EVALUATION OF BLIND SIGNAL SEPARATION METHODS

Daniël Schobben
Eindhoven University of Technology
Electrical Engineering Department
Building EH 5.29, P.O. BOX 513
5600 MB Eindhoven, Netherlands
ds@altavista.net
http://www.esp.ele.tue.nl/~daniels

Kari Torkkola
Motorola
Phoenix Corporate Research Labs
2100 E. Elliot Rd., MD EL508
Tempe, AZ 85284, USA
a540aa@email.mot.com
http://members.home.net/torkkola

Paris Smaragdis
MIT Media Lab
Rm E15-401C
20 Ames Street
Cambridge, MA 02139, USA
paris@media.mit.edu
http://sound.media.mit.edu/~paris

ABSTRACT

Recently, many new Blind Signal Separation (BSS) algorithms have been introduced. Authors evaluate the performance of their algorithms in various ways. Among these are speech recognition rates, plots of separated signals, plots of cascaded mixing/unmixing impulse responses and signal to noise ratios. Clearly, not all of these methods give a good reflection of the performance of these algorithms. Moreover, since the evaluation is done using different measures and different data, results cannot be compared. As a solution we provide a unified methodology of evaluating BSS algorithms along with providing data online such that researches can compare their results. We will focus on acoustical applications, but many of the remarks apply to other BSS application areas as well.

1. INTRODUCTION

Blind Signal Separation (BSS) is the process that aims at separating a number of source signals from observed mixtures of those sources [1, 2, 3, 4, 5]. For example, in an acoustical application, these mixtures might originate from a recording using multiple microphones. The term "blind" comes from the very weak assumptions made about the mixing and the sources. Typically only independence of the sources is assumed possibly together with some knowledge of the probability density functions of the sources.

It seems that BSS can have a large number of applications in the audio realm, especially in the area of signal enhancement by removing undesired source components from a desired signal [6, 7, 8, 9, 10]. Thus, this area has recently received a lot of attention. However, currently it is not possible to compare different algorithms reliably as every research paper seems to measure a different aspect of the performance using a different dataset.

The purpose of this paper is to remedy this problem by discussing what is needed for BSS performance evaluation. Obviously there is no single perfect measure of goodness since there is no single definition of the problem (ICA, separation, deconvolution). In addition to merely evaluating the success of a separation algorithm, a more ambitious problem is to construct a set of tests of variable difficulty examining a set of distinct properties that would provide valuable information about the weaknesses of the algorithm. This implies full control over the separation problem which in turn implies the need for synthetic test cases. Synthetic cases can be used to examine algorithm performance in trivial up to ill-conditioned cases thus rating separation ability accurately. By having control over the type of the sources we can also see the effect that they might have on algorithm performance.

However, synthetic cases fail to capture some elements of the real world. There is a certain level of complexity in the real world which we cannot confidently reproduce by synthetic means. The statistics of room responses, the dynamic quality of the convolution (even at seemingly static cases), factors such as the physical presence of the sources in the environment and the particular patterns of background noise, are hard to reproduce but present additional complications worth examining.

Thus we propose a suite of test cases divided into two main categories:

1. A number of controllable synthetic separation problems will be provided. They will test the limits of algorithms covering a wide range of attributes. As the sources are available performance measurement is straightforward.

2. With real world recordings clean sources are not available and measuring the separation quality is often difficult. We provide a test methodology that provides the best of both worlds: the realism of true audio recordings in real environments, and the ability to accurately measure the separation performance on these recordings. This

comes from recording each source *separately* in a real environment as described later.

Providing methodologies together with data sets makes it possible for researches to compare their algorithms in a more objective way.

The rest of this paper is organized as follows. First, we will discuss what aspects of BSS tasks could be measured and controlled to characterize BSS algorithms. We will choose a few most important ones for synthetic test cases. Next, we will discuss possible measures of performance together with the recording setup that is used for the evaluation of BSS systems. This is important since the recording setup determines what data is available for the evaluation of the BSS algorithm. We will describe a setup that combines realism with easy performance measurement. Finally, we will discuss the chosen test cases which are made available online for downloading.

2. WHAT IS DIFFICULT IN BSS?

Before starting to discuss measures that indicate the degree of separation achieved we will discuss what conditions could increase the difficulty of a BSS task. These conditions will thus be candidates for parameters to vary when constructing the test cases.

Convolutive mixing of the sources is inherent in almost all imaginable audio and acoustic BSS applications. In addition, we also enumerate some aspects that are related to instantaneous mixing. As the whole paper, the enumerated conditions are geared towards audio situations:

1. The closer the mixing is to a singular matrix the harder the separation task is for algorithms that do not exhibit the equivariant behaviour [1, 3]. In the presence of noise the task becomes harder also for equivariant algorithms. The level of difficulty can be controlled by adjusting the eigenvalue spread of the mixing filter matrix [11].

2. There is a continuum from instantaneous mixing to delayed mixing, i.e. convolutive mixing with only one nonzero coefficient per filter. This can be used to measure the ability of an algorithm to deal with simple convolutive mixing.

3. There is also a continuum from delayed mixing to real world convolutive mixing, which can be explored by changing the sparseness and the duration of the mixing filters. This, tested, can rate an algorithm's ability to deal with increasingly complicated mixing filters.

   In real recordings these aspects can be controlled to some extent by changing the positions of the microphones and sources. The easiest cases are in general when the mixing matrix has strong direct paths with little crosstalk; i.e. every source is close to it's microphone. Also the acoustical characteristics of the recording room can be controlled (anechoic vs. hard walled chamber). Introducing more reverberation makes the separation task more difficult in general.

4. In any kind of a mixing situation the probability density functions (pdf) of the sources have an effect. Usually the closer they are to Gaussians, the harder the separation gets.

5. The spectra of the sources may vary from narrowband to wideband which can have great influence on the performance of the algorithm. Tests should include sounds of both classes since some algorithms might rely on these qualities.

6. Some algorithms make use of the *difference* of the spectra of different source signals. Therefore it is useful to include test cases with distinct source spectra and test cases with similar source spectra.

7. Also in any kind of mixing the available amount of data needed to successfully learn to separate a *static* mixing situation characterizes how well the algorithm might perform in *dynamic* mixing circumstances. There is a continuum from static mixing to rapidly varying mixing. This can be used to vary the level of difficulty when testing an algorithm's tracking capabilities. When there is no comprehensive data set available with dynamic mixtures tracking capabilities can be judged from the convergence of the algorithm on static mixtures.

8. The ability to deal with silences is also needed, at least for static algorithms. Sections of silence from a source should not cause the algorithm to diverge. For example a case with a speaker with background noise little sections of silence should not cause a wildly different estimate so that reconvergence is necessary when the speaker appears again.

9. Increasing the number of sources together with the number of mixtures increases the degree of difficulty significantly. For example, algorithms that work well in the 2-by-2 case might fail miserably in the 4-by-4 case. At the limit of convolved unmixing we have a 1-by-1 case which corresponds to blind deconvolution.

10. Keeping the number of sources fixed but varying the number of available mixtures can greatly influence the behaviour of the algorithm. In general, at least the same number of mixtures as the number of sources is required. If there is further information available lesser number might suffice. By using more mixtures than there are sources, the capabilities of the algorithm to tolerate noise or to improve the separation performance could be characterized.

11. The amount and the quality of noise in the mixtures can be controlled using:

    (a) A single noise signal independent of all sources mixed to each sensor signal.

    (b) Different noise components, independent of all sources and each other, mixed to each sensor signal.

    (c) Similarity of the noise pdf/spectrum to the source signals.

Together, if all these aspects could be characterized, a fairly complete picture of the capabilities of an algorithm could be obtained. Synthetic test cases that cover these aspects will be introduced and discussed in Section 4.

3. HOW TO MEASURE? - RECORDING SETUPS AND PERFORMANCE MEASURES

In this section we will discuss how to evaluate the goodness of the actual separation on the basis of test data. In addition to separation, BSS algorithms can be characterized by distortion, i.e., how the original signals are distorted from how each microphone would observe them in the absence of other sources.

It appears that there are three fundamentally different methods to evaluate separation depending on what data is available.

The first one is based on the impulse responses of the mixing channels and the separation filters. By convolving the mixing and separation systems, it is simple to measure how far (in dB) the resulting filters are from a scaled unit response. As the mixing channels are required this method only applies to the synthetic case. Although this method is well defined it might not accurately evaluate the separation of the signals. For example, when the sources have low energy contents for certain frequency intervals the BSS algorithm might fail in finding an unmixing system that achieves separation for those frequencies. This does not affect the quality of separation as there is almost no frequency content in the signals for these frequencies.

The second one is based on the test signals themselves. For each separated signal the residuals of the other sources are compared against the desired source. Ideally these residuals should equal zero. Note that to be able to do this for a real recording, static mixtures are required, in which only one source is active at a time, together with a labeling that indicates these locations in the test mixtures.

A third way is to directly evaluate the independence of the separated outputs. It is difficult however to come up with a measure of independence that can be estimated accurately and gives a clear indication of quality of separation.

In the following subsection we will describe a recording setup that enables evaluation of real recordings using the second method mentioned above and overcoming its limitations.

3.1. Recording Setup

Consider the mixing/unmixing system in Figure 1. In this system the source signals $s_1 \dots s_J$ are filtered by the multi-channel acoustical transfer function H yielding the microphone signals $x_1 \dots x_J$. It is assumed that the number of sources equals the number of microphones. For synthetic cases the source signals are filtered by the premeasured multi-channel acoustical transfer function H. In that case the unmixing system w can be evaluated using the known mixing system H and the unmixed signals y_i can be judged using the known source signals s_i. This makes it possible to evaluate distortion and separation.

When real recordings are used however, the only available data are the microphone signals x_i, $i = 1 \dots J$. Both the separation and the distortion that is introduced by the unmixing system cannot be determined directly from the microphone signals.

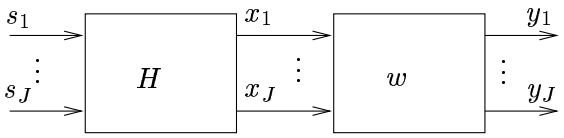

Figure 1: Cascaded mixing/unmixing system

The following approach combines the realism of true audio recordings in real environments, and the ability to accurately measure the separation performance on these recordings. In this approach multi-channel recordings are made in a room when only one of the sources is active at a time. These recorded signals are

denoted x_{i,s_j}, i.e. the contribution of the j^{th} source to the i^{th} microphone. The mixed data is obtained by adding these independent contributions for all sources, i.e. $x_i = \sum_j x_{i,s_j}$. Now the x_i represent the microphone signals that would have been obtained if all sources were active simultaneously. This is justified for acoustical applications, since sound waves are additive too.

Note that all speakers must be present during the recordings, even if they are silent, as the room acoustics are influenced by their presence. The approach is therefore limited to the recording of non-moving sources, as the source movements cannot be reproduced exactly in general. Using this approach, multi-channel recordings are as realistic as they can be without losing information that is required for the evaluation. A method for measuring the quality of separation and measuring the distortion due to the BSS algorithm using these recorded signals is described in the next subsection.

3.2. Performance Measures

3.2.1. Distortion

The distortion of the j^{th} separated output can be defined as

$$D_j = 10\log\left(\frac{E\{(x_{j,s_j} - \alpha_j y_j)^2\}}{E\{(x_{j,s_j})^2\}}\right) \quad (1)$$

with $\alpha_j = E\{x_{j,s_j}^2\}/E\{y_j^2\}$. The separated signal indices are chosen such that y_j corresponds to the j^{th} source and $E\{.\}$ denotes the expectation operator. In this definition, the separation results are not distorted when they are equal (up to a scaling factor α_j) to x_{j,s_j}, i.e. the contribution to microphone j of the j^{th} source alone. The permutation and scaling of the signals do not affect the distortion and are therefore left out of the definition.

When synthetic mixtures are used this approach can be followed too as the x_{i,s_j} can be calculated from the mixing system and the sources s_j so that the same approach can be followed.

A more detailed impression of the distortion can be obtained by plotting it as a function of frequency using a Short Time Fourier Transform (STFT)

$$D_j(\omega) = 10\log\left(\frac{E\{(\text{STFT}\{x_{j,s_j} - \alpha_j y_j\})^2\}}{E\{(x_{j,s_j})^2\}}\right) \quad (2)$$

with again $E\{.\}$ expectation over time. Matlab code to generate such plots has been made available online[1].

[1] http://www.ele.tue.nl/ica99

The baseline for distortion is the original microphone signal. Whatever happens to the signal from the source to the microphone cannot be determined as the actual sources are not available. This distortion measure is thus biased to favor methods that do not perform deconvolution in addition to separation as any deconvolution would be observed as distortion.

3.2.2. Separation

The quality of separation of the j^{th} separated output can be defined as

$$S_j = 10\log\left(\frac{E\{(y_{j,s_j})^2\}}{E\{(\sum_{i\neq j} y_{j,s_i})^2\}}\right) \quad (3)$$

with y_{j,s_i} the j^{th} output of the cascaded mixing/unmixing system when only s_i is active. Other definitions of quality of separation involving the mixing/unmixing system are less suited since BSS is about signal separation and not about system identification. The separation can also be plotted as a function of the frequency using

$$S_j(\omega) = 10\log\left(\frac{E\{(\text{STFT}\{y_{j,s_j}\})^2\}}{E\{(\text{STFT}\{\sum_{i\neq j} y_{j,s_i}\})^2\}}\right) \quad (4)$$

Matlab code for the evaluation of separation has also been made available online[1].

4. DATA SETS

4.1. Synthetic Tests

Data has been made available[1] which can be used for the evaluation of BSS algorithms. This suite includes synthetic data and non-synthetic data. Both subsets include the same original sources, mixed in synthetic and real environments respectively.

4.1.1. Source Signals

The set of sources includes, various speech phrases, music passages, environmental sounds and synthetic tones.

The speech sources consist of the same set of sentences read by different speakers. The sentences include various ratios of vowel/consonants so as to have some elementary indication of bandwidth. Given that several sentences will be read by the same speakers we can use that to provide mixtures of these, on which heuristic assumptions will be harder because of source similarity.

The music sources consist of a set of recordings of various characteristics. We include music passages of

various spectral shapes and centroids to measure the dependency of an algorithm on these characteristics of sound, as well as its behaviour with respect to the relation of these characteristics across sources i.e. a narrowband and a wideband source, a high and a low centroid source etc. There are also music passages featuring wide dynamic changes to test how well an algorithm can track sources that suddenly fade out or even disappear.

Environmental sounds will include common types of background noise, mostly to provide an indication of performance for the case of speech/background noise mixing. The examples include street noise as a full-band signal which can completely cover the bandwidth of speech and 'hide' it very effectively. Certain sounds which because of their sparse and self similar character will be hard to eliminate are also included (e.g. bouncing ball). Also a few relatively narrowband machinery sounds will be included which exhibit a variety of spectral centroids.

Finally the synthetic sounds will provide basic tests on the influence of bandwidth, PDFs, frequency variance, spectral centroids and interruptions. The set consists of a sine wave, a square wave, a sawtooth wave, Gaussian noise and Cauchy noise. Where applicable the frequency of the sound can change so as not to provide a completely stationary source. The same waveforms are also provided with random interruptions during their progress to test how well an algorithm can track disappearing sources. The evaluation of these cases will be easier since plots of the outputs can be provided and intuitive observations on the performance of the algorithms can be made out of these.

4.1.2. Mixing

For the synthetic tests the above sources were mixed in arbitrary groups using a set of different mixing situations, including instantaneous mixing matrices, sparse convolutive mixing matrices, dense convolutive mixing matrices and estimates of real world mixing matrices.

The set of instantaneous mixing matrices includes both well and ill conditioned problems. In addition some cases are contaminated with added noise in the sources. This set can measure the speed and quality of convergence as well as the accuracy of an algorithm, under both clean and degenerate conditions. The order of the matrices is varied to test how well an algorithm can deal with an arbitrary number of sources.

The set of sparse convolutive matrices includes a few mixing matrices spanning the range between simple delayed mixing to more complicated filters derived out of theoretical analysis of room reflections. This test will rate how algorithms will perform with convolutive mixtures of increasing complexity.

For the dense convolutive mixing matrices, the set contains cases of dense convolutive matrices whose filters are obtained from various kinds of manipulated noise sequences. Real room impulse responses are very similar to exponentially decaying noise with exponential or Cauchy distribution, so such series are used as filters. The parameters of the series comprising the delays are tuned to various levels to control complexity. Fast decay and high kurtosis are the simplest cases since they produce short and sparse filters, whereas longer decay times and lower kurtosis will generate harder cases. We also provide an additional degree of complexity where at the simplest level there is only one filter in the mixing matrix which is dense, while the rest are just delay taps, and by progressively increasing the number of dense filters in the mixing matrix we cover all cases and reach the hardest one where all filters are dense.

Finally we have two cases which use filters measured from a real room and a dummy head used for binaural recordings. These are real, dense mixing matrices which can lead us as close as we synthetically can to the real world. They will be used to get an accurate reading of performance (since we know of the mixing matrix) before we move to the real cases.

4.2. Real Recordings as Test Cases

The real world recordings are done in two different rooms; a near anechoic room and a typical living room. Live speakers are used and their contributions are recorded independently. Music sources are reproduced using loudspeakers. The same music tracks are used as in the synthetic case. Again, all of these contributions are recorded independently so that the performance measures from subsection 3.2 can be applied. As the music signals are known, algorithms that solve BSS together with for example acoustical echo canceling can be evaluated, too. The clean speech recordings that are done in the near anechoic room are also used as source signals for the synthetic case. The transfer functions in the near anechoic room resemble a simple delayed mixing matrix which should be relatively easy to deal with for most BSS algorithms. The living room, however, corresponds to a mixing matrix with dense filters. Estimates of the room mixing matrices are measured too so that they also can be used for the synthetic cases. Also, a dummy head is used for binaural recordings. This data can be used to evaluate BSS algorithms for applications like hearing devices where the microphones are relatively small and cheap and shadowed by the head.

4.3. Evaluation Software

The following Matlab files have been made available for downloading

- An evaluation routine of the distortion introduced by the Blind Signal Separation algorithm as described in Eq. (1)
- A plot routine which plots the distortion as a function of frequency as described in Eq. (2)
- An evaluation routine for the quality of Blind Signal Separation as described in Eq. (3)
- A plot routine which plots the separation as a function of frequency as described in Eq. (4)
- A utility that reads multiple wavfiles that contain a multitrack recording

5. CONCLUSIONS

In this paper, performance measures for BSS algorithms are presented together with a data set of real world signals. Authors in the field of BSS are encouraged to try their algorithms on this data and evaluate their algorithms in the same way such that results can be compared.

6. FUTURE WORK

New tracks will be recorded to cover a wider range of source types and mixing systems.

Since there will always be a limited amount of test cases as prerecorded signals, a statistically more reliable view of the performance of an algorithm could be obtained by generating the data according to a speech-like or a music-like distribution, and by doing Monte Carlo experiments over a large number of different realizations of the data.

Also, the performance measures can be improved by incorporating perceptual measures, for example.

Therefore, the benchmark site will be under construction to incorporate these features.

7. ACKNOWLEDGEMENTS

The authors would like to thank Russ Lambert and Lucas Parra for valuable suggestions concerning the test cases, Alex Westner for providing the real room filters, and Keith Martin with Bill Gardner for providing the head related transfer functions as measured from a dummy head.

8. REFERENCES

[1] S. Amari, A. Cichocki, and H. H. Yang. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems 8*. MIT Press, 1996.

[2] A.J. Bell and T.J. Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[3] J-F. Cardoso and B. Laheld. Equivariant adaptive source separation. *IEEE Transactions on Signal Processing*, 44(12):3017–3030, December 1996.

[4] P. Comon. Independent component analysis – a new concept? *Signal Processing*, 36(3):287–314, 1994.

[5] Jeanny Herault and Christian Jutten. Space or time adaptive signal processing by neural network models. In *Neural Networks for Computing, AIP Conf. Proc.*, volume 151, pages 206–211, Snowbird, UT, USA, 1986.

[6] Mark Girolami. Noise reduction and speech enhancement via temporal anti-Hebbian learning. In *Proc. ICASSP*, Seattle, WA, USA, May 12-15 1998.

[7] T.-W. Lee, A.J. Bell, and Reinhold Orglmeister. Blind source separation of real world signals. In *Proc. International Conference on Neural Networks (ICNN'97)*, Houston, TX, June 9-12 1997.

[8] Henrik Sahlin and Holger Broman. Signal separation applied to real world signals. In *Proceedings of 1997 Int. Workshop on Acoustic Echo and Noise Control (IWAENC97)*, London UK, September 11-12 1997.

[9] D.W.E. Schobben and P.C.W. Sommen. Transparent communication. In *Proceedings IEEE Benelux Signal Processing Chapter Symposium*, pages 171–174, Leuven, Belgium, March 26-27 1998.

[10] Kuan-Chieh Yen and Yunxin Zhao. Robust automatic speech recognition using a multi-channel signal separation front end. In *Proc. 4th Int. Conf. on Spoken Language Processing (ICSLP'96)*, Philadelphia, PA, October 1996.

[11] Russell H. Lambert. *Multichannel blind deconvolution: FIR matrix algebra and separation of multipath mixtures.* PhD dissertation, University of Southen California, Department of Electrical Engineering, May 1996.