INTRODUCING A SIMPLE FUSION FRAMEWORK FOR AUDIO SOURCE SEPARATION

Xabier Jaureguiberry[*]*, Gaël Richard Pierre Leveau, Romain Hennequin Emmanuel Vincent*

Institut Mines-Télécom
Télécom ParisTech, CNRS LTCI
37-39, rue Dareau 75014 Paris, France

Audionamix
114, avenue de Flandre
75019 Paris, France

Inria
615 rue du Jardin Botanique
54600 Villers-lès-Nancy, France

ABSTRACT

We propose in this paper a simple fusion framework for underdetermined audio source separation. This framework can be applied to a wide variety of source separation algorithms providing that they estimate time-frequency masks. Fusion principles have been successfully implemented for classification tasks. Although it is similar to classification, audio source separation does not usually take advantage of such principles. We thus introduce some general fusion rules inspired by classification and we evaluate them in the context of voice extraction. Experimental results are promising as our proposed fusion rule can improve separation results up to 1 dB in SDR.

Index Terms— audio source separation, data fusion, non-negative matrix factorization, machine learning

1. INTRODUCTION

Underdetermined audio source separation is an important field of research which has reached sufficient quality to challenge a variety of industrial issues. Numerous algorithms and models have been proposed in the literature. They exploit different features, such as sparsity [1, 2], morphological characteristics [3, 4, 5] or perceptual grouping [6], in order to segregate each source from the others. Efficient methods have usually been developed for a specific source separation problem and their performance also depends on the tuning of the models and the algorithms.

To a certain extent, audio source separation can be seen as a classification problem: while a classifier is used to assign an object to a class, a separator, *i.e.*, an algorithm for source separation, assigns sound events such as spectral shapes or time-frequency bins to a source amongst others. Moreover, as for source separation algorithms, classifiers are often designed and tuned for a specific task.Yet the diversity of results given by different classifiers on a same task has led researchers to combine classifiers [7] and to develop the principles of *data fusion* for classification [8]. In particular, *late* fusion approaches focus on the decision level: they combine the decisions taken by several classifiers in order to formulate a new decision which is expected to be more reliable. Each classifier having its strengths and weaknesses, *late* fusion tends to reinforce decisions by exploiting complementary and contradictory information given by distinct classifiers.

Although it is similar to classification, audio source separation barely exploits fusion principles. Kirbiz *et al.* implemented in [9] a form of *late* fusion: NMF (Non-negative Matrix Factorization) based separation is applied *in parallel* to different time-frequency resolution spectrograms and the resulting time-domain source estimates are combined through an adaptive filter bank. Their paper thus applies *data fusion* principles to an audio source separation problem thanks to the design of a specific algorithm. In image processing, Meganem *et al.* also propose in [10] to compute several source estimates thanks to different analysis parameters and to use a correlation measure in order to select the best estimate.

We introduce in this article some simple *late* fusion approaches applied to multiple parallel separation algorithms in a way similar to classifier fusion. We focus on algorithms based on time-frequency masking so that fusion consists of combining estimated masks. Our experiments show that the proposed approach can improve the SDR (Signal to Distortion Ratio) of separated sources up to 1 dB which is very significant in comparison with improvements achieved in the literature in the last few years [11]. Our approach differs from [9] in that we propose a very simple framework, easy to implement for any source separation task and which could be applied to a large variety of source separation algorithms without redesigning them. Our method also differs from [10] in that we derive a *soft* combination of separation results instead of proceeding to the *hard* selection of one result amongst the others. The fusion rules we propose will be introduced in Sec. 2. The estimation of fusion coefficients will be discussed in Sec. 3. We will then present our experimental framework in Sec. 4 before analyzing the results of our experiments in Sec. 5. Finally we conclude and suggest some future works in Sec. 6.

[*]This work was partly supported under the research programme Quaero, funded by OSEO, the French State agency for innovation.

2. FUSION IN SOURCE SEPARATION

We introduce in this section the general fusion framework we developed for underdetermined audio source separation. According to the terminology of data fusion, our work focuses on the decision level, meaning that the fusion step occurs right after the separation step. The literature also refers to such a framework as *late* fusion. Moreover, we limit our study to audio source separation algorithms based on time-frequency masking in order to reinforce the comparison with data fusion for classification.

2.1. Separation step

Let us consider a mono signal denoted by $\mathbf{x}(t)$. We assume that this signal is the mixture of N sources $\mathbf{x_n}(t)$ so that

$$\mathbf{x}(t) = \sum_{n=1}^{N} \mathbf{x_n}(t) \tag{1}$$

where t is the time index and n the source index.

We define a separator $\mathcal{S}_k$ as an algorithm for source separation together with the corresponding models, hyperparameters and initializations. This separator provides an estimate $\widetilde{\mathbf{x}}_\mathbf{n}^\mathbf{k}(t)$ of each source. Algorithms based on time-frequency masking firstly compute a time-frequency representation of the mixture signal that we will denote as $\mathbf{X}(f,t)$ where f is the frequency index. Most often, the Short Time Fourier Transform (STFT) is employed. A time-frequency mask $\widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}(f,t)$ is then estimated for each source n so that the time-frequency representation of the estimated source $\widetilde{\mathbf{x}}_\mathbf{n}^\mathbf{k}$ is given by

$$\widetilde{\mathbf{X}}_\mathbf{n}^\mathbf{k} = \widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k} \circ \mathbf{X} \tag{2}$$

where the operator $\circ$ denotes the Hadamard (element-wise) product. The elements $\widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}(f,t)$ of the masks are nonnegative and they verify $\sum_{n=1}^{N} \widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}(f,t) = 1$. Each element represents the contribution of source n in the time-frequency bin (f,t) as estimated by the separator $\mathcal{S}_k$. We refer to binary masking when the gain is constrained to be either 0 or 1 and to soft masking otherwise [12].

Once the masks have been estimated for each source, the estimated time-domain signal $\widetilde{\mathbf{x}}_\mathbf{n}^\mathbf{k}(t)$ is computed by inverting the resulting time-frequency representation $\widetilde{\mathbf{X}}_\mathbf{n}^\mathbf{k}$ in (2).

Depending on the type of sources that are involved in the mixture, different ways of estimating the masks $\widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}$ have been proposed in the literature [1, 2, 4, 5] and each one leads to different estimates of the sources. However, a given algorithm can also lead to different estimated sources depending on the tuning of the underlying models and their initialization. As a consequence, we consider in the following that *two separators $\mathcal{S}_1$ and $\mathcal{S}_2$ are distinct if they use either different source separation algorithms or the same algorithm with different parameter initializations and/or tunings*.

2.2. Fusion of soft masks

When the masks are binary, the parallel with a classification problem is obvious as discussed in [13]. The goal of *late* fusion in classification is to combine the decision of several classifiers, expecting that it would improve the classification performance. In the same manner, combining several binary masks might allow the estimation of another binary mask which will improve the separation quality. According to [7], one of the simplest ways of combining multiple classifiers is to use a majority vote rule.

However, soft masking methods such as adaptive Wiener filtering are known to give slightly better separation results than binary masking approaches [14, 15]. It seems then more promising to design some simple ways of combining several separators which estimate soft masks. In this case, since $\sum_{n=1}^{N} \widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}(f,t) = 1$, an element $\widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}(f,t)$ can be considered as the probability of the time-frequency bin (f,t) being part of the source n, similarly to multiclass classification problems [16]. We thus propose to estimate a new soft mask $\widetilde{\mathbf{M}}_\mathbf{n}(f,t)$ as a simple linear combination of K different masks so that

$$\widetilde{\mathbf{M}}_\mathbf{n}(f,t) = \sum_{k=1}^{K} \alpha_k(f,t) \widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}(f,t). \tag{3}$$

In order to keep the interpretation of $\widetilde{\mathbf{M}}_\mathbf{n}(f,t)$ as a probability, we impose that $\forall k$ and $\forall (f,t), \alpha_k(f,t) \geq 0$ and $\sum_{k=1}^{K} \alpha_k(f,t) = 1$. According to [17], each coefficient $\alpha_k(f,t)$ should reflect the degree of confidence in the separator $\mathcal{S}_k$ depending on the time-frequency bin (f,t) under consideration. In the following, we will refer to $\{\alpha_k(f,t)\}_{k=1}^{K}$ as the set of *fusion coefficients*.

2.3. Temporal fusion

In order to evaluate the potential of the proposed fusion rule (3), we suggest that in a first approach the fusion coefficients remain independent of (f,t). As a consequence, the time-frequency representation $\widetilde{\mathbf{X}}_\mathbf{n}$ of a source after fusion can be simplified as follows

$$\begin{aligned} \widetilde{\mathbf{X}}_\mathbf{n} &= \widetilde{\mathbf{M}}_\mathbf{n} \circ \mathbf{X} = \left(\textstyle\sum_{k=1}^{K} \alpha_k \widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k}\right) \circ \mathbf{X} \\ &= \textstyle\sum_{k=1}^{K} \alpha_k \left(\widetilde{\mathbf{M}}_\mathbf{n}^\mathbf{k} \circ \mathbf{X}\right) = \sum_{k=1}^{K} \alpha_k \widetilde{\mathbf{X}}_\mathbf{n}^\mathbf{k}. \end{aligned} \tag{4}$$

By applying the inverse transform to $\widetilde{\mathbf{X}}_\mathbf{n}^\mathbf{k}$, our fusion rule can be expressed in the time domain as

$$\widetilde{\mathbf{x}}_\mathbf{n}(t) = \sum_{k=1}^{K} \alpha_k \widetilde{\mathbf{x}}_\mathbf{n}^\mathbf{k}(t). \tag{5}$$

Our fusion rule thus becomes in this case a linear combination of the estimated time-domain signals, which we will refer to as *temporal fusion by linear combination*.

3. ESTIMATION OF THE FUSION COEFFICIENTS

As we have already highlighted, a fusion coefficient somehow reflects the degree of confidence in its corresponding separator. In audio source separation, the quality of separation is often measured by the SDR (Signal to Distortion Ratio) [18] expressed in decibels (dB). For instance, the SDR of the source estimate $\widetilde{\mathbf{x}}_{\mathbf{n}}(t)$ is given by

$$\text{SDR} = 10\log_{10}\frac{\sum_t \|\mathbf{x_n}(t)\|^2}{\sum_t \|\widetilde{\mathbf{x}}_{\mathbf{n}}(t) - \mathbf{x_n}(t)\|^2} \tag{6}$$

where $\mathbf{x_n}(t)$ denotes the original source involved in (1). The computation of the SDR for each separator $\mathcal{S}_k$ could enable us to measure its performance and then to determine its corresponding fusion coefficient α_k accordingly. However, as in practice the original sources $\mathbf{x_n}(t)$ are usually unavailable, we must think of estimating the fusion coefficients without accounting on the SDR of the test mixture.

In our experimental framework, knowledge of the original sources allows us to compute the upper bound on performance of our temporal fusion rule (see Sec. 3.1) and to investigate a learning method for the fusion coefficients (see Sec. 3.2). We also propose in Sec. 3.3 a blind approach to the estimation of the fusion coefficients in which we do no take advantage of the original sources.

3.1. Oracle temporal fusion

If the original sources are known, the SDR can be computed for the estimated signals after fusion as defined in (6). In order to evaluate the potential of our temporal fusion rule, we can estimate the set of fusion coefficients which maximizes the SDR of the resulting signals $\widetilde{\mathbf{x}}_{\mathbf{n}}(t)$ after fusion. Maximizing (6) leads to a standard Quadratic Programming (QP) [19] problem under linear equality and inequality constraints:

$$\begin{array}{ll} \underset{(\alpha_1,\ldots,\alpha_k,\ldots,\alpha_K)}{\text{argmin}} & \sum_t \left\| \mathbf{x_n}(t) - \sum_{k=1}^K \alpha_k \widetilde{\mathbf{x}}^{\mathbf{k}}_{\mathbf{n}}(t) \right\|^2 \\ \text{subject to} & \left\{ \begin{array}{l} \forall k, \alpha_k \geq 0 \\ \sum_{k=1}^K \alpha_k = 1 \end{array} \right. . \end{array} \tag{7}$$

In the following, we refer to this approach as *oracle temporal fusion* as it gives the best SDR that we could expect using (5). It is to be compared to an approach by *oracle separator selection* which consists of selecting the separator which gives the best SDR for a given mixture amongst the set of K possible separators.

3.2. Learned temporal fusion

Solving the QP problem in (7) relies on the availability of the original sources that compose the test mixture. If these original sources are not available, we propose to learn the fusion coefficients from a training dataset. Let us consider that we have a training dataset composed of several mixtures and their original sources. In order to learn an average set of coefficients, we propose to concatenate all the examples of the training dataset and estimate a set of fusion coefficients on this concatenated signal thanks to the method proposed in Sec. 3.1. These learned fusion coefficients are then applied to the test mixture without knowing its original sources.

In the following, we will refer to this approach as *learned temporal fusion*. It is to be compared to an approach by *learned separator selection* which consists of selecting amongst the set of K possible separators the separator which gives the best SDR on average on the training database. This latter approach is often used in practice when the original sources of the test mixture are unknown in order to choose a suitable separator *a priori*. As such, we will consider it as the baseline in our experiments.

3.3. Temporal fusion by mean

Contrary to *oracle temporal fusion* and to *learned temporal fusion* which always require the knowledge of some original sources (respectively the original sources of the test mixture and the original sources of the training mixtures), we propose here a blind approach to our temporal fusion rule by simply taking the mean of the estimated signals, which is equivalent to set $\forall k, \alpha_k = 1/K$ in (5).

4. EXPERIMENTAL FRAMEWORK

In this section, we propose to test the above fusion framework in the context of *voice extraction*. In voice extraction, the signal to be separated is composed of a lead singing voice mixed with a musical background. The mixing model in (1) is now expressed as

$$\mathbf{x}(t) = \mathbf{v}(t) + \mathbf{m}(t) \tag{8}$$

where $\mathbf{v}(t)$ is the lead voice signal and $\mathbf{m}(t)$ the music signal.

4.1. Signal models

Following Sec. 2, we are going to estimate both the voice signal and the music signal thanks to K separators which each outputs two soft masks $\widetilde{\mathbf{M}}^{\mathbf{k}}_{\mathbf{v}}$ and $\widetilde{\mathbf{M}}^{\mathbf{k}}_{\mathbf{m}}$, respectively for the voice and the music signals. Both masks will be estimated with the model proposed by Durrieu *et al.* in [4].

The model is defined in the time-frequency domain in terms of matrices representing time-varying Power Spectral Densities (PSDs), defined as the squared magnitude of the Short Time Fourier Transform (STFT). The PSD of the observed mixture signal $\mathbf{x}(t)$ is thus represented by the matrix $|\mathbf{X}|^2(f,t) = |\text{STFT}\{\mathbf{x}\}(f,t)|^2$ of size $F \times T$, where F is the number of frequency bins and T the number of time frames.

Assuming that the two sources are statistically independent, the observed PSD matrix $|\mathbf{X}|^2$ is modeled by

$$\mathbf{D} = \mathbf{W}_M\mathbf{H}_M + [\mathbf{W}_E\mathbf{H}_E] \circ [\mathbf{W}_F\mathbf{H}_F] \tag{9}$$

where the index M refers to the musical model and the indices E and F refer respectively to the source and the filter models for the voice. $\mathbf{W}_M$ and $\mathbf{H}_M$ are respectively the *dictionary* of spectral bases (of size $F \times K_M$) and the *activation matrix* (of size $K_M \times T$) of the music model. Both these matrices are randomly initialized.$\mathbf{W}_E$ and $\mathbf{H}_E$ are respectively the dictionary and the activation matrix of the voice source part which model the PSD contributed by the glottal source and $\mathbf{W}_F$ and $\mathbf{H}_F$ the dictionary and the activation matrix of the voice filter part which model the PSD contributed by the filtering of the vocal tract. $\mathbf{W}_E$ is fixed and initialized as a collection of harmonic combs. $\mathbf{W}_F$ is fixed and initialized as a collection of smooth spectral shapes. $\mathbf{H}_E$ and $\mathbf{H}_F$ are randomly initialized.

Parameter estimation is performed by Secan iterative gradient descent method with multiplicative update rules which minimize a certain divergence d between the model $\mathbf{D}$ and the observation $|\mathbf{X}|^2$

$$\mathcal{D}\Big(|\mathbf{X}|^2 \Big| \mathbf{D}\Big) = \sum_{f=1}^{F}\sum_{t=1}^{T} d\left(|\mathbf{X}|^2(f,t) \Big| \mathbf{D}(f,t)\right). \quad (10)$$

Here, we use the Itakura-Saito (IS) divergence:

$$d_{IS}(x|y) = \frac{x}{y} - \log\frac{x}{y} - 1. \quad (11)$$

Once the model parameters are estimated, the following soft Wiener masks are computed:

$$\begin{cases} \widetilde{\mathbf{M}}_{\mathbf{v}} = ([\mathbf{W}_E\mathbf{H}_E] \circ [\mathbf{W}_F\mathbf{H}_F])./\mathbf{D} \\ \widetilde{\mathbf{M}}_{\mathbf{m}} = (\mathbf{W}_M\mathbf{H}_M)./\mathbf{D}. \end{cases} \quad (12)$$

The time-domain signals $\widetilde{\mathbf{v}}(t)$ and $\widetilde{\mathbf{m}}(t)$ are then computed by inverse STFT of the estimated PSDs, respectively $\widetilde{\mathbf{V}} = \widetilde{\mathbf{M}}_{\mathbf{v}} \circ \mathbf{X}$ and $\widetilde{\mathbf{M}} = \widetilde{\mathbf{M}}_{\mathbf{m}} \circ \mathbf{X}$.

Note that the dimensions of the model matrices as well as the random initial values given to these matrices are known to influence the quality of the separation results [20, 21].

4.2. Experimental settings

We evaluated the fusion approaches proposed in Sec. 3 on a dataset of $L = 7$ professionally-produced music recordings from diverse genres [22]. All recordings are sampled at 44.1 kHz.

In order to evaluate the temporal fusion rule, we propose three distinct experiments :

1. **fusion on the number of components of the music model:** here, a separator is defined by the number of components K_M of its music model (*i.e.*, the number of columns of the dictionary W_M) introduced in (9). Other hyperparameters are fixed to standard values [4] and are common to all separators. We defined $K = 20$ distinct separators for K_M varying from 5 to 100 components by steps of 5. Note that the model matrices are initialized in the same way for all separators using a single random seed.
2. **fusion on the initializations:** in this case, all hyperparameters (including the number of music components K_M) are the same for all separators. However, each separator has its own seed which is itself randomly selected. We thus selected $K = 20$ distinct seeds in order to define $K = 20$ different separators.
3. **fusion on both the number of components and the initializations:** here, a separator is defined by both its number of components K_M and a proper seed. As for the two previous experiments, we chose $K = 20$ distinct separators for K_M varying from 5 to 100 components by steps of 5 and for 20 different random seeds.

For each of these experiments, we tested the three different fusion approaches introduced in Sec. 3, namely *oracle temporal fusion*, *learned temporal fusion* and *fusion by mean*. We have simulated the *learned temporal fusion* approach by learning the coefficients on the concatenation of all our dataset signals except the one under consideration, in reference to a traditional *leave one out* protocol. Finally, all experiments were independently repeated ten times and the SDRs were averaged over these ten runs.

5. RESULTS

Table 1 gives the average SDR of the estimated voice source for our baseline system, the *oracle separator selection* approach and the three fusion approaches described earlier. Each result is given for the three proposed experiments, namely when the K separators have distinct number of music components K_M, distinct initializations or both distinct number of music components and initializations. Detailed results are given in Fig. 1 for the specific case of fusion on the number of music components K_M. The figure depicts the gain in SDR of the fusion approaches and of *oracle separator selection* with respect to the baseline by *learned separator selection*. All measurements were obtained with the BSS_EVAL toolbox [23]. Sound examples are also available online[1].

5.1. Influence of the hyperparameters and the initialization

The results of *oracle separator selection* and *learned separator selection* clearly show the influence of the initialization and the hyperparameters (here, the number of components of the music model) on the separation performance. Indeed, *learned separator selection* results are always outperformed by *oracle learned selection* results by 1 dB on average. Moreover, Fig. 1 also highlights that the number of music components which gives the best SDR is different for each example of our dataset. The tuning of a separator is thus important for separation quality.

[1] http://www.tsi.telecom-paristech.fr/aao/?p=829

	Variable K_M		Variable initialization		Variable K_M and initialization	
	Selection	Fusion	Selection	Fusion	Selection	Fusion
Oracle	4.36	4.69	4.31	4.56	4.49	4.80
Learned	3.36	3.86	3.43	3.87	3.25	3.82
Fusion by mean	4.00		3.98		4.01	

Table 1. Average SDR (dB) achieved by the proposed temporal fusion methods.

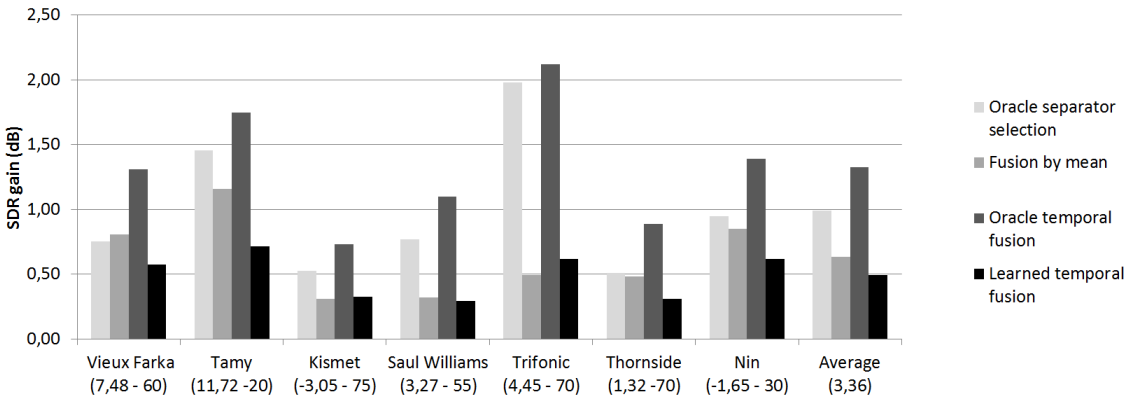

Fig. 1. SDR gain (dB) of different fusion methods compared to the baseline (*learned separator selection*), when varying the number of components K_M. The SDR of the baseline and the value of K_M which gives the highest SDR are given in parentheses below the name of each example.

5.2. Fusion by mean

The fusion method by mean, presented at the bottom of Table 1 for our three experiments, outperforms on average our baseline (*learned separator selection*). Fig. 1 also shows that this statement is verified for all examples of our dataset, when we consider separators which differ by their number of components K_M. On average, the gain is of nearly 0.7 dB which is significant according to [11]. This result highlights that the learning of an average separator to use with any test mixture is out of reach mainly because of the difficulty to build a representative database. The fusion by mean seems thus to be an interesting alternative and overcomes to a certain extent the need of choosing this number of components and the initialization of the model matrices.

5.3. Oracle and learned fusions

The *oracle temporal fusion* results given in Table 1 mainly show that the proposed fusion rule (5) reaches the expected fusion property of exceeding the performance of each individual separator involved. Indeed, we can notice that the *oracle temporal fusion* results outperform both the *oracle separator selection* and the *learned separator selection* strategies, respectively by about 0.3 dB and 1.3 dB on average. Fig. 1 confirms this assertion for all examples of our dataset.

However, *learned temporal fusion* hardly reaches the performance of *oracle temporal fusion* and is even outperformed by the fusion by mean. The learning method we propose is thus perfectible. For instance, we could expect better results if we were able to use a more representative database or to adapt the learning step to the test signal.

Finally, by increasing the number of separators involved in the fusion process, we could also expect to increase the SDR. Yet, our three experiments shows that the fusion on the number of components, on the initializations and on both result in approximately the same gain in SDR. As a consequence, it could also be interesting to investigate the fusion between separators which differ by the structure of the underlying models and not only by their hyperparameters and initializations.

6. CONCLUSION

We proposed a general *late* fusion framework inspired by classifier fusion which was applied to audio source separation algorithms based on time-frequency masking. We demonstrated its potential with a simplified temporal fusion rule applied to a voice extraction problem, as *oracle fusion* results reached higher SDR than individual separator results. The good performance of the fusion by mean showed that it can be a simple alternative to the fine tuning and initialization of separation models. However, the poor performance of *learned fusion* suggested that our learning dataset was not enough representative of the test mixtures. More recent experimental results showed that on a well-defined problem with a representative learning database, the *learned fusion* can effectively outperform the *fusion by mean* and nearly reach *oracle fusion* results. To go further, future work will be focused on testing the full time-frequency fusion rule through the combination of time-frequency masks as well as investigating the fusion of separators with models of different structure.

7. REFERENCES

[1] P. D. O'Grady, B. A. Pearlmutter, and S. T. Rickard, "Survey of sparse and non-sparse methods in source separation," *International Journal of Imaging Systems and Technology*, vol. 15, no. 1, pp. 18–33, 2005.

[2] Ö. Yılmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[3] D. P. W. Ellis and R. J. Weiss, "Model-based monaural source separation using a vector-quantized phase-vocoder representation," in *Proc. of IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*, 2006, vol. 5, pp. V–957–960.

[4] J.-L. Durrieu, G. Richard, B. David, and C. Févotte, "Source/filter model for unsupervised main melody extraction from polyphonic audio signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 3, pp. 564–575, 2010.

[5] G. Mysore, P. Smaragdis, and B. Raj, "Non-negative hidden Markov modeling of audio with application to source separation," *Latent Variable Analysis and Signal Separation*, pp. 140–148, 2010.

[6] D. F. Rosenthal and H. G. Okuno, *Computational auditory scene analysis*, L. Erlbaum Associates Inc., 1998.

[7] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[8] I. Bloch, A. Hunter, A. Appriou, A. Ayoun, et al., "Fusion: General concepts and characteristics," *International Journal of Intelligent Systems*, vol. 16, no. 10, pp. 1107–1134, 2001.

[9] S. Kirbiz and P. Smaragdis, "An adaptive time-frequency resolution approach for non-negative matrix factorization based single channel sound source separation," in *Proc. of IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*, 2011, pp. 253–256.

[10] I. Meganem, Y. Deville, and M. Puigt, "Blind separation methods based on correlation for sparse possibly-correlated images," in *Proc. of IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*. IEEE, 2010, pp. 1334–1337.

[11] E. Vincent, S. Araki, F. Theis, G. Nolte, et al., "The signal separation evaluation campaign (2007–2010): Achievements and remaining challenges," *Signal Processing*, vol. 92, no. 8, pp. 1928–1936, 2012.

[12] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval, "Adaptation of Bayesian models for single-channel source separation and its application to voice/music separation in popular songs," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1564–1578, 2007.

[13] C. Raphael, "A classifier-based approach to score-guided source separation of musical audio," *Computer Music Journal*, vol. 32, no. 1, pp. 51–59, 2008.

[14] A. M. Reddy and B. Raj, "Soft mask methods for single-channel speaker separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 6, pp. 1766–1776, 2007.

[15] Y. Li and D. L. Wang, "On the optimality of ideal binary time–frequency masks," *Speech Communication*, vol. 51, no. 3, pp. 230–239, 2009.

[16] D. M. J. Tax and R. P. W. Duin, "Using two-class classifiers for multiclass classification," in *Proc. of 16th IEEE International Conference on Pattern Recognition*, 2002, vol. 2, pp. 124–127.

[17] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 3, pp. 418–435, 1992.

[18] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[19] D. P. Bertsekas, *Nonlinear programming*, Athena Scientific, 1999.

[20] N. Bertin, R. Badeau, and G. Richard, "Blind signal decompositions for automatic transcription of polyphonic music: Nmf and k-svd on the benchmark," in *Proc. of IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*, 2007, vol. 1, pp. I–65–68.

[21] S. Wild, J. Curry, and A. Dougherty, "Improving non-negative matrix factorizations through structured initialization," *Pattern Recognition*, vol. 37, no. 11, pp. 2217–2232, 2004.

[22] M. Vinyes, "MTG MASS database," `http://www.mtg.upf.edu/static/mass/resources`, 2008.

[23] C. Févotte, R. Gribonval, and E. Vincent, "BSS_EVAL Toolbox User Guide – Revision 2.0," IRISA, Tech. Rep. 1706, `http://www.irisa.fr/metiss/bss_eval/`, 2005.