Blind Audio-Visual Source Separation based on Sparse Redundant Representations

Anna Llagostera Casanovas[1]*, Gianluca Monaci[2], Pierre Vandergheynst[1], Rémi Gribonval[3]

***Abstract*—In this paper we propose a novel method which is able to detect and separate audio-visual sources present in a scene. Our method exploits the correlation between the video signal captured with a camera and a synchronously recorded one-microphone audio track. In a first stage, audio and video modalities are decomposed into relevant basic structures using redundant representations. Next, synchrony between relevant events in audio and video modalities is quantified. Based on this co-occurrence measure, audio-visual sources are counted and located in the image using a robust clustering algorithm that groups video structures exhibiting strong correlations with the audio. Next periods where each source is active alone are determined and used to build *spectral* Gaussian Mixture Models (GMMs) characterizing the sources acoustic behavior. Finally, these models are used to separate the audio signal in periods during which several sources are mixed. The proposed approach has been extensively tested on synthetic and natural sequences composed of speakers and music instruments. Results show that the proposed method is able to successfully detect, localize, separate and reconstruct present audio-visual sources.**

***Index Terms*—Audio-visual processing, blind source separation, sparse signal representation, Gaussian Mixture Models.**

I. Introduction

It is well known from every-day experience that visual information strongly contributes to the interpretation of acoustic stimuli. This is particularly evident if we think about speech: speaker lips movements are correlated with the produced sound and the listener can exploit this correspondence to better understand speech, especially in adverse environments [1], [2]. The multi-modal nature of speech is exploited since at least two decades to design speech enhancement [3], [4], [5] and speech recognition algorithms [6], [7] in noisy environments. Lately, this paradigm has been adopted also in the speech separation field to increase the performances of audio-only methods [8], [9], [10], [11], [12].

Audio-visual analysis is receiving increasing attention from the signal processing and computer vision communities, as it is at the basis of a broad range of applications, from automatic speech/speaker recognition to robotics or indexing and segmentation of multimedia data [13], [14], [15], [16]. Let us consider the example of a meeting. The scene is composed of several people speaking in turns or, sometimes, having parallel conversations. Detecting the current speaker/speakers and associating to each one of them the correct audio portions is extremely useful. For example, one could select one person and obtain the corresponding speech and image without the interference of other speakers. It can then be possible to index the whole meeting by using a speech-to-text algorithm. In this way one can search through amounts of indexed data by keywords and recover the target scene (or the person or exact date where the word appeared for example). The core of all these applications is the audio-visual source separation. In this paper we present a new algorithm which is able to automatically detect and separate the audio-visual sources that compose a scene.

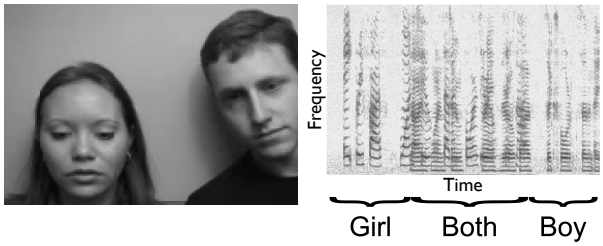


Fig. 1. Example of a sequence considered in this work. The sample frame [left] shows the two speakers; as highlighted on the audio spectrogram [right], in the first part of the clip the girl on the left speaks alone, then the boy on the right starts to speak as well, and finally the girl stops speaking and the boy speaks alone.

One typical sequence that we consider in this work, taken from the *groups* section of the CUAVE database [17], is shown in Fig. 1. It involves two speakers arranged as in Fig. 1 [left] that utter digits in English. As highlighted in Fig. 1 [right], in the first part of the clip the girl on the left speaks alone, then the boy on the right starts to speak as well, and finally the girl stops speaking and the boy speaks alone. In this case, one audio-visual source is composed of the image of one speaker and the sounds that she/he produces. However, we must not associate to this source a part of the image (or soundtrack) belonging to the other speaker. What we want to do here is to detect and separate these audio-visual sources.

In a first stage towards a complete audio-visual source separation, several methods exploited synchrony between audio and video channels to improve the results in the *audio* source separation domain when *two* microphones are available [8],

This work was partly funded by the Swiss NFS through grant number 200021-117884 and by the EU Framework 7 FET-Open project FP7-ICT-225913-SMALL: Sparse Models, Algorithms and Learning for Large-Scale data.

[1]Anna Llagostera Casanovas and Pierre Vandergheynst are with the Signal Processing Laboratory 2, EPFL, Switzerland.

[2]Gianluca Monaci contributed to this work while at the Signal Processing Laboratory 2, EPFL, Switzerland. Now, he is with the Video & Image Processing group, Philips Research, Eindhoven, the Netherlands.

[3]Rémi Gribonval is with Centre de Recherche INRIA Rennes - Bretagne Atlantique, Rennes, France.

e-mail: {anna.llagostera,pierre.vandergheynst}@epfl.ch, gianluca.monaci@philips.com, remi.gribonval@inria.fr

[9], [10], [11], [12]. In [10] the audio activity for each source (speaker) is assessed by computing the amount of motion in a previously detected mouth region. Then, the sources activity is used to improve the audio separation results when important noise is present. This method can only be used in speech mixtures recorded with more than one microphone. Approaches described in [8], [9], [11], [12] first build audio-visual models for each source and then they use them to separate a given *audio* mixture. For those last methods, the sources in the mixture and the video part of each one of them need to be known in advance, and the audio-visual source model is also built off-line.

Only two methods attempt a complete audio-visual source separation using a video signal and the corresponding *one-microphone* soundtrack [18], [19]. Barzelay and Schechner propose in [18] to assess the temporal correlation between audio and video *onsets*, which are respectively the beginning of a sound and a significant change on the speed or direction of a video structure. Audio-visual objects (AVO) are assumed to be composed of the video structures whose onsets match a majority of audio onsets and the audio signal associated to those audio onsets. The audio part of each AVO is computed by tracking the frequency formants that follow the presence of its audio onsets. In [19] a similar approach using canonical correlation analysis for finding correlated components in audio and video is presented. This approach uses trajectories of "interest" points in the same way as in [18] and it adds an implementation using microphone arrays. The main differences between those approaches and our method are the following:

1. The objective of the proposed method is to separate and reconstruct *audio-visual* sources. We want to stress that our sources are audio-visual, not only audio or video. Existing methods do that only partially: they locate the video structures more correlated with the audio and separate the audio (in [19] there is no evidence however). Both methods do not attempt to reconstruct the video part of the sources. Concerning the audio, in [18] the separated soundtracks are recovered with an important energy loss due to the formants tracking, while in [19] no separated soundtracks are shown or analyzed.
2. We separate audio-visual sources using a simple and very important observation: it is very unlikely that sources are mixed all the time. Thus we detect periods during which audio-visual sources are active alone and periods during which they are mixed. This is a very important step because once one has this information, any one microphone *audio* source separation technique can be used. Thus, we do not need to know in advance the characteristics of the sources composing the mixture (off-line training is not needed anymore), since acoustic models for the sources can be learnt in periods where they are active alone.

In this research work, the robust separation of audio-visual sources is achieved by solving four consecutive tasks. First, we estimate the number of audio-visual sources present in the sequence (i.e. one silent person cannot be considered as a source). Second, the visual part of these sources is localized in the image. Third, we detect the temporal periods during which each audio-visual source is active alone. Finally, these time slots are used to build audio models for the sources and separate the original soundtrack when several sources are active at the same time. From a purely audio point of view, the video information ensures the blindness of the one microphone audio source separation explained in Section VI-B. The number of sources in the sequence and their characteristics are determined by combing audio and video signals. As a result, our algorithm does not need any previous information or off-line training to separate the audio mixture and accomplish the whole audio-visual source separation task.

The paper has the following structure: in Section II we describe the *Blind Audio-Visual Source Separation* (BAVSS) algorithm, while Section III details the audio and video features used to represent both modalities. Section IV presents the method employed to assess and quantify the synchrony between audio and video relevant events: a key-point in our algorithm. Next, in Section V and Section VI the methodology used for the video and audio separation respectively is explained in depth. Section VII introduces the performance measures that are used in the evaluation of our method. Section VIII presents the separation results obtained on real and synthesized audio-visual clips. Finally, in Section IX achievements and future research directions are discussed.

II. Blind Audio-Visual Source Separation (BAVSS)

Figure 2 schematically illustrates the whole Blind Audio-Visual Source Separation (BAVSS) process. We observe N audio-visual sources, each one composed of its visual part and its audio part. Thus, the soundtrack can be expressed as a set of N audio sources $a(t) = \{a_1(t), a_2(t), \ldots, a_N(t)\}$, and the video signal as a set of N video sources $v(x_1, x_2, t) = \{v_1(x_1, x_2, t), v_2(x_1, x_2, t), \ldots, v_N(x_1, x_2, t)\}$. Audio and video signals are decomposed using redundant representations into K audio *atoms* $\phi_k^{(a)}(t)$ and M video *atoms* $\phi_m^{(v)}(x_1, x_2, t)$ respectively, as explained in Section III. Audio and video atoms describe meaningful features of each modality in a compact way: an audio atom indicates the presence of a sound and each video atom represents a part of the image and its evolution through time.

In the next block, the fusion between audio and video modalities is performed at the atom level by assessing the temporal synchrony between the presence of a sound and an oscillatory movement of a video structure as explained in Section IV. The result is a set of correlation scores $\chi_{k,m}$ that associate each audio atom k to each video atom m according to their synchrony.

Next audio-visual sources are counted and localized using a clustering algorithm that spatially groups video structures whose movement is synchronous with the presence of sounds in the audio channel (Section V-A). These initial steps are the most important ones for the BAVSS process since they assess the relationships between audio and video structures and determine the number N of present audio-visual sources. Thus, in order to recover an estimate of the video part of each source we only need to assign the video atoms to the sources taking into account their positions in the image (the procedure is detailed in Section V-B).

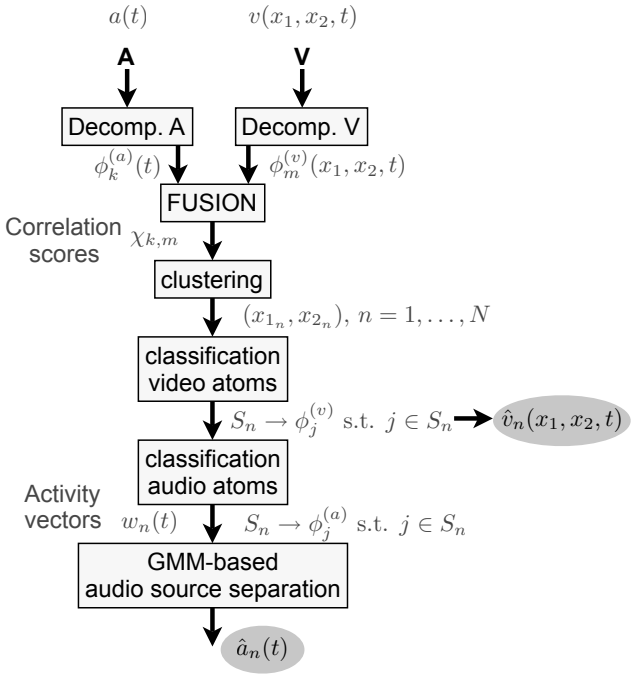


Fig. 2. Block diagram of the proposed audio-visual source separation algorithm. Audio and video channels are **decomposed** using redundant representations. Temporal correlation between *relevant events* in both modalities is assessed and quantified in the **fusion** stage, giving as a result the correlation scores $\chi_{k,m}$ between audio and video atoms. Next, video atoms that present strong correlations with the whole soundtrack are grouped together using a **clustering** algorithm that determines the number of audio-visual sources N in the scene and locates them on the image. Then, video atoms are assigned to the corresponding sources using a proximity criterion, which provides an estimation of the video part of the sources. At this point, audio atoms are classified into the sources taking into account their correlation with the labelled video atoms. The activity of each source (represented by activity vectors in the diagram) is determined according to the audio atoms classification. Finally, *spectral* GMMs for the sources are built in temporal periods where the sources are active alone and these models are used to separate sources when they are mixed. In this way the audio part of the sources is also estimated and the process is completed.

Then, each audio atom is assigned to one source according to the classification of the associated video atoms. However, this labelling of the audio atoms is not sufficient to clearly separate the audio sources. This is due to the fact that until this point our method only assesses the temporal synchrony between audio and video structures, and thus it is not discriminant when several sources are mixed. Thus we use the audio atoms classification to detect the temporal periods of activity of each source as explained in Section VI-A. The audio mixture is separated according to the *spectral* Gaussian Mixture Models that are built in time slots during which each source is active alone (Section VI-B). In this final step we obtain the estimates for the audio part of the sources and the complete audio-visual separation is achieved. The choice of the GMMs for the audio separation is motivated by their simplicity and the fact that GMMs can effectively represent the variety of sounds structures [20]. However, once the periods of activity of the sources are determined any one microphone audio source separation algorithm can be used.

Two main assumptions are made on the type of sequences that we can analyze. First, we assume that for each detected video source there is one and only one associated source in the audio mixture. This means that if there is an audio "distracter" in the sequence (e.g. a person speaking out of the camera's field of view), it is considered as noise and its contribution to the soundtrack is associated to the sources found in the video. This assumption simplifies the analysis, since we know in advance that a one-to-one relationship between audio and video entities exists. The relaxation of this assumption will be the object of future investigation. Moreover, we consider the video sources approximately static globally, i.e. their location over the image plane do not change too much (sources never switch their positions for example). Again, this second assumption is made for simplicity and it can be removed by using a 3-D clustering of the video atoms (using also the temporal dimension) instead of a 2-D clustering. The video decomposition gives the position of the atom at each time instant and thus we can group together atoms that stay close through time to the video atoms most correlated to the soundtrack.

III. Audio and Video Representations

The effectiveness of the proposed algorithm is basically due to the representations used for describing the audio and video signals. These representations decompose the signals according to their salient structures, whose variations in characteristics such as dimensions or position represent a relevant change in the whole signal. For example, a variation in one pixel value may mean movement or not, but a position change of one full structure will probably have this meaning. Next subsections describe representation techniques used for audio and video signals.

A. Audio Representation

The audio signal $a(t)$ is decomposed using the Matching Pursuit algorithm (MP) [21] over a dictionary of Gabor atoms $\mathcal{D}^{(a)}$, where a single window function, $g^{(a)}$, generates all the atoms that compose the dictionary. Each atom $\phi_k^{(a)} = U_k g^{(a)}$, is built by applying a transformation U_k to the mother function $g^{(a)}$. The possible transformations are scaling by $s > 0$, translation in time by u and modulation in frequency by ξ. Then, indicating with an index k the set of transformations (s, u, ξ), an atom can be represented as

$$\phi_k^{(a)}(t) = \frac{1}{\sqrt{s}}\, g^{(a)}\left(\frac{t-u}{s}\right) e^{i\xi t}\,, \tag{1}$$

where the value $1/\sqrt{s}$ makes $\phi^{(a)}(t)$ unitary. According to these definition, each audio atom represents a sound and, more concisely, a concentration of acoustic energy around time u and frequency ξ.

Thus, an audio signal $a(t)$ can be approximated using K atoms as

$$a(t) \approx \sum_{k=0}^{K-1} c_k \phi_k^{(a)}(t)\,, \tag{2}$$

where c_k corresponds to the coefficient for every atom $\phi_k^{(a)}(t)$ from dictionary $\mathcal{D}^{(a)}$.

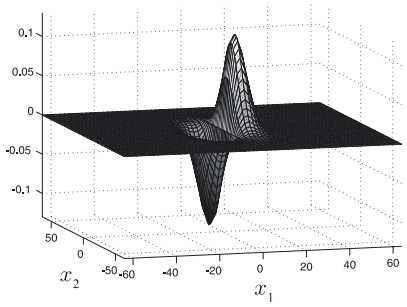


Fig. 3. The generating function $g^{(v)}(x_1, x_2)$ expressed by (4).

MP decomposition provides a sparse representation of the audio energy distribution in the time-frequency plane, highlighting the frequency components evolution. Moreover, MP performs a denoising of the input signal, pointing out the most relevant structures [21].

B. Video Representation

The video signal is represented using the 3D-MP algorithm proposed by Divorra and Vandergheynst in [22]. The video signal is decomposed into a set of video atoms representing salient video components and their temporal transformations (i.e changes in their position, size and orientation). Unlike the case of simple pixel-based representations, when considering image structures that evolve in time we deal with dynamic features that have a true geometrical meaning. Furthermore, sparse geometric video decompositions provide compact representations of information, allowing a considerable dimensionality reduction of the input signals.

First of all, the first frame of the video signal, $I_1(x_1, x_2)$, is approximated with a linear combination of atoms retrieved from a redundant dictionary $\mathcal{D}^{(v)}$ of 2-D atoms as

$$I_1(x_1, x_2) \approx \sum_{p \in \Omega} c_p \phi_p^{(v)}(x_1, x_2)\,, \tag{3}$$

where c_p is the coefficient corresponding to each 2-D video atom $\phi_p^{(v)}(x_1, x_2)$ and Ω is the subset of selected atom indexes from dictionary $\mathcal{D}^{(v)}$. As in the audio case, the dictionary is built by varying the parameters of a mother function, an edge-detector atom with odd symmetry, that is a Gaussian along one axis and the first derivative of a Gaussian along the perpendicular one (see Fig. 3). The generating function $g^{(v)}$ is thus expressed as

$$g^{(v)}(x_1, x_2) = 2x_1 \cdot e^{-(x_1^2 + x_2^2)}\,. \tag{4}$$

Then, this 2-D atoms are tracked from frame to frame using a modified MP approach based on a Bayesian decision criteria as explained in [22]. The possible transformations applied to $g^{(v)}$ to build the video atoms are: translations over the image plane $\vec{r} = (r_1, r_2)$, scaling $\vec{s} = (s_1, s_2)$ to adapt the atom to the considered image structure and rotations θ to locally orient the function along the edge.

Thus, the video signal can be approximated using M 3-D video atoms $\phi_m^{(v)}$ as

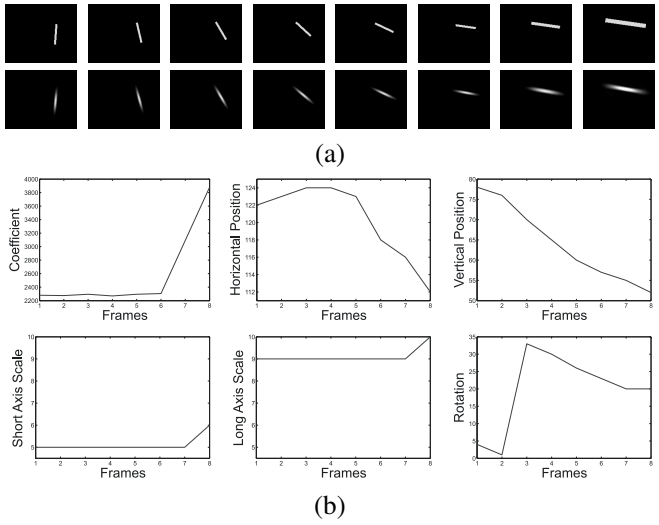


Fig. 4. (a) [Top row] Original synthetic sequence made by a white bar moving black uniform background. [Bottom row] Approximation using one video atom. (b) Parameter evolution of the atom. From left to right and from up down: coefficient $c_{m(t)}$, horizontal position r_1, vertical position r_2, short axis scale s_1, long axis scale s_2, rotation θ.

$$V(x_1, x_2, t) \approx \sum_{m=0}^{M-1} c_{m(t)} \phi_m^{(v)}(x_1, x_2, t)\,, \tag{5}$$

where the coefficients $c_{m(t)}$ vary through time and where each video atom $\phi_m^{(v)}$ is obtained by changing from frame to frame the parameters $(r_{1_m}, r_{2_m}, s_{1_m}, s_{2_m}, \theta_m)$ of a reference 2-D atom $\phi_m^{(v)}(x_1, x_2)$:

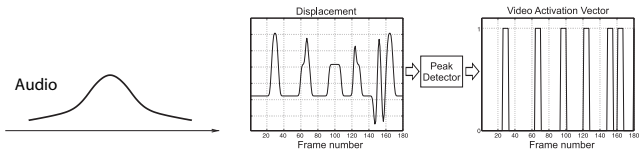
$$\phi_m^{(v)}(x_1, x_2, t) = \phi_{m(t)}^{(v)}(x_1, x_2)\,. \tag{6}$$

An illustration of this video decomposition can be observed in Figure 4, where the approximation of a simple synthetic object by means of a single video atom is performed. Figure 4(a) shows the original sequence (top row) and its approximation composed of a single geometric term (bottom row). Figure 4(b) depicts the parametric representation of the sequence: we find the temporal evolution of the coefficient $c_{m(t)}$ and of the position, scale and orientation parameters. This 3D-MP video representation provides a parametrization of the signal which concisely represents the image geometric structures *and* their temporal evolution.

As explained in Section II the correlation between audio and video signals is determined by assessing the temporal synchrony between the presence of a sound and an oscillatory movement of a relevant video structure. At this point, the video is already decomposed into relevant structures (atoms) and what we need is to compute their movement. Thus, for each video atom $\phi_m^{(v)}$ we compute a feature describing its displacement $d_m(t) = \sqrt{r_{1_m}^2(t) + r_{2_m}^2(t)}$ by using the position parameters $(r_{1_m}(t), r_{2_m}(t))$ extracted from the tracking step of the decomposition at each frame t.

IV. Audio-video Atomic Fusion

Quantifying the relationships between audio and video structures is the most important part in the whole process. All



(a) (b)

Fig. 5. Audio feature $\mathsf{f}_k(t)$ (a) and displacement function $d_m(t)$ with corresponding *Activation Vector* $\mathsf{y}_m(t)$ obtained for a video atom (b).

the audio-visual information that is used in the next steps of the algorithm is extracted here. Thus, the fusion method that we choose in order to assess the correlation between audio and video determines the performance of the proposed method.

As explained before, approaches in audio-visual analysis are based in an assumption of synchrony between related events in audio and video channels, i.e. when a person is speaking his/her lips movements are temporally correlated to the speech. According to this observation, *correlation scores* $\chi_{k,m}$ are computed between each audio atom $\phi_k^{(a)}$ and each video atom $\phi_m^{(v)}$. 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 u and s respectively (Fig. 5(a)).

Video feature: An *Activation Vector* $\mathsf{y}_m(t)$ [23] is built for each atom displacement function $d_m(t)$ by detecting the peaks locations as shown in Fig. 5(b). The Activation Vector peaks are filtered by a window of width $W = 13$ samples in order to model delays and uncertainty.

There are two important remarks to be done concerning the video features that we use. First of all, it is important to clarify that the peaks on the displacement function $d_m(t)$ represent an oscillatory movement of the atom m. Thus, the *Activation Vector* $\mathsf{y}_m(t)$ does not depend on the original or relative position of the video atom m in the image. Notice that the peaks are situated at the time instant where a change in the direction of the movement appears. That can be interpreted as a change in the sign of the acceleration of the atom or, what is the same, an oscillation on the movement of that atom. The second remark concerns the choice of the parameter that models delays between audio and video *relevant events*. Here $W = 13$ samples corresponds to 0.45 seconds, a time delay between a movement and the presence of the corresponding sound that appears to be appropriate. From informal tests the setting of W results not to be critical as its value can be changed within a range of several samples without affecting significantly the algorithm performance.

Finally, a scalar product is computed between audio and video features in order to obtain the *correlation scores*:

$$\chi_{k,m} = \langle \mathsf{f}_k(t), \mathsf{y}_m(t) \rangle \ , \qquad \forall\, k, m \ . \tag{7}$$

This value is high when the audio atom and a peak in the video atom's displacement overlap in time or, what is the same, when a sound (audio energy) occurs more or less at the same time than the video structure is moving. Thus, a high correlation score means high probability for a video structure of having generated the sound.

V. Video Separation

A. Spatial Clustering of Video Atoms

The idea now is to spatially group all the structures belonging to the same source in order to estimate the source position on the image. We define the empirical *confidence value* κ_m of the m-th video atom as the sum of the MP coefficients c_k of all the audio atoms associated to it in the whole sequence, $\kappa_m = \sum_k c_k$, with k such that $\chi_{k,m} \neq 0$. 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. Thus, a video atom m whose motion presents a high synchrony with sounds in the audio channel will have a high confidence value κ_m, since a large number of important audio atoms in the sequence will be associated to this video atom in the audio-video atomic fusion step (Section IV). In contrast, low values for k_m correspond to video atoms whose motion is occasionally (and not continually) synchronous to the sounds.

Typically, the video part of each source is composed of groups of atoms presenting high confidence values κ_m (and thus high coherence with the audio signal), which are concentrated in a small region in the image plane. Thus, a spatial clustering becomes a natural way to count the sources in the sequence and estimate their position in the image. Let each video atom be characterized by its position over the image plane and its confidence value, i.e. $((r_{1_m}, r_{2_m}), \kappa_m)$. In this work, we cluster the video atoms correlated with the audio signal (i.e. with $\kappa_m \neq 0$) following these three steps:

1. **Clusters Creation:** The algorithm creates Z clusters $\{C_i\}_{i=1}^{Z}$, by iteratively selecting the video atoms with highest confidence value (and thus highest coherence with the audio track) and adding to them video atoms closer than a *cluster size* R defined in pixels. Video atoms belonging to a cluster can not be the center of a new cluster. Thus each new cluster is generated by the video atom with highest confidence value from those which have not been classified yet;
2. **Centroids Estimation:** The center of mass of each cluster is computed taking the confidence value of every atom as the mass. The resulting centroids are the coordinates in the image where the algorithm locates the audio-visual sources;
3. ***Unreliable Clusters* Elimination:** We define the *cluster confidence value* K_{C_i} as the sum of the confidence values κ_j of the atoms belonging to the cluster C_i, i.e. $K_{C_i} = \sum_{j \in C_i} \kappa_j$. Based on this measure, *unreliable clusters*, i.e. clusters with small confidence value K_{C_i} are removed, obtaining the final set of $N \leq Z$ clusters, $\{C'_n\}_{n=1}^{N}$, with centroids (x_{1_n}, x_{2_n}). In this step we remove cluster C_i if

$$K_{C_i} < 0.1 \cdot \max_h K_{C_h} \quad \text{with } h = 1, \ldots, Z \ , \ h \neq i \ . \tag{8}$$

Fig. 6. Example of the video sources reconstruction. On the left picture the left person is speaking while on the right picture the right person is speaking.

Further details about this clustering algorithm can be found in [24]. At this stage a good localization of sources in the image is achieved. The number of sources N does not have to be specified in advance since a confidence measure is introduced to automatically eliminate unreliable clusters. In [24] we show that the results are not significantly affected by the cluster parameters choice. For R ranging between 40 and 90 pixels the proposed clustering algorithm has been proved to detect the correct number of sources N (in all experiments image dimensions are 120×176 pixels). In fact, when we decrease the *cluster size* R more possible sources appear (Z increases), but all these clusters are far from the mouth and present a small correlation with the audio signal. Thus, step 3 of the algorithm easily removes clusters that do not represent an audio-visual source as their confidence K_{C_i} is much smaller.

B. Video Atoms Classification

This step classifies *all* video atoms closer than the cluster size R to a centroid into the corresponding source. Notice that only video atoms moving coherently with sounds ($\kappa_m \neq 0$) are considered for the video localization in Section V-A. Each such group of video atoms describes the video modality of an audio-visual source, achieving thus the video separation objective. Then, an estimate of the video part of the $n-$th source, S_n, can be computed simply as

$$\hat{v}_n(x_1, x_2, t) = \sum_{j \in S_n} c_{j(t)} \phi_j^{(v)}(x_1, x_2, t) \ . \tag{9}$$

Figure 6 shows an example of the reconstruction of the current speaker detected by the algorithm. Only video atoms close to the sources estimated by the presented technique are considered. Thus, to carry out the reconstruction, the algorithm adds their energy and the effect is a highlight of the speaker's face. In both frames, the correct speaker is detected.

VI. Audio Separation

A. Audio Atoms Classification

For every audio atom we take into account all related video atoms, their correlation scores and their classification into a source. Accordingly, an audio atom should be assigned to the source gathering most video atoms. Since we also want to reward synchrony, the assignation of each audio entity $\phi_k^{(a)}$ is performed in the following way:

1. Take all the video atoms $\phi_m^{(v)}$ correlated with the audio atom $\phi_k^{(a)}$, i.e. for which $\chi_{k,m} \neq 0$;
2. Each of these video atoms is associated to an audio-visual source S_n ; for each source S_n compute a value H_{S_n} that is the sum of the correlation scores between the audio atom $\phi_k^{(a)}$ and the video atoms $\phi_j^{(v)}$ s.t. $j \in S_n$:
$$H_{S_n} = \sum_{j \in S_n} \chi_{k,j} \, ; \tag{10}$$
Thus, this step rewards sources whose video atoms present a high synchrony with the considered audio atom.
3. Classify the audio atom into the source S_n if the value H_{S_n} is "big enough": here we require H_{S_n} to be twice as big as any other value H_{S_h} for the other sources. Thus we attribute $\phi_k^{(a)}$ to S_n if
$$H_{S_n} > 2 \cdot H_{S_h} \quad \text{with } h = 1, \ldots, N \, , \; h \neq n \, . \tag{11}$$
If this condition is not fulfilled (this is typically the case when several sources are simultaneously active), this audio atom can belong to several sources and further processing is required. This decision bound is not a very critical parameter since it only affects the classification of the audio atoms in time slots with several active sources. In periods with only one source, the difference between the score for the considered source H_{S_n} and the others is enormous and it is thus easy to classify the atom into the correct source.

Using the labels of audio atoms, time periods during which only one source is active are clearly determined. This is done using a very simple criterion: if in a continuous time slot longer than Δ seconds all audio atoms are assigned to source S_n, then during this period only source S_n is active. In all experiments the value of Δ is set to 1 second. The choice of this parameter has been done according to the length of the analyzed sequences (around 20 seconds). This value has to be small enough to ensure that in a period there is *only one* source active. At the same time, it has to be big enough to allow the presence of periods where to train the source audio models. Thus, Δ could be set automatically according to the length of the analyzed clip, e.g. one tenth of the sequence length.

When several sources are present, temporal information alone is not sufficient to discriminate different audio sources in the mixture. To overcome this limitation, in these *ambiguous* time slots a time-frequency analysis is performed, which is presented in details in the next section.

B. GMM-based Audio Source Separation

As explained in Section II, the choice of the *spectral* Gaussian Mixture Models (GMMs) as our method for the separation of the audio part of the sources has been motivated by two main reasons. Inspite of its simplicity, we can achieve good audio separation since GMMs are able to model multiple Power Spectral Densities or, what is the same, several frequency behaviors for the same source. This is a very interesting property given the diverse nature of sounds. Thus GMMs have the capacity of modelling non-stationary signals contrary to classical Wiener filters [20].

Here, we perform a one microphone GMM-based audio source separation inspired by the supervised approach in [25] but introducing the video information. The method in [25] needs to know *in advance* the sources that compose the

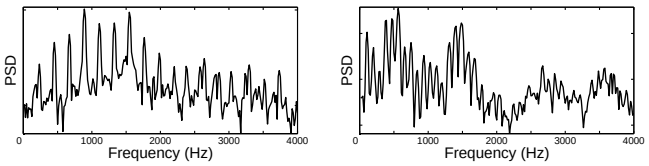


Fig. 7. Example of *spectral* GMM states learned by our algorithm for female [left] and male [right] speakers. Each state i is represented by its PSD in dB: $\log(r_i^2(f))$.

mixture and their characteristics: the audio model for each source is built off-line. Here the information extracted from the video signal through previous steps of our algorithm allows the application of the method without any off-line training. Thus, the separation that we perform is completely *blind* since no previous information about the sources is required.

The idea is to model the short time Fourier spectra of the sources by GMMs learned from training sequences $a_n^{train}(t)$. Using these models, the audio source separation is performed applying time-frequency masking on the Short Time Fourier Transform (STFT) domain. We will first explain our model for the sources, next the process we use to learn these models and finally the separation part.

Given an audio signal $z(t)$, we denote the STFT of this signal $Z(\tau, f)$ and $Z_{\tau'} = Z(\tau, f)|_{\tau=\tau'}$ the short time Fourier spectrum of the signal at time τ'. The short time Fourier spectra of the signal, Z_τ, are modeled with a GMM, i.e. the probability density function of Z_τ is given by

$$p(Z_\tau|\Lambda^{spec}) = \sum_i u_i N(Z_\tau; R_i)\,, \tag{12}$$

with

$$N(Z_\tau; R_i) = \prod_f \frac{1}{\pi r_i^2(f)} \exp\left[-\frac{|Z_\tau(f)|^2}{r_i^2(f)}\right]. \tag{13}$$

Here $Z_\tau(f)$ is the complex value of the short time Fourier spectrum Z_τ at frequency f and $r_i^2(f)$, representing the local Power Spectral Density (PSD) at frequency f in the state i of the GMM, is the diagonal element of the diagonal covariance matrix $R_i = \mathrm{diag}[r_i^2(f)]$. This *spectral* GMM is denoted $\Lambda^{spec} = \{u_i, R_i\}_i$.

Figure 7 shows two states of the GMMs that are learned by this method for a female [left] and a male [right] speaker. The states correctly characterize the sources frequency behavior: the male's audio energy is mainly present at lower frequencies (Fig. 7 [left]) while the female's harmonics (peaks in the PSD) start to appear at higher frequencies. A deeper analysis of this figure shows that for the female speaker, the fundamental frequency f_0 is around 220Hz (harmonics appear at multiples of 220Hz) while for the male it is around 110Hz. Those values for f_0 are within the range of the average speaking fundamental frequency for women (between 188 and 221 Hz) and for men (between 100 and 146 Hz) [26].

Let us now describe the **learning process**. For each source n, a training sequence $a_n^{train}(t)$ is composed of the detected time slots where the source is active alone, which are determined in Section VI-A. Next, the training sequence $a_n^{train}(t)$ is represented on the time-frequency plane $A_n^{train}(\tau, f)$ by applying a STFT using temporal windows of 512 samples length (64ms at 8kHz of sampling frequency) with 50% overlap. Then, the model $\Lambda_n^{spec} = \{u_{n,i}, R_{n,i}\}_i$ is learned by maximization of the likelihood $p(A_{n_\tau}^{train}|\Lambda_n^{spec})$. This maximization is iteratively adjusted using the Expectation Maximization (EM) algorithm initialized by Vector Quantization (VQ) to Q_n states. The formulas used for the parameters re-estimation are shown in Algorithm 1 and explained in detail in [20].

Algorithm 1: Learning of the *spectral* GMM parameters $\Lambda_n^{spec} = \{u_{n,i}, R_{n,i}\}_i$ by Expectation Maximization

Input: Short time Fourier spectra of the training signal $A_{n_\tau}^{train}$
Output: *Spectral* GMM $\Lambda_n^{spec} = \{u_{n,i}, R_{n,i}\}_i$
foreach *EM iteration (l)* **do**

1. Compute the weights $\gamma_i^{(l)}(\tau)$ such that $\sum_i \gamma_i^{(l)}(\tau) = 1$ and

$$\gamma_i^{(l)}(\tau) \propto u_{n,i}^{(l)} N(A_{n_\tau}^{train}; R_{n,i}^{(l)})\,, \tag{14}$$

where $\propto$ means proportionality and $N(.)$ is expressed by equation (13).

2. Update the weights of the Gaussians $u_{n,i}$:

$$u_{n,i}^{(l+1)} = \frac{1}{T}\sum_\tau \gamma_i^{(l)}(\tau)\,. \tag{15}$$

3. Update the covariance matrices $R_{n,i}$:

$$r_{n,i}^{2\ (l+1)}(f) = \frac{\sum_\tau \gamma_i^{(l)}(\tau)|A_n^{train}(\tau, f)|^2}{\sum_t \gamma_i^{(l)}(\tau)}\,. \tag{16}$$

end

The method used for the **audio separation** is explained in Algorithm 2 for a mixture of $N = 2$ sources. This is done for simplicity and the procedure can be generalized to a higher number of sources. Thus, for each time instant we look for the most suitable couple of states given the mixture spectrum. This information is used to build a time-frequency Wiener mask for each source (19) by combining the *spectral* PSDs in the corresponding states $(r_{1,i^*(\tau)}^2, r_{2,j^*(\tau)}^2)$ with the knowledge about the sources activity w_n. When only one source is active, this weight w_n assigns all the soundtrack to this speaker. Otherwise, $w_n = 0.5$ and the analysis takes into account only the audio GMMs. In a further implementation we could assign intermediate values to w_n that account for the degree of correlation between audio and video. However, such cross-modal correlation has to be accurately estimated to avoid the introduction of separation errors.

VII. BAVSS PERFORMANCE MEASURES

A. Sources activity detection

The performance of the proposed method is highly related to accuracy in the estimation of the temporal periods in which each source is active *alone*. For our method, it is not fundamental to detect *all* the time instants during which sources are active alone, provided that the length of the detected

Algorithm 2: Single-channel Audio Source Separation using knowledge about sources activity

Input: Mixture x, Spectral GMMs $\Lambda_n^{spec} = \{u_{n,i}, R_{n,i}\}_i$ and activity vectors w_n for the sources $n = 1, 2$

Output: Estimation of the sources audio part $\hat{a}_1$ and $\hat{a}_2$

A. Compute the STFT of the mixture $X(\tau, f)$ from the temporal signal x ;

foreach $\tau = 1, 2, \ldots, T$ **do**

1. Find the best combination of states (PSD) according to the mixture spectrum X_τ, that is

$$(i^*(\tau), j^*(\tau)) = \underset{(i,j)}{\operatorname{argmax}} \gamma_{ij}(\tau)\,, \tag{17}$$

where $\gamma_{ij}(\tau)$ is the probability of choosing the combination of states (i, j) at time τ for the observation X_τ with $\sum_{ij} \gamma_{ij}(\tau) = 1$ and

$$\gamma_{ij}(\tau) \propto u_{1,i} u_{2,j} N(X_\tau; R_{1,i} + R_{2,j})\,. \tag{18}$$

2. Build a time-frequency local mask using knowledge about sources activity. For source $n = 1$:

$$M_1(\tau, f) = \frac{r^2_{1,i^*(\tau)}(f) \cdot w_1(\tau)}{r^2_{1,i^*(\tau)}(f) \cdot w_1(\tau) + r^2_{2,j^*(\tau)}(f) \cdot w_2(\tau)}\,, \tag{19}$$

and then $M_2(\tau, f) = 1 - M_1(\tau, f)$.

3. Apply the local masks to the mixture $X(\tau, f)$ to obtain the estimated source STFT:

$$\hat{A}_n(\tau, f) = M_n(\tau, f) X(\tau, f)\,. \tag{20}$$

end

B. Reconstruct estimations of the sources audio part in the temporal domain $\hat{a}_n$ from the STFT estimations $\hat{A}_n$

period is long enough to train the source audio models. In fact, errors occur only when our algorithm estimates that one source is active alone while in fact some of the other sources are active too. In these error frames our algorithm will learn an audio model for source S_i that represents the frequency behavior of several sources mixed, and that will cause errors in the separation. Two measures assess the performance of our method in this domain: the *activity-error-rate* (ERR) and the *activity-efficiency-rate* (EFF).

Let N be the number of audio-visual sources and F_T be the number of video frames. For any fixed time and source S_i we define:

$$S_i^{\mathrm{ON}} \;:=\; \text{“Source } S_i \text{ is active”}\,, \tag{21}$$

$$S_i^{\mathrm{OFF}} \;:=\; \text{“Source } S_i \text{ is NOT active”}\,. \tag{22}$$

Let S_j with $j = 1, \ldots, N\,, i \neq j$ be the set of sources different from S_i . Then we define:

$$E_{j\neq i}^{\mathrm{OFF}} \;:=\; \mathrm{AND}\,\{S_j^{\mathrm{OFF}}\ \forall j \neq i\}\,, \tag{23}$$

$$E_{j\neq i}^{\mathrm{ON}} \;:=\; \mathrm{NOT}\,\{E_{j\neq i}^{\mathrm{OFF}}\} = \mathrm{OR}\,\{S_j^{\mathrm{ON}}\ \forall j \neq i\}\,. \tag{24}$$

$E_{j\neq i}^{\mathrm{OFF}}$ is the event where *all* sources different from S_i are inactive and $E_{j\neq i}^{\mathrm{ON}}$ is the complementary event where *one or more* of the sources different from source S_i are active.

The *activity-error-rate* (ERR) for source S_i is defined as

$$\underset{i}{\mathrm{ERR}} = \frac{F(S_i^{\mathrm{ON}}\ \mathrm{AND}\ E_{j\neq i}^{\mathrm{OFF}} | E_{j\neq i}^{\mathrm{ON}})}{F_T}\,, \tag{25}$$

where $F(A|B)$ is a function that returns the number of frames where our algorithm estimates that the event A has place and the ground truth soundtracks indicate that the current event is B. Thus, the ERR represents the percentage of time during which the algorithm makes an important error since it decides that source S_i is active alone and it is not true (one or more of the other sources are active too).

The *activity-efficiency-rate* (EFF) for source S_i is defined as

$$\underset{i}{\mathrm{EFF}} = \frac{F(S_i^{\mathrm{ON}}\ \mathrm{AND}\ E_{j\neq i}^{\mathrm{OFF}} | S_i^{\mathrm{ON}}\ \mathrm{AND}\ E_{j\neq i}^{\mathrm{OFF}})}{F_i}\,, \tag{26}$$

where F_i is the number of frames where source S_i is active alone. Thus, the EFF represents the percentage of time in which a source is active alone that our method is able to detect. This parameter is very important parameter given the short duration of the analyzed sequences: the higher is EFF, the longer is the period during which we learn the source audio models and, consequently, we can expect to obtain better results on the audio separation part.

B. Audio source separation

The BSS Evaluation Toolbox is used to evaluate the performance of the proposed method in the Audio Separation part. The estimated audio part of the sources $\hat{a}_n$ is decomposed into: $\hat{a}_n = a_{\text{target}} + e_{\text{interf}} + e_{\text{artif}}$, as described in [27]. a_{target} is the target audio part of the source and e_{interf} and e_{artif} are, respectively, the interferences and artifacts error terms. These three terms should represent the part of $\hat{a}_n$ perceived as coming from the wanted source a_n, from other unwanted sources $(a_{n'})_{n'\neq n}$ and from other causes. Two quantities are computed using this toolbox, the source-to-interferences ratio (SIR), and the sources-to-artifacts ratio (SAR), defined as:

$$\mathrm{SIR} \;=\; 10 \log_{10} \frac{\|a_{\text{target}}\|^2}{\|e_{\text{interf}}\|^2} \tag{27}$$

$$\mathrm{SAR} \;=\; 10 \log_{10} \frac{\|a_{\text{target}} + e_{\text{interf}}\|^2}{\|e_{\text{artif}}\|^2} \tag{28}$$

Thus, the SIR measures the performance of our method in the rejection of the interferences and the SAR quantifies the presence of distortions and “burbling” artifacts on the separated audio sources. By combining SIR and SAR one can be sure of eliminating the interfering source without introducing too many artifacts in the separated soundtracks.

For a given mixture and using the knowledge about the original audio part of the sources a_n, oracle estimators for single-channel source separation by time-frequency masking are computed using the BSS Oracle Toolbox. These oracle estimators are computed using the ground truth waveforms in order to result in the smallest possible distortion. As a result, SIR_{oracle} and SAR_{oracle} establish the upper bounds for the proposed performance measures. For further details about the oracles estimation, please refer to [28].

Finally, in order to compare our results to those obtained in [18], we compute the preserved-signal-ratio (PSR) for source S_i using the method described in [29] as

$$\mathrm{PSR} = \frac{\|M_i(\tau, f)a_i(\tau, f)\|^2}{\|a_i(\tau, f)\|^2}, \tag{29}$$

where $a_i(\tau, f)$ is STFT of the original audio signal corresponding to source S_i and $M_i(\tau, f)$ is the time-frequency mask estimated using equation (19) and used in the audio demixing process. Thus, this measure represents the amount of acoustic energy that is preserved after the separation process.

VIII. Experiments

In a first set of experiments (Section VIII-A), the proposed BAVSS algorithm is evaluated on synthesized audio-visual mixtures composed of two persons speaking in front of a camera. These sequences present an artificial mixture generated by temporally shifting the audio and video signals corresponding to one of the speakers so that it overlaps with the speech of the other person. The performance of the proposed method in identifying the number of sources in the scene, locating them the image and determining the activity periods of each one of them is assessed. Furthermore, a *quantitative* evaluation of the algorithm's results in terms of audio separation is performed since the original soundtracks (ground truth) of each speaker separately are available for these sequences.

As explained before, at present only two other methods have attempted a complete audio-visual source separation [18], [19]. The method presented in [19] does not provide any qualitative or quantitative result in terms of audio separation. In fact, this paper is mostly concentrated in the localization of the sources in the image and the only reference to the audio separation part states that the quality of the separated soundtracks is not good. Regarding the method presented in [18], two measures are used to evaluate quantitatively its performance in the audio separation part: the improvement of the signal-to-interference ratio (SIR) and the preserved-signal-ratio (PSR). In the last part of Section VIII-A these two quantities are used to compare our results to those obtained by the approach in [18] when analyzing sequences composed of two speakers.

In Section VIII-B we present a second set of experiments in which speakers and music instruments are mixed. The complexity of the sequences is higher given the more realistic background and the presence of distracting motion. These sequences are real audio-visual mixtures where both sources are recorded at the same time. Thus, it is not possible to obtain a quantitative evaluation of the algorithm's performances as in Section VIII-A since the audio ground truth is not available in this case. The main objective of Section VIII-B is to demonstrate *qualitatively* that our BAVSS method can deal successfully with complex real-world sequences involving speech and music instruments.

Videos showing all the experiments and the estimated audio-visual sources after applying our method are available online at *http://lts2www.epfl.ch/~llagoste/BAVSSresults.htm*.

A. CUAVE Database: Quantitative Results

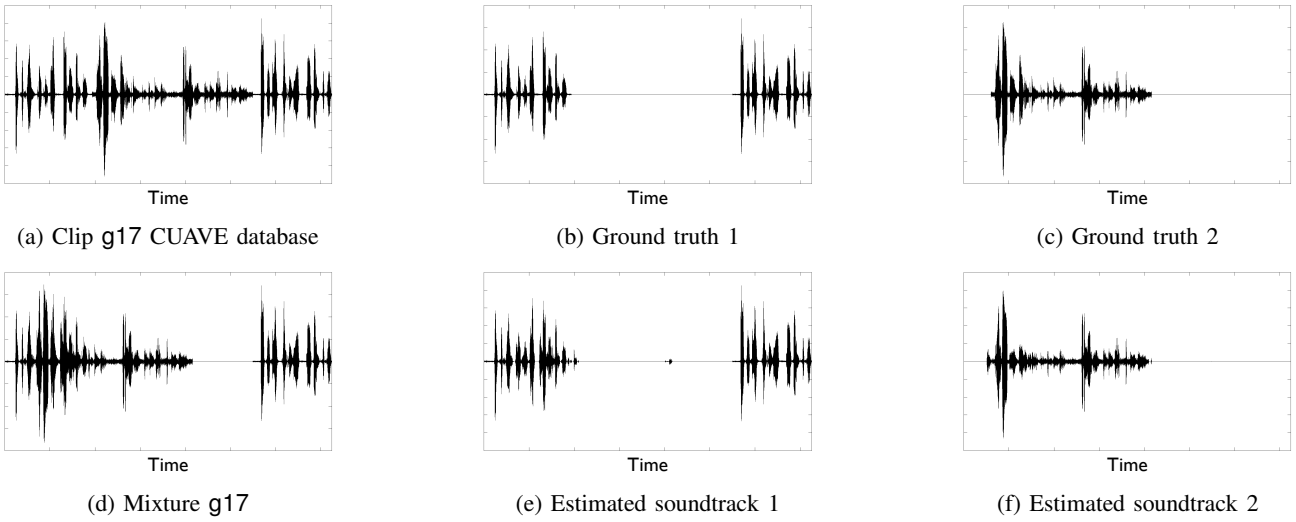
Sequences are synthesized using clips taken from the *groups* partition of the CUAVE database [17] with two speakers uttering sequences of digits alternatively. A typical example sequences is shown in Fig. 1. The video data is sampled at 29.97 frames/sec with a resolution of 480×720 pixels, and the audio 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 $M = 100$ video atoms and the soundtrack is decomposed into $K = 2000$ atoms. The number of atoms extracted from the decomposition does not need to be set a priori. It can be automatically chosen setting a threshold on the reconstruction quality.

Ground truth mixtures are obtained by temporally shifting audio and video signals of one speaker in order to obtain time slots with both speakers active simultaneously. In the resulting synthetic clips, four cases are represented: both persons speak at the same time, only the boy or the girl speaks or silence. For further details on the procedure adopted to build the synthetic sequences the reader is referred to [24]. An example of this procedure on the audio part is shown on Fig. 8. In (a) the figure shows the original clip g17 of CUAVE database, in (b) the ground truth for source 1 (which is the period during which speaker 1 is uttering numbers) and in (c) the ground truth for source 2 which is obtained by shifting its audio part. In Fig. 8 (d) we can see the input to our algorithm, a mixture built by adding ground truth waveforms 1 and 2.

Figure 8 also gives a first *qualitative* evaluation of our method. It is possible to compare the ground truth to the estimated audio part of the sources separated using the proposed method (Fig. 8 (e)-(f)). Waveforms are very similar and the audible quality of the estimated sequences is also remarkable. The separation of the mixture when both sources are active is good as the numbers that each speaker is uttering are clearly understandable at a good quality.

Results obtained when analyzing ten different synthesized audio-visual sequences from CUAVE database are summarized in Table I. In all cases the number of sources present in the scene and their position in the image has been correctly detected. An OK in third column means that the estimated position of the video source is always over the video part of the source and never over the background or the other source.

As explained before, two measures are used to evaluate the performance of our method in determining the time slots where sources are active alone. Results in table I show that in all sequences the error rate (ERR) is under the 10%, and only in four cases we are over the 3%. Errors are concentrated in the boundaries of the source activity, that is just before the person starts to speak or after he/she stops, because in general motion in the video signal is not completely synchronous with sounds in the audio channel. Concerning our method's efficiency (EFF), only in three cases we are able to detect less than 50% of periods where sources are alone, and we average a 69%, which is a high percentage if we think about longer sequences. Low values for EFF are caused by the presence of video motion correlated to the audio on the source that is not active. In fact, it is difficult to detect the complete periods



(a) Clip g17 CUAVE database (b) Ground truth 1 (c) Ground truth 2

(d) Mixture g17 (e) Estimated soundtrack 1 (f) Estimated soundtrack 2

Fig. 8. Comparison between real (b)-(c) and estimated (e)-(f) soundtracks when analyzing a synthetic sequence (d) generated by applying a temporal shift to speaker 2 in clip g17 of CUAVE database (a).

when sources are active alone without introducing errors, since there is a trade-off between them. If we choose to detect all the periods (EFF increases), more false positives will appear (ERR increases too) and, as explained before, the models for each source will not be correct. Here we prefer to have a high confidence when we decide that one source is active alone, even if then the efficiency decreases.

A 100% on EFF means that periods in which the source is active alone are perfectly detected. In this case, *blind* results for SIR and SAR are the best results that we can achieve using the GMM-based audio separation method in Section VI-B since the training sequences are as long as possible. Consequently, the upper bounds for the performance in the blind separation of the audio track are clearly conditioned by the duration of the training sequences and the algorithm we use for the one microphone audio separation. While in some sequences the GMM-based separation seems suitable with performances up to 29dB of SIR (sequence g17), for some speakers this does not seem to be the case (8dB of SIR in sequence g12 even if the combined EFF for both speakers is 81%). However, taking into account the short duration of the analyzed sequences (20-30 seconds) and the training sequences (less than 8 seconds), results are satisfactory. Remember that the oracles in Table I represent the best results that we can obtain through *any* audio source separation method based on frequency masking *if we know in advance the ground truth soundtracks*. In fact, oracles guarantee the minimum distortion by computing the optimal time-frequency mask given the original separated soundtracks. The average SIR that we obtain (16dB) is slightly better than the state-of-the-art on single-channel audio separation [20] and, unlike this method, we do it without any kind of supervision. As explained before, the combination between audio and video signals in our approach eliminates the necessity of knowing *in advance* the sources in the mixture and its acoustic characteristics, which is typical in one microphone audio separation methods. Furthermore, in all the resulting separated soundtracks here, even the ones that present worse SIR, the numbers that each speaker utters can be well understood.

In sequence g15 we can observe a major problem: there is no detected period when speaker 2 is active alone (see EFF in Table I). Consequently, it is not possible to train a model of that source and our separation method cannot be applied. This happens because there is video motion correlated to the audio on source 1 (which is inactive) all over the duration of the period during which only source 2 is active. However, we can expect that with longer sequences (and longer time slots with each source active alone) this problem does not appear anymore, since in that case it is unlikely that correlated video motion is present on the inactive source all the time.

The audio separation task is extremely challenging for sequences g14 and g19, since in this case the mixture is composed by two male speakers. The fundamental frequencies of the speakers are extremely close and, as a result, their formants energy is highly overlapped in the spectrogram. Even in this difficult context, quantitative results (with an average SIR of 17dB) are close to those obtained when analyzing sequences with a male-female combination.

The comparison between our method and the approach in [18] presents some difficulties. First, the test set in [18] is composed of three very short sequences (duration ranging between 5 and 10 seconds), and only one of those sequences contains a mixture composed of speakers. Furthermore, they avoid distracting motion by locating the the camera close to the speakers faces, i.e. we can only observe the lips in the video corresponding to the male speaker. Although the differences are considerable, here we compare the results in the speakers sequence in [18] with the mean results through all the sequences that we have analyzed. In [18], they report an improvement in the SIR of 14dB and a PSR of 57.5% (those values represent the mean between the male and female results). Here we obtain an average SIR of 16dB and an average PSR of 85%. Thus, our approach compares specially favorable in terms of PSR, that is the amount of acoustic energy that is preserved after the separation process. In fact, when demixing the audio part of the sources our methods

Sequence	Source	Position in the image	Activity accuracy (%) ERR	EFF	SIR (dB) blind	oracle	SAR (dB) blind	oracle	PSR (%)
g12	$n=1$	OK	0	74	14	33	4	19	83
	$n=2$	OK	2	87	8	32	7	19	92
g13	$n=1$	OK	3	64	10	36	4	21	66
	$n=2$	OK	0	63	11	37	5	21	87
g14*	$n=1$	OK	6	95	13	39	9	24	100
	$n=2$	OK	0	73	25	39	4	22	65
g15	$n=1$	OK	3	68					
	$n=2$	OK	0	0					
g16	$n=1$	OK	8	45	10	37	7	22	100
	$n=2$	OK	2	82	18	38	3	21	56
g17	$n=1$	OK	1	95	20	40	11	23	95
	$n=2$	OK	0	83	29	39	11	24	94
g18	$n=1$	OK	0	52	24	38	6	23	84
	$n=2$	OK	10	69	12	38	7	22	94
g19*	$n=1$	OK	6	44	15	33	7	19	86
	$n=2$	OK	0	52	15	32	5	18	84
g20	$n=1$	OK	0	90	20	35	9	21	88
	$n=2$	OK	0	77	19	36	9	21	86
g21	$n=1$	OK	0	64	16	38	6	23	87
	$n=2$	OK	1	100	13	38	7	23	90
MEAN			2	69	16	37	7	21	85

TABLE I

RESULTS OBTAINED WITH SYNTHETIC SEQUENCES GENERATED FOR DIFFERENT CLIPS OF CUAVE DATABASE. SEQUENCES MARKED WITH AN ASTERISK (*) PRESENT TWO MALE SPEAKERS INSTEAD OF ONE MALE AND ONE FEMALE. COLUMNS 1, 2, 3 REPRESENT RESPECTIVELY THE ANALYZED SEQUENCE, THE NUMBER OF DETECTED AUDIO-VISUAL SOURCES AND IF THE POSITION IN THE IMAGE ESTIMATED BY THE ALGORITHM IS CORRECT. IN COLUMN 4 TWO QUANTITIES THAT EVALUATE THE ACCURACY OF OUR METHOD IN DETECTING THE PERIODS IN WHICH SOURCES ARE ACTIVE ALONE: THE ERROR RATE [LEFT] AND THE EFFICIENCY RATE [RIGHT]. COLUMNS 5 AND 6 SHOW A QUANTITATIVE COMPARISON BETWEEN RESULTS ON AUDIO SEPARATION OBTAINED USING OUR BLIND METHOD [LEFT] AND ORACLES COMPUTED USING GROUND TRUTH SOUNDTRACKS [RIGHT]. COLUMN 7 PRESENTS THE PERCENTAGE OF ENERGY FROM THE ORIGINAL SOUNDTRACK THAT IS KEPT AFTER THE AUDIO SEPARATION PROCESS.

keeps the 85% of the energy in the original audio signal while in [18] more than the 40% of this energy is lost. These results are related to the audio separation method used in each case: our GMM-based separation seems more suitable than the frequency tracking used in [18] when we consider the PSR.

B. LTS Database: Qualitative results in a challenging environment

More challenging sequences including speakers and music instruments have been recorded in order to *qualitetively* test the performance of the proposed method when dealing with complex situations. The original video data is sampled at 30 frames/sec with a resolution of 240×320 pixels, and the audio at 44 kHz. For its analysis, the video has been resized to a 120×160 pixels, while the audio has been sub-sampled to 8 kHz. The length of the sequences is close to 1 minute in this case. The video signal is decomposed into $M = 120$ atoms and the soundtrack is decomposed into $K = 6000$ atoms. As explained before, a quantitative evaluation can not be performed in this case since in this section we consider real mixtures where both sources are recorded at the same time.

In the first experiment (*movie1*) we analyze an audio-visual sequence where two persons are playing music instruments in front of a camera. A frame of this movie is shown in Fig. 9. In some temporal periods they play at the same time while in others they do a *solo*. A first difficulty is given

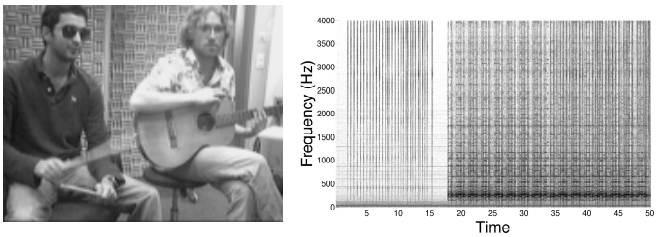


Fig. 9. Challenging audio-visual sequence where one person is playing a guitar and another one is hitting two drumsticks in a complex background. A frame of this movie [left] and the corresponding audio spectrogram [right] are represented. Drumsticks are active in the begining of the sequence, then the guitarist starts to play and finally both instruments are mixed.

by the fact that the video decomposition has to reflect the movement of the present structures, which is not an easy task when trying to model the drumsticks and their trajectory. Thus, while the hand that is playing the guitar moves in a smooth way, drumsticks movement is much more fast and abrupt. Another problem are some movements correlated with the sound, specially those of the guitarist's leg, and the proximity of the sources. If we compare this sequence with the ones presented in the literature we can see that, in those cases, either the sources are much more separated in the image [19] or distracting motion is avoided by visually zooming into the sources [18]. Furthermore, these methods always present flat, or almost flat, backgrounds. Here the complex background (see

Fig. 10. Video sources reconstruction for *movie1*. The atoms that are highlighted in the images are those that characterize the left source [left] and the right source [right] respectively. Background is composed of the residual energy after the 3D-MP video decomposition and provides an easier visualization of the reconstructed sources. Finally, crosses mark the position in the image where our algorithm locates the sources.

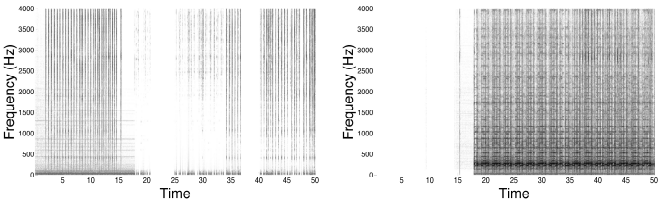


Fig. 11. Estimated spectrograms for drumsticks [left] and guitar [right] in *movie1*. Drumsticks are silent in the middle of the sequence and the guitar at the beginning. Spectrograms show that the sources behavior is correctly detected by the proposed method.

(a) (b)

Fig. 12. Two frames belonging to *movie2* (a) and *movie3* (b). On both frames, one person is uttering numbers while a guitarist is playing. Frame (b) shows the distracting motion caused by a person who is crossing the scene behind the sources. The estimated source positions are marked with crosses.

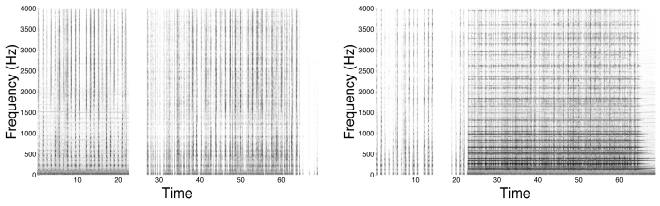


Fig. 13. Estimated spectrograms for speech [left] and guitar [right] in *movie2*. In the first part the speaker is uttering numbers alone, next there is a short period where the guitar starts to play while the speaker is silent and in the last part both sources are mixed.

Fig. 9) makes the video decomposition task more complicated since a considerable part of the video atoms has to be used to represent it.

When analyzing *movie1* with the proposed BAVSS method, the number of sources and its position in the image are perfectly detected (see crosses on Fig. 10). A reconstruction of the image using the atoms assigned to each source is shown in Fig. 10. In the left picture it is possible to see how the stick is successfully represented by one video atom, and in the right one, the atoms that surround the guitar are highlighted. In this sequence, the activity periods of each source are also detected. A good characterization of the sources in the frequency domain is achieved, which leads to a satisfactory audio separation of the sources. Figure 11 shows the spectrograms that we obtain. We can see that drumsticks sounds [left] are much more sharp in the spectrogram (well-localized in time, broad range in frequency) while the guitar spectrogram [right] has much more energy and it is composed by several harmonic sounds. Concerning the audible quality of the estimated soundtracks, the audio part of the drumsticks is perfectly reconstructed at the beginning and it only presents some distortion at the end, where they are mixed with the guitar sounds. In addition, it is almost impossible to hear the guitar in the drumsticks soundtrack. Finally, the quality of the guitar reconstruction is good even though there are some attenuated drumstick sounds in the last part.

Second and third experiments are very similar. They present an audio-visual mixture composed of speech and guitar sounds. In *movie2* a male speaker is uttering numbers (Fig. 12(a)), while in *movie3* there is a female speaker and another person crosses the scene generating thus distracting motion (Fig. 12(b)). These sequences share one challenging difficulty, the fact that acoustic energy of the guitar is considerably stronger than the energy coming from the speech. Furthermore, it is not possible to equalize the energies of both sources since they are recorded at the same time.

Results obtained when analyzing these two sequences are similar. The number of present sources and their spatial position are correctly determined (see crosses in Fig. 12). Despite of not detecting the whole periods during which each source is active alone, the periods that we detect are correct and long enough to represent the sources frequency behavior. Finally, concerning the audio separation part, even though the speakers estimated soundtracks are pretty clean, in the case of the guitar we can still hear speech. A first reason for this behavior is the unbalanced energy between sources that we discussed before. Another one, and maybe the main one, could be the fact that the guitar sounds present many harmonics that overlap with speech in the spectrogram. Thus, some frequency formants of speech are also characterized in the acoustic model of the guitar and we can not eliminate them in the audio separation part using this separation method.

Spectrograms of the estimated audio part of the sources for *movie2* can be observed in Fig. 13. We can observe that the short time slot where the guitar is active alone is perfectly detected (between seconds 22 and 27) since it is not present in the speaker spectrogram [left]. It is also possible to see the residual energy of the speech signal that remains in the first part of the guitar spectrogram [right].

Even if the distracting motion present on *movie3* (Fig. 12(b)) seems not to affect the performance of the proposed method, results concerning the audio separation are slightly worse in this case. However, since the activity periods for the sources are also correctly detected, this degradation in performance

cannot be due to the background motion but rather to the fact that female harmonics overlap more often with the guitar ones in the spectrogram.

IX. Discussion

In this paper we have introduced a novel algorithm to perform Blind Audio-Visual Source Separation. We consider sequences made of one audio signal and the associated video signal, without the stereo audio track usually employed for the audio source separation task. The method correlates salient acoustic and visual structures that are represented using atoms taken from redundant dictionaries. Video atoms synchronous with the audio track and that are spatially close are grouped together using a clustering algorithm that counts and localizes on the image plane audio-visual sources. Then, using this information and exploiting the coherence between audio and video signals, the audio activity of the sources is determined and its audio part is separated and reconstructed.

One of the contributions of this paper is an extensive evaluation of the proposed method on sequences involving speakers and music instruments. This systematic study of the algorithm performances represents a sensible improvement with respect to previously published works in [18], [19] that test algorithms' performances on few, very short sequences. Here, a first set of experiments has been performed on synthetic sequences built from CUAVE database in which two persons utter numbers in front of a camera. In all cases, the scene has been well interpreted by our algorithm, leading to state-of-the-art audio-visual source separation. The audible quality of the separated audio signals is good. A rigorous evaluation of the audio separation results has been performed using the BSS Evaluation Toolbox. These quantitative results do not show any significant difference between sequences where two male speakers are mixed and those where a male and a female appear. A second set of tests has been performed on more realistic sequences where speakers are mixed with music instruments. Even if the nature of this second set of sequences does not allow a quantitative evaluation of the results, we have demonstrated that the proposed BAVSS method is able to deal with less static sources, complex backgrounds and distracting motion representing a much more realistic environment. Given the short length of the analyzed sequences, a possible improvement for the audio separation part could be the adaptation of a general acoustic model to the detected sources as explained in [25].

References

[1] Q. Summerfield, "Some preliminaries to a comprehensive account of audio-visual speech perception," in *Hearing by Eye: The Psychology of Lipreading*, B. Dodd and R. Campbell, Eds. Lawrence Erlbaum Associates, 1987, pp. 3–51.

[2] W. H. Sumby and I. Pollack, "Visual contribution to speech intelligibility in noise," *Journal of the Acoustical Society of America*, vol. 26, no. 2, pp. 212–215, 1954.

[3] L. Girin, J.-L. Schwartz, and G. Feng, "Audio-visual enhancement of speech in noise," *Journal of the Acoustical Society of America*, vol. 109, no. 6, pp. 3007–3020, 2001.

[4] S. Deligne, G. Potamianos, and C. Neti, "Audio-visual speech enhancement with AVCDCN (Audiovisual Codebook Dependent Cepstral Normalization)," in *Proc. Int. Conf. Spoken Language Proc. (ICSLP)*, 2002, pp. 1449–1452.

[5] R. Goecke, G. Potamianos, and C. Neti, "Noisy audio feature enhancement using audio-visual speech data," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, 2002, pp. 2025–2028.

[6] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," *Proc. IEEE*, vol. 91, no. 9, pp. 1306–1326, 2003.

[7] S. Lucey, T. Chen, S. Sridharan, and V. Chandran, "Integration strategies for audio-visual speech processing: applied to text-dependent speaker recognition," *IEEE Trans. Multimedia*, vol. 7, no. 3, pp. 495–506, 2005.

[8] 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.

[9] R. Dansereau, "Co-channel audiovisual speech separation using spectral matching constraints," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, vol. 5, 2004, pp. 645–648.

[10] S. Rajaram, A. V. Nefian, and T. Huang, "Bayesian separation of audio-visual speech sources," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, vol. 5, 2004, pp. 657–660.

[11] B. Rivet, L. Girin, and C. Jutten, "Mixing audiovisual speech processing and blind source separation for the extraction of speech signals from convolutive mixtures," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 15, no. 1, pp. 96–108, 2007.

[12] W. Wang, D. Cosker, Y. Hicks, S. Saneit, and J. Chambers, "Video assisted speech source separation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, vol. 5, 2005, pp. 425–428.

[13] G. Chetty and M. Wagner, "Audio visual speaker verification based on hybrid fusion of cross modal features," in *Pattern Recognition and Machine Intelligence (PReMI)*, 2007, pp. 469–478.

[14] H. G. Okuno and K. Nakadai, "Real-time sound source localization and separation based on active audio-visual integration." in *IWANN (1)*, ser. Lecture Notes in Computer Science, J. Mira and J. R. Álvarez, Eds., vol. 2686. Springer, 2003, pp. 118–125.

[15] J. Fritsch, M. Kleinehagenbrock, S. Lang, G. A. Fink, and G. Sagerer, "Audiovisual person tracking with a mobile robot," in *In Proc. Int. Conf. on Intelligent Autonomous Systems*. IOS Press, 2004, pp. 898–906.

[16] C. Saraceno and R. Leonardi, "Indexing audiovisual databases through joint audio and video processing," *International Journal of Imaging Systems and Technology*, vol. 9, no. 5, pp. 320–331, 1999.

[17] 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 Journal on Applied Signal Processing*, vol. 2002, no. 11, p. 1189, Nov. 2002.

[18] Z. Barzelay and Y. Y. Schechner, "Harmony in motion." in *CVPR*. IEEE Computer Society, 2007.

[19] C. Sigg, B. Fischer, B. Ommer, V. Roth, and J. Buhmann, "Nonnegative cca for audiovisual source separation," in *IEEE Workshop on Machine Learning for Signal Processing*. IEEE Press, 2007.

[20] L. Benaroya and F. Bimbot, "Wiener based source separation with HMM/GMM using a single sensor," in *Proc. 4th Int. Symp. on Independent Component Anal. and Blind Signal Separation (ICA2003)*, Nara, Japan, Apr. 2003, pp. 957–961.

[21] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[22] O. Divorra Escoda, G. Monaci, R. Figueras i Ventura, P. Vandergheynst, and M. Bierlaire, "Geometric video approximation using weighted matching pursuit," *IEEE Transactions on Image Processing*, vol. 18, no. 8, pp. 1703–1716, 2009.

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

[24] A. Llagostera Casanovas, G. Monaci, and P. Vandergheynst, "Blind audiovisual source separation using sparse redundant representations," EPFL, LTS-REPORT-2007-001, 2007. [Online]. Available: http://infoscience.epfl.ch/record/99671/files/

[25] 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.

[26] R. Baken and R. Orlikoff, *Clinical Measurement of Speech and Voice*, 2nd ed. San Diego, CA: Singular Publishing Group Thomson Learning, 2000.

[27] 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.

[28] E. Vincent, R. Gribonval, and M. D. Plumbley, "Oracle estimators for the benchmarking of source separation algorithms," *Signal Processing*, vol. 87, no. 8, pp. 1933–1950, 2007.
[29] O. Yilmaz 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.