(n)$.

Eventually, the algorithm applies the simple delay-and-sum beamformer to the responses $\widehat{s}_i^k(n)$, $i = 1, \ldots, m$, to form the kth original source estimate.

3. Improvement of the Clustering Step

This section describes our novel improvement of the clustering step of the above audio BSS algorithm. The goal of this step is to cluster the components $\mathbf{c}(n)$ subject to a similarity measure that reflects their affiliation to the same original source. In [1, 10], the similarity between the ith and the jth component, respectively, denoted by $c_i(n)$ and $c_j(n)$, is defined as

$$\mathbf{S}_{ij} = \hat{\mathrm{E}}[\mathbf{P}_i c_j(n)]^2 + \hat{\mathrm{E}}[\mathbf{P}_j c_i(n)]^2, \tag{7}$$

where $\mathbf{P}_i$ denotes a projector on the subspace spanned by

$$c_i(n-L+1), \ldots, c_i(n+L-1), \tag{8}$$

and $\hat{\mathrm{E}}$ denotes the sample mean operator.

In [10], the standard agglomerative hierarchical clustering (AHC) method is used. Since this produces so-called "hard" partition of $\mathbf{c}(n)$, the resulting $d \times (mL)$ *partition* matrix $\mathbf{U}$, whose elements describe affiliations of components to clusters, has elements equal either to one or zero with that $\sum_{k=1}^{d} \mathbf{U}_{kj} = 1$. This means that any independent component may contribute to one cluster (reconstructed source) only. For the reconstruction, the weights in (5) are set to λ_{kj}=$\mathbf{U}_{kj}$.

The problem with this approach is that, in practice, there is still a lot of residual interference between the components $\mathbf{c}(n)$. Therefore, the idea of "fuzzy" reconstruction is applied in [1] by allowing each component to contribute to each reconstructed source (cluster) with an appropriate weight. The clusters are still determined by AHC, but the weights λ_{kj} in (5) take nonnegative values defined as

$$\lambda_{kj} = \left(\frac{\sum_{i \in K_k, i \neq j} \mathbf{S}_{ji}}{\sum_{i \notin K_k, i \neq j} \mathbf{S}_{ji}} \right)^{\alpha}, \tag{9}$$

where K_k contains indices of components in the kth cluster, and α is an adjustable positive parameter that controls "hardness" of the weighting.

The modification that we propose in this paper consists in applying the idea of the "fuzzy" assignment already in the clustering algorithm. Therefore, we propose to replace the hard AHC clustering method used in [1, 10] by a fuzzy clustering algorithm. Specifically, we use the Relational Fuzzy c-Means algorithm (RFCM) [11], which is a version of the standard Fuzzy c-Means algorithm (FCM) [12].

In brief, the FCM algorithm seeks the optimum fuzzy partition matrix $\mathbf{U}$ of data vectors $\mathbf{y}_1, \ldots, \mathbf{y}_K$ by minimizing the objective function

$$J_f(\mathbf{U}, \mathbf{V}) = \sum_{k=1}^{c} \sum_{j=1}^{K} (\mathbf{U}_{kj})^f \mathbf{D}_{kj}, \tag{10}$$

where the exponent f is the "fuzzyfication" parameter greater than one, c is the number of clusters, $\mathbf{V} = [\mathbf{v}_1 \ldots \mathbf{v}_c]$ is a matrix of cluster centroids, and $\mathbf{D}$ is a matrix of Euclidean distances between the centroids and the data vectors, namely, $\mathbf{D}_{kj} = \|\mathbf{v}_k - \mathbf{y}_j\|^2$. The algorithm proceeds iteratively by subsequent updating of the cluster centroids and the partition matrix $\mathbf{U}$ until convergence is achieved.

The RFCM algorithm is designed to allow application of FCM on data that cannot be represented by vectors $\mathbf{y}_1, \ldots, \mathbf{y}_K$. Relations between objects to be clustered are described by pairwise dissimilarities in a $K \times K$ matrix $\mathbf{R}$. In this case, cluster centroids cannot be explicitly defined nor their distances to data. RFCM avoids the problem using the fact that the distances of centroids from data can be expressed as a function of matrices $\mathbf{U}$ and $\mathbf{R}$ if the distance defined by $\mathbf{R}$ is Euclidean. Since the similarity measure in (7) needs not necessarily satisfy this assumption, a spreading transform is applied to ensure the convergence. For more details about RFCM and the spreading transform see [11].

In our application of RFCM for the clustering of independent components, the similarity matrix $\mathbf{S}$ is first transformed into the dissimilarity matrix $\mathbf{R}$ by setting $\mathbf{R}_{ij} = 1/(a + \mathbf{S}_{ij})$, $i \neq j$, $\mathbf{R}_{ii} = 0$, where a is a parameter chosen so that elements of $\mathbf{R}$ take values as uniformly spaced in $(0, 1)$ as possible. Our practical choice of a was 0.3. The fuzzyfication parameter was set to a common value $f = 2$.

For the reconstruction step in the separation algorithm, we have tried two possible utilizations of the resulting partition matrix $\mathbf{U}$ from the clustering:

- Indirect use of $\mathbf{U}$ by taking hard clustering defined by assigning each component to the cluster with the highest membership and doing the fuzzy reconstruction using (9), and
- direct use of $\mathbf{U}$ by setting

$$\lambda_{kj} = \left(\frac{\mathbf{U}_{kj}}{1 - \mathbf{U}_{kj}} \right)^{\alpha}, \tag{11}$$

where α is an adjustable positive parameter as in (9).

We found that the weights λs determined by both approaches are suitable for reconstruction in (5) as they give similar results.

4. Experimental Results

Since the source signals are estimated via linear MIMO filtering (2), the separation can be evaluated by means of the Signal-to-Interference ratio (SIR). However, this criterion does not reflect real acoustic quality of the separated signals. Therefore, in our second experiment, we test the overall quality of separated sources by recognizing them via ASR system.

On the other hand, SIR serves as a suitable criterion for determining affiliation of an independent component to a cluster. Therefore, we utilize it to define a reference method to compare the quality of partition obtained by different clustering algorithms.

SIR of the kth estimated source on the ith microphone is defined as

$$\mathrm{SIR}_i^k = \frac{\hat{\mathrm{E}}\left[\sum_{l=1}^{m} \sum_{\tau=0}^{L-1} w_{il}^k(\tau) s_l^k(n-\tau)\right]^2}{\hat{\mathrm{E}}\left[\sum_{l=1}^{m} \sum_{\tau=0}^{L-1} w_{il}^k(\tau)(x_l(n-\tau) - s_l^k(n-\tau))\right]^2} \tag{12}$$

where w_{il}^k, $l = 1, \ldots, m$, are the separating filters of the length L estimated by the algorithm. Note that the evaluation of this criterion requires knowledge of the responses (3) of the original

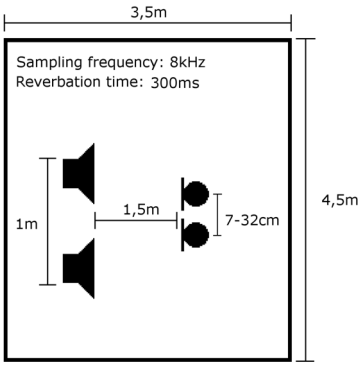


Figure 1: Conditions of our experiment.

sources on microphones. We obtain the responses by separate recordings of each source while the other sources are silent.

4.1. Comparison of clustering techniques

The goal of this experiment is to evaluate performance of the RFCM clustering approach proposed in this paper and to compare it with the agglomerative hierarchical clustering (AHC) previously used in [1, 10].

For the performance evaluation, we introduce a *reference* hard clustering that is decided based on known SIR of each independent component. Specifically, SIR of each independent component is evaluated, and the component is assigned to the source (cluster) with the highest SIR.

The experiment was performed on data that were obtained by playing two audio sources over two loudspeakers and recorded by two microphones in a room described by Figure 1. There were eight various distances between the microphones and six combinations of the following audio sources: two man voices, two woman voices, a typewriter sound, and a gaussian noise. As a whole, it gives $4 \cdot 7 \cdot 6 = 168$ different separation scenarios. For the separating filter length L, there are $2L$ independent components, which means $168 \cdot 2L$ assignment decisions.

Assuming that the reference clustering described above gives correct decisions, we compare them with the decisions obtained by the AHC method. To allow the comparison with the results of RFCM, the fuzzy clustering is reduced into a hard clustering by use of maximum-membership decision. Table 1 shows number of incorrect decisions obtained by the methods for various separating filter length L. The RFCM achieves lesser number of incorrect component assignments compared to AHC for all three filter lengths.

	L=20	L=25	L=30
AHC	1266	1529	2088
RFCM	1230	1488	1961
Total decisions	6720	8400	10080

Table 1: *Number of incorrect assignments of AHC and RFCM computed subject to the reference clustering.*

In this experiment, the computational burden of RFCM and AHC algorithms was measured in terms of the time necessary for performing all of the 168 clustering tasks. Experiment was performed in Matlab 7.2 on a PC with single core 2,6GHz processor and 1GB RAM. The results are shown in Table 2. As can be seen, RFCM is more than twice faster than AHC. Moreover, the burden of AHC grows with the number of clustered components (the algorithm needs to determine more levels of hierarchy), whereas the number of iterations needed by RFCM to converge is approximately the same for all filter lengths.

	L=20	L=25	L=30
AHC [s]	14.96	17.70	19.00
RFCM [s]	6.45	6.60	6.52

Table 2: *Time necessary for algorithms to complete all 168 clustering tasks*

4.2. Blind Speech Separation and ASR

We have conducted a large series of experiments that should prove the ability of the proposed BSS algorithm to separate speech from interfering noise. Speech signals employed in our experiment were taken from the European database of recorded broadcast news (BN) that was collected within COST278 action in 2003 [13] and later also in 2005. The database contains complete records of TV news in 10 European languages. We used its Czech part in order to be able to perform speech recognition by means of our Czech LVCSR system [14]. The test set included 653 utterances taken from 9 Czech BN shows. They represented a large variety of spoken data, from clear studio speech of professional speakers to spontaneous utterances recorded in very noisy conditions. The total number of words in the test set was 10,322. When transcribed by our ASR system, the overall recognition rate for this set was 81.02%.

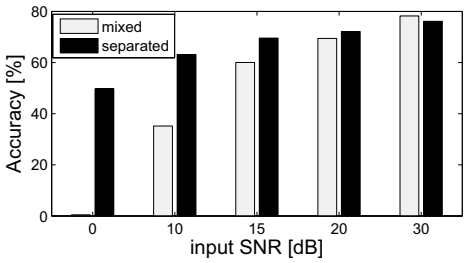
Two types of interference were simulated: 1) a test utterance mixed with a Gaussian noise and 2) a test utterance mixed with another utterance (from the same data set). Each mixture obeys the convolutive mixing model (1) where two microphones are considered ($m = 2$). The convolving filters were randomly generated for each mixture so that $h_{ij}(\tau)$ has the Gaussian distribution with zero mean and variance τ^{-2}. The length of the filters was 2000 taps, which corresponds to 125ms of reverberation time in 16kHz sampling. Each mixture was separated by a filter of length $L = 20$ that was estimated by the BSS algorithm using only segment of 6000 samples where both the speech and the interfering signal were active.

The mixed utterances as well as the separated ones were submitted to the recognizer and the recognition performance was evaluated in terms of accuracy defined as

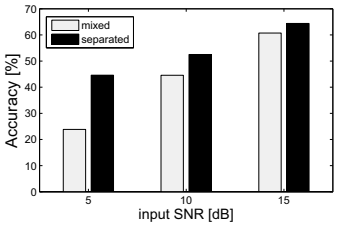
$$100 \cdot (C - D - I - S)/C,$$

which is computed via comparison of a reference text with the recognized one. Here, C is the number of words in the reference text, D is the number of deletions, I is the number of insertions, and S denotes the number of substitutions.

In two series of experiments we simulated different mixing conditions (with regards to SNR). In the first one, we added Gaussian noise of varying power to each test utterance, in the second one, interfering speech with varying gain was added. In each experiment, an average SNR value was calculated and a recognition accuracy value was obtained from the ASR system. After that, we used the proposed BSS method to separate the utterances back. They were sent to the ASR system again to evaluate the accuracy achieved for the enhanced signals. The level of enhancement received for selected SNR values can be



(a)



(b)

Figure 2: Accuracy achieved by recognizing mixed utterances with (a) a gaussian stationary noise and (b) an interfering utterance.

observed from diagrams in Fig. 2. If we focus, for example, just on 10 dB SNR case, we can see that adding gaussian noise reduced recognition accuracy (from the original value 81.02 %) to 35.16 %, but after applying the BSS method, the score was increased to 63.13 %. In case of added speech and the same 10 dB SNR situation, the method helped to enhance recognition accuracy from 44.57 % to 52.52 %.

In general, the results confirm significant improvement of the recognition rate after applying the separation algorithm, especially, in cases of low SNR (≈0dB). A slight decline of accuracy was observed for very high SNR (30dB), which is caused by some distortion of the separated signal due to higher value of the parameter α in (9). For instance, by taking $\alpha = 1$ instead of 2, the accuracy grows from 76.13% to 77.73%.

5. Conclusions

An improved method for blind source separation in time-domain was proposed. Improvement consists in utilization of the RFCM algorithm in the clustering step of the method. The RFCM is shown to be faster and more efficient in grouping of independent components compared to AHC hard clustering used in previously proposed version of the algorithm. Objective evaluation of the method's performance is given by experiment with ASR system. Here the utterances taken from the Czech part of European broadcast news database were mixed with noise and subsequently separated by the proposed method. Both mixed and separated data were recognized by ASR system. The comparison of results shows, that proposed method is effective preprocessing technique and can notably improve accuracy of ASR system assuming that multiple recordings of noisy utterance are available.

6. Acknowledgements

The work was supported by the Czech Science Foundation in projects no. 102/08/0707 and 102/07/P384.

7. References

[1] Koldovský, Z., Tichavský, P., "Time domain blind audio source separation using advanced component clustering and reconstruction," *Proc. of Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA 2008)*, Trento, Italy, pp. 216-219, May 2008.(patent pending)

[2] Hyvärinen, A., Karhunen, J., Oja, E., *Independent component analysis*, Wiley-Interscience , New York, 2001.

[3] Pham, D-T., Cardoso, J-F., "Blind separation of instaneous mixtures of non-stationary sources," *IEEE Trans. Signal Processing*, vol. 49, no. 9, pp. 1837-1848, 2001.

[4] S. Ikeda and N. Murata, "A method of ICA in time-frequency domain," in *Proc. Int. Workshop Ind. Compon. Anal. Blind Signal Separation (ICA99)*, pp. 365371, Jan. 1999.

[5] Tichavský P., Yeredor A., Nielsen J., "A fast approximate joint diagonalization algorithm using a criterion with a block diagonal weight matrix," *Proc. of IEEE Int. Conf. Acoust, Speech, Signal Processing*, pp. 3321-3324, Las Vegas, USA, March 2008.

[6] Buchner, H., Aichner, R., and Kellermann, W.: "A Generalization of Blind Source Separation Algorithms for Convolutive Mixtures Based on Second-Order Statistics," *IEEE Trans. on Speech and Audio Processing*, Vol. 13, No. 1, pp. 120-134, January 2005.

[7] Mori, Y., Saruwatari, H., Takatani, T., Ukai, S., Shikano, K., Hiekata, T., Ikeda, Y., Hashimoto, H., and Morita, T.: "Blind Separation of Acoustic Signals Combining SIMOModel-Based Independent Component Analysis and Binary Masking," *EURASIP Journal on Applied Signal Processing*, Vol. 2006, Article ID 34970, 17 pages, 2006.

[8] Thomas, J., Deville, Y., Hosseini, S.: "Time-domain Fast Fixed-Point Algorithms for Convolutive ICA," *IEEE Signal Processing Letters*, Vol. 13, No.4, pp. 228-231, 2006.

[9] S. Douglas, M. Gupta, H. Sawada, and S. Makino, "Spatio-temporal FastICA algorithms for the blind separation of convolutive mixtures," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1511-1520, July 2007.

[10] Koldovský, Z., Tichavský, P., "Time-domain blind audio source separation using advanced ICA methods," *Interspeech 2007: The 8th Annual Conference of the International Speech Communication Association*, Antwerp, Belgium, pp. 846-849, 2007.

[11] Hathaway R-J., Bezdek J-C., Davenport J. W., "On relational data versions of c-means algorithm," *Elsevier - Pattern recognition letters*, no. 17, pp. 607-612, 1996.

[12] Bezdek, J.C., "Fuzzy Mathematics in Pattern Classification," *PhD Thesis at Cornell University*, Ithaca, NY, 1973.

[13] Vandecatseye, A., et al, "The COST278 pan-European Broadcast News Database," *Proc. of 4th International Conference on Language Resources and Evaluation*, Lisboa, Portugal, May 2004.

[14] Zdansky, J., Cerva, P., Silovsky, J., Nouza, J.: "Acoustic Model Management Strategies for Improved Automatic Transcription of Broadcast Programs," *Proc. of SPECOM'2007*, Moscow, Russia, pp. 503-508, Oct. 2007.