BLIND AUDIOVISUAL SOURCE SEPARATION USING OVERCOMPLETE DICTIONARIES

Anna Llagostera Casanovas[1], *Gianluca Monaci*[1], *Pierre Vandergheynst*[1], *Rémi Gribonval*[2]

[1]Signal Processing Institute, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[2]IRISA (CNRS & INRIA) - projet METISS, France

ABSTRACT

In this work we present a method to perform a complete audiovisual source separation without need of previous information. This method is based on the assumption that sounds are caused by moving structures. Thus, an efficient representation of audio and video sequences allows to build relationships between synchronous structures on both modalities. A robust clustering algorithm groups video structures exhibiting strong correlations with the audio so that sources are counted and located in the image. Using such information and exploiting audio-video correlation, the audio sources activity is determined. Next, *spectral* GMMs are learnt in time slots with only one source active so that it is possible to separate them in case of an audio mixture. Audio source separation performances are rigorously evaluated, clearly showing that the proposed algorithm performs efficiently and robustly.

Index Terms— Audiovisual processing, blind source separation, sparse signal representation, GMM

1. INTRODUCTION

When looking at an audiovisual sequence, our interest is focused in the part of the image that moves synchronously with the occurrence of a sound, because intuitively we feel that this movement has generated the sound. In this paper, we name audiovisual source this image segment together with the set of sounds that it has generated. For example, here we analyze sequences where two speakers are uttering numbers. One audiovisual source is thus composed of the image of one speaker and the sounds that he produces. However, we cannot associate to this source a part of the image (or soundtrack) belonging to the other speaker. What we want to do here is to separate these sources by completing four consecutive objectives. First, we want to know how many audiovisual sources are present in the sequence (one silent person cannot be considerated). Second, the visual part of these sources has to be determined and located in the image. Third, we need to detect the temporal periods where these audiovisual sources are active, i.e. when each person is speaking. Finally, these time slots can be used to build the source audio models and separate the original soundtrack when several sources are active at the same time. The first three objectives are achieved by using the method explained in [1] and reviewed shortly in Sec. 3. From a purely audio point of view, this is the part that ensures the blindness of the audio mixture separation, since the number of sources and their activity periods are the only information needed for the audio separation method explained in Sec. 3.4.

Few methods exist that exploit audiovisual coherence to separate *stereo* audio mixtures [2, 3, 4, 5]. All the existing algorithms consider the problem from an *audio source separation point of view*, i.e. they use the audio-video synchrony as side information to improve and overcome limitations of classical Blind Audio Source Separation (BASS) techniques [6]. We want to stress three important differences between the proposed approach and existing audiovisual separation methods :

1. State-of-the-art audiovisual separation algorithms exploit stereo audio signals, using classic BASS techniques helped by visual information. In contrast the audio signal we consider here comes from only *one microphone*;
2. Existing methods simplify the task of associating audio and video information. Either the audio-video association is given *a priori*, i.e. it is known which audio signal corresponds to which video signal [4, 5], either it is considered the case where one audiovisual source is mixed with an *audio-only* source [2, 3]. Here, in contrast, we simultaneously separate audio-video sources building correlations between acoustic and visual entities;
3. Existing algorithms, except for [4], require off-line training to build the audiovisual source model. This is mainly due to the fact that the algorithms in [2, 3, 5] try to map video information into the audio feature space using techniques similar to lip-reading (requiring moreover accurate mouth parameters that are difficult to acquire). Here, in contrast, no training is required.

In Sec. 2 we describe the audio and video features used to represent both modalities, while Sec. 3 details the *Blind Audiovisual Source Separation* (BAVSS) algorithm. In Sec. 4 we present the separation results obtained on real and synthesized audiovisual clips. Finally, in Sec. 5 achievements and future research directions are discussed.

2. AUDIO AND VIDEO REPRESENTATIONS

Audio Representation – The audio signal $a(t)$ is decomposed using the Matching Pursuit algorithm (MP) over a redundant dictionary of Gabor atoms $\mathcal{D}^{(a)}$ [7]. Thus, the signal $a(t)$ is approximated using K atoms as

$$a(t) \approx \sum_{k=0}^{K-1} c_k \phi_k^{(a)}(t) \,, \tag{1}$$

where c_k are the coefficients for every atom $\phi_k^{(a)}(t)$.

Video Representation – The video signal is represented using the video MP algorithm adopted in [7]. The sequence is decomposed into a set of video atoms representing salient visual components and their temporal transformations. The video signal $V(x_1, x_2, t)$ is approximated using N video atoms $\phi_n^{(v)}$ as

$$V(x_1, x_2, t) \approx \sum_{n=0}^{N-1} c_{n(t)} \phi_n^{(v)}(x_1, x_2, t) \,, \tag{2}$$

where $c_{n(t)}$ are the coefficients. The atoms $\phi_n^{(v)}$ are edge-like functions that are tracked across time. Each function is represented by a set of parameters describing its shape and position and that evolve through time [7]. The displacement of each video atom, $d_n(t) = \sqrt{t_{1_n}^2(t) + t_{2_n}^2(t)}$, is computed from its position parameters $(t_{1_n}(t), t_{2_n}(t))$.

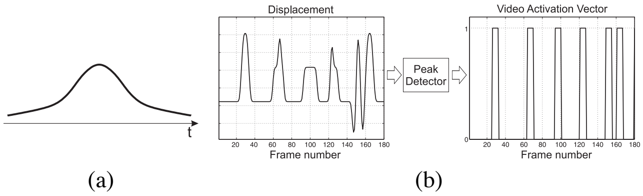


Fig. 1. Audio feature $\mathsf{f}_k(t)$ (a) and displacement function $d_n(t)$ with corresponding *Activation Vector* $\mathsf{y}_n(t)$ obtained for a video atom (b).

3. BLIND AUDIOVISUAL SOURCE SEPARATION (BAVSS)

The BAVSS process is composed of four main steps. First, video sources are localized using a clustering algorithm that spatially groups the video structures that are correlated with the audio atoms. Second, a spatial criterion is used to separate the sources. Third, the correlations between audio and video events are employed to identify temporal periods with only one active source (audio localization). Fourth, the sources frequency behavior is learned in time periods during which sources are active alone in order to separate them in the mixed periods.

Two main assumptions are made on the type of analyzed sequences. First, for each detected video source there is one and only one associated source in the audio mixture. This means that audio "distractors" in the sequence (e.g. a person speaking out of the camera's field of view) are considered as noise and their contribution to the mixture is associated to the sources found in the video. Moreover, we consider the video sources approximately static, i.e. their positions over the image plane do not change too much. This assumption is less stringent as it can be removed by analyzing the sequences using shifting time windows.

3.1. Video Source Localization

Correlation scores $\chi_{k,n}$ are computed between each audio atom $\phi_k^{(a)}$ and each video atom $\phi_n^{(v)}$. These scores measure the degree of synchrony between *relevant events* in both modalities : the presence of an audio atom (energy in the time-frequency plane) and a peak in the video atom displacement (oscillation from an equilibrium position).

Audio feature – The feature $\mathsf{f}_k(t)$ that we consider is the energy distribution of each audio atom projected over the time axis. In the case of Gabor atoms it is a Gaussian function whose position and variance depend on the atoms parameters (Fig.1(a)).

Video feature – An *Activation Vector* $\mathsf{y}_n(t)$ [7] is built for each atom displacement function $d_n(t)$ by detecting the peaks locations as shown in Fig. 1(b). The Activation Vector peaks are filtered by a window of width $W = 13$ samples in order to model delays and uncertainty.

Finally, a scalar product is computed between both features to obtain the *correlation scores*, $\chi_{k,n} = \langle \mathsf{f}_k(t), \mathsf{y}_n(t) \rangle, \forall\, k, n$. This value is high if the audio feature and the video displacement peak exhibit a big temporal overlap. Thus, a high correlation score means high probability for a video structure of having generated the sound.

The idea, now, is to spatially group all the structures belonging to the same speaker in order to estimate its position on the image. We define the empirical *confidence value* κ_n of the n-th video atom as the sum of the MP coefficients c_k of all the audio atoms associated to it in the whole sequence, $\kappa_n = \sum_k c_k$, with k such that $\chi_{k,n} \neq 0$. This value is a measure of the number of audio atoms related to this video structure and their weight in the MP decomposition of the audio track. Each video atom thus is characterized by its position over the image plane and by its confidence value, i.e. $((t_{1_n}, t_{2_n}), \kappa_n)$. We group all the video atoms correlated with the audio signal (i.e. with $\kappa_n \neq 0$) with the **clustering** algorithm described in [1]. The number of sources does not have to be specified in advance since a confidence measure is introduced to automatically eliminate unreliable clusters. The algorithm is robust and the localization results do not critically depend on the cluster parameters choice.

3.2. Video Source Separation

This step classifies *all* the video atoms closer than the cluster size D to a centroid into the corresponding source (in previous step only atoms with $\kappa_n \neq 0$ are considered). Each such group of video atoms, S_i, describes the video modality of an audiovisual source, achieving thus the Video Separation objective.

3.3. Audio Source Localization

The objective of this phase is to determine the temporal periods during which the sources are active. First, each audio atom $\phi_k^{(a)}$ is classified into its corresponding source in the following way :

1. Take all video atoms $\phi_n^{(v)}$ correlated with the audio atom $\phi_k^{(a)}$;
2. Each of these video atoms is associated to an audiovisual source S_i; for each source S_i compute a value H_{S_i} that is the sum of the correlation scores between the audio atom $\phi_k^{(a)}$ and the video atoms $\phi_j^{(v)}$ s.t. $j \in S_i$: $H_{S_i} = \sum_{j \in S_i} \chi_{k,j}$;
3. Classify the audio atom into the source S_i if the value H_{S_i} is twice as big as any other value H_{S_h} for the other sources. If this condition is not fulfilled, this audio atom can belong to several sources and further processing is required.

Using this labelling time periods during which only one source is active are clearly determined. This is done using a simple criterion : if in a continuous time slot longer than Δ seconds all audio atoms are assigned to S_i, then during this period only source S_i is active. In all experiments the value of Δ is set to 1 second.

3.4. Audio Source Separation

We perform a GMM-based audio source separation by modifying the method used in [8]. Here, the main difference is the use of the video information, which allows us to perform a *blind* separation since no previous information about the audio sources is required.

For each source k, the short-term Fourier spectrum $S_k(t)$ of the audio signal is modeled as a complex circular Gaussian random variable, with probability density $N_C(.)$, zero mean and diagonal covariance matrixes $R_{k,i} = \text{diag}[r_{k,i}^2(f)]$, that is:

$$p(S_k(t)|\Lambda_k^{spec}) = \sum_i u_{k,i} N_C(S_k(t); \bar{0}, R_{k,i}), \qquad k = 1, 2 \quad (3)$$

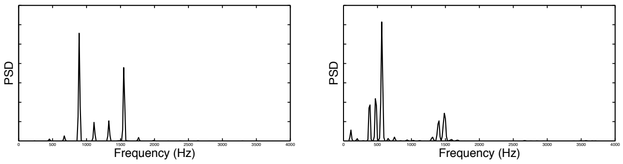


Fig. 2. Example of *spectral* GMM states for a female [left] and a male [right] speakers. Each state i is represented by its PSD : $r_i^2(f)$.

where $u_{k,i}$ are the weights of the gaussians with $\sum_i u_{k,i} = 1$. Then, the source ***spectral* GMMs** are defined as : $\Lambda_k^{spec} = \{u_{k,i}, R_{k,i}\}_i$. For each state i of the model, the diagonal of the covariance matrix $r_{k,i}^2(f)$ represents a local Power Spectral Density (PSD), as shown in Fig. 2. This figure shows how these states correctly characterize the sources frequency behavior: the lowest frequencies (on the left in the figures) concentrate the male's audio energy while the female's formants start to appear at higher frequencies. The typical periodicity in frequency of the speech can also be observed in the female's graphic.

The **learning process** is detailed next. For each source, the time slots where it is active alone form its training sequence. First, this training sequence is represented on the time-frequency plane by applying a Short Time Fourier Transform (STFT) using temporal windows of 512 samples length with 50% overlap. As a result, we obtain a set of short time Fourier spectra ($|S_k|^2$), but we are only interested in those that are representative of the source behavior. Thus, we use the K-Means algorithm to group all these PSDs into a number Q_K of spectral shapes that characterize the source k (similar PSDs are grouped together). Finally, for each state i, the *spectral* GMM parameters $\Lambda_k^{spec} = \{u_{k,i}, R_{k,i}\}_i$ are iteratively adjusted by using the Expectation Maximization algorithm. The formulas used for the parameters re-estimation are explained in depth in [8].

The method used for the **mixture separation** is explained in Algorithm 1. For each time instant we look for the most suitable couple of states given the mixture spectrum. This is done by maximizing a resemblance measure $\Theta(\cdot,\cdot)$, which is the inverse of the euclidean distance. This information is used to build a time-frequency Wiener mask for each source (5) by combining the *spectral* PSDs in the corresponding states $(r_{1,i*(t)}^2, r_{2,j*(t)}^2)$ with the knowledge about the sources activity w_k. When only one source is active, this weight w_k assigns all the soundtrack to this speaker. Otherwise, $w_k = 0.5$ and the analysis takes into account only the audio GMMs. In a further implementation we could assign intermediate values to w_k that account for the degree of correlation between audio and video. However, such cross-modal correlation has to be accurately estimated to improve the separation results.

Algorithm 1: Monochannel Source Separation using knowledge about sources activity

```
Input: Mixture $x$, Spectral GMMs $\Lambda_k^{spec} = \{u_{k,i}, R_{k,i}\}_i$
       and activity vectors $w_k$ for the sources $k = 1, 2$
Output: Estimation of the sources $\hat{s}_1$ and $\hat{s}_2$
Compute STFT of the mixture $X$ from the temporal signal $x$ ;
foreach $t = 1, 2, ..., T$ do
    1. Find the best combination of states (PSD) according to
    the mixture spectrum $|X(t)|^2$, that is :
        $(i^*(t), j^*(t)) = \arg\max_{(i,j)} \Theta(|X(t)|^2, r_{1,i}^2 + r_{2,j}^2)$ ,   (4)
    where $\Theta(.,.)$ is a resemblance measure.
    2. Build a time-frequency local mask using knowledge
    about the sources activity:
        $M_1(t,f) = \frac{r_{1,i*(t)}^2(f) * w_1(t)}{r_{1,i*(t)}^2(f) * w_1(t) + r_{2,j*(t)}^2(f) * w_2(t)}$   (5)
    3. Apply this local mask to the mixture spectrum $X(t)$ to
    obtain the estimated source spectrum :
        $\hat{S}_1(t,f) = M_1(t,f) X_1(t,f)$   (6)
end
Reconstruct the estimation of the source in the temporal
domain $\hat{s}_1$ from the STFT estimation $\hat{S}_1$ ;
```

4. EXPERIMENTS

The proposed BAVSS algorithm is evaluated on synthesized audio-visual mixtures, in order to have an objective evaluation of the algorithm's performances. Sequences are synthesized using clips taken from the *groups* partition of the CUAVE database [9] with one girl and one boy uttering sequences of digits alternatively. The video data is sampled at 29.97 frames/sec with a resolution of 480 × 720 pixels, and the audio at 44 kHz. The video has been resized to a 120 × 176 pixels, while the audio has been sub-sampled to 8 kHz. The video signal is decomposed into $N = 100$ video atoms and the soundtrack is decomposed into $K = 2000$ atoms.

Ground truth mixtures are obtained by temporally shifting audio and video atoms of one speaker in order to obtain time slots with both speakers active simultaneously. For further details on the adopted procedure, please refer to [1]. Fig. 3 shows the results obtained by the proposed method when analyzing clip g20 of CUAVE database. Waveforms are very similar for original and estimated tracks, and the audible quality of the estimated sequences is also remarkable.

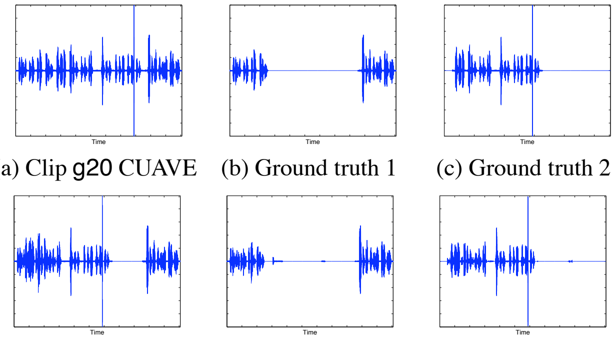
The **BSS Evaluation Toolbox** is used to evaluate the performance of the proposed method in the Audio Separation part. The estimated sources $\hat{s}_k$ are decomposed into: $\hat{s}_k = s_{target} + e_{interf} + e_{artif}$, as described in [6]. $s_{target} = f(s_k)$ is a version of the real sources s_k modified by an allowed distortion $f \in \mathcal{F}$, and e_{interf}, and e_{artif} are, respectively, the interferences and artifacts error terms. These three terms should represent the part of $\hat{s}_k$ perceived as coming from the wanted source s_k, from other unwanted sources $(s_{k'})_{j' \neq j}$ and from other causes. Three quantities are used for the evaluation, the source to distortion ratio, the source-to-interferences ratio, and the sources-to-artifacts ratio, defined as :

$$\mathrm{SDR} = 10 \log_{10} \frac{\|s_{target}\|^2}{\|e_{interf} + e_{artif}\|^2} \tag{7}$$

$$\mathrm{SIR} = 10 \log_{10} \frac{\|s_{target}\|^2}{\|e_{interf}\|^2} \tag{8}$$

$$\mathrm{SAR} = 10 \log_{10} \frac{\|s_{target} + e_{interf}\|^2}{\|e_{artif}\|^2} \tag{9}$$

For a given mixture, oracle estimators for single-channel source separation by time-frequency masking are computed by using the **BSS Oracle Toolbox** . The real-valued masks are subject to a unitary sum constraint. For further explanation about how the oracle masks are estimated, please refer to [10]. Then, SDR_{oracle}, SIR_{oracle} and SAR_{oracle} are established as upper bounds for the performance measures.



(a) Clip g20 CUAVE (b) Ground truth 1 (c) Ground truth 2

(d) Mixture (e) Estimated source 1(f) Estimated source 2

Fig. 3. Comparison between real (b)-(c) and estimated (e)-(f) soundtracks for a synthetic sequence (d) generated by applying a shift of 150 frames to the male speaker in clip g20 of CUAVE database (a).

Table 1 shows an important progress in all the sequences when using the GMM-based separation instead of the probabilistic method presented in [1]. Oracle values are also provided. Obtained results are satisfactory when the detected periods where each one of the present speakers is active alone are long enough (sequence g20). Due to the short duration of the analyzed sequences (30-40 seconds), the detection of this time slots has to be correctly performed in order to have enough time to train the $spectral$ GMMs properly or, what is the same, to have enough samples of the sources frequency behavior to be able to separate them in the future. In this case, separated sequences have a good audible quality and we have observed that results can be easily improved by iterating the separation algorithm. However, when the training sequences are shorter (sequences g12 and g21), the quality of the separated tracks gets worse although the numbers that each speaker utters can be easily understood.

5. CONCLUSION

In this paper we have introduced a novel algorithm to perform Blind Audiovisual Source Separation. We consider sequences made of one soundtrack and the video signal associated, without the stereo audio signal usually employed for the BASS task. The method correlates acoustic and visual structures that are represented using atoms taken from redundant dictionaries. Video atoms that exhibit strong correlations with the audio track and that are spatially close are grouped together using a robust clustering algorithm that can confidently count and localize on the image plane audiovisual sources. Then, using such information and exploiting the coherence between audio and video signals, audio sources are localized as well and separated. The presented algorithm needs time periods with sources active alone to learn GMMs that model their behavior and separate the mixture. This condition is however not very restrictive, since it is rare that in real-world mixtures all the sources are active all the time.

Several tests are performed in real-world and synthetic sequences, and encouraging results are obtained for both of them. The audible quality of the separated audio signals is reasonably good when the detected periods with only one speaker are long enough. An evaluation of the audio separation results has been performed using the BSS Evaluation Toolbox [6]. Separation results are still far from oracle results but clearly improve those obtained using the algorithm in [1]. Given the short length of the analyzed sequences, a possible improvement could be the adaptation of a general model for speech in time slots with a single speaker.

Sequence			probabilistic[1]	GMM-based	oracle
g12	female	SDR	-3.00	2.63	18.08
		SIR	0.18	9.38	32.32
		SAR	2.77	4.14	18.25
	male	SDR	-4.15	4.73	19.66
		SIR	5.18	11.40	31.72
		SAR	-2.46	6.09	19.95
g20	female	SDR	4.58	8.77	20.37
		SIR	12.49	19.91	34.73
		SAR	5.58	9.16	20.54
	male	SDR	5.43	9.91	21.44
		SIR	20.37	20.65	36.38
		SAR	5.61	10.33	21.58
g21	female	SDR	1.76	4.64	21.08
		SIR	8.72	14.71	35.61
		SAR	3.28	5.24	21.24
	male	SDR	1.63	5.49	21.61
		SIR	12.17	12.46	36.83
		SAR	2.29	6.70	21.75

Table 1. Results obtained with synthetic sequences generated for different clips of CUAVE database. All results are in dB.

6. REFERENCES

[1] A. Llagostera Casanovas, G. Monaci, and P. Vandergheynst, "Blind Audiovisual Source Separation Using Sparse Representations," in *Proc. IEEE ICIP*, 2007.

[2] D. Sodoyer, L. Girin, C. Jutten, and J.-L. Schwartz, "Developing an audio-visual speech source separation algorithm," *Speech Commununication*, vol. 44, no. 1-4, pp. 113–125, 2004.

[3] R. Dansereau, "Co-channel audiovisual speech separation using spectral matching constraints," in *Proc. IEEE ICASSP*, 2004, pp. 645–648.

[4] S. Rajaram, A. V. Nefian, and T.S. Huang, "Bayesian separation of audio-visual speech sources," in *Proc. IEEE ICASSP*, 2004, pp. 657–660.

[5] W. Wang, D. Cosker, Y. Hicks, S. Saneit, and J. Chambers, "Video assisted speech source separation," in *Proc. IEEE ICASSP*, 2005, pp. 425–428.

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

[7] G. Monaci, Ò. Divorra, and P. Vandergheynst, "Analysis of multimodal sequences using geometric video representations," *Signal Processing*, vol. 86, no. 12, pp. 3534–3548, 2006.

[8] 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 music," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1564–1578, 2007.

[9] E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. N. Gowdy, "Moving-talker, speaker-independent feature study, and baseline results using the CUAVE multimodal speech corpus," *EURASIP JASP*, , no. 11, pp. 1189–1201, 2002.

[10] E. Vincent, R. Gribonval, and M. D. Plumbley, "Oracle estimators for the benchmarking of source separation algorithms," Tech. Rep. C4DM-TR-06-03, QMUL, 2006.