View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Infoscience - École polytechnique fédérale de Lausanne

SCHOOL OF ENGINEERING - STI
SIGNAL PROCESSING INSTITUTE
Gianluca Monaci

ELD 241 (Bâtiment ELD)
Station 11
CH-1015 LAUSANNE

Tel: +41 21 693 2657
Fax: +41 21 693 7600
e-mail: gianluca.monaci@epfl.ch

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

TRACKING ATOMS WITH PARTICLES

Gianluca Monaci, Pierre Vandergheynst, Emilio Maggio, Andrea Cavallaro

École Polytechnique Fédérale de Lausanne (EPFL)

Signal Processing Institute Technical Report

TR-ITS-2006.11

October 2, 2006

Tracking Atoms with Particles

Gianluca Monaci, Pierre Vandergheynst
Signal Processing Institute
École Polytech. Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland

Emilio Maggio, Andrea Cavallaro
Dept. of Electronic Engineering
Queen Mary, University of London
Mile End Road, London E1 4NS (UK)

Abstract

We present a general framework and an efficient algorithm for tracking relevant video structures. The structures to be tracked are implicitly defined by a Matching Pursuit procedure that extracts and ranks the most important image contours. Based on the ranking, the contours are automatically selected to initialize a Particle Filtering tracker. The proposed algorithm deals with salient video entities whose behavior has an intuitive meaning, related to the physics of the signal. Moreover, as the interactions between such structures are easily defined, the inference of higher level signal configurations can be made intuitive. The proposed algorithm improves the performance of existing video structures trackers, while reducing the computational complexity. The algorithm is demonstrated on audiovisual source localization.

Index Terms

Video signal processing, tracking, feature extraction, audiovisual processing, sparse signal representation.

I. Introduction

Object tracking is usually performed based on an appropriate description of the appearance of a target, either at a global or local level. Examples of global descriptions are simple templates [1], color histograms [2], or active appearance models [3]. Examples of local analysis are the methods developed to independently track and match feature points. The seminal work in this field is the KLT tracker [4] where stable corners are detected and then their appearance is represented by an affine invariant template computed on a small region around the point. The points detected at subsequent frames are matched based on the appearance. More advanced feature point detectors have been proposed to account for rotation, scale changes of the underlying object structures [5]. All the above mentioned methods are designed from a tracking-centric point of view : (i) stable structures are used to facilitate tracking, and (ii) the representation is designed to reduce ambiguity between feature points [6]. The interpretation of the information obtained after tracking in the context of the considered signal is postponed to a subsequent analysis stage. But are stable structures also relevant from a signal representation point of view?

We argue that a signal-centric (as opposed to a tracking-centric) representation can extend the application of a feature tracking system by fusing analysis and tracking in a single general framework. The ability of tracking relevant structures of moving images would provide spatio-temporal information that is intrinsically meaningful for the representation of the video signal. Considering natural image sequences composed of successive 2D projections of 3D objects describing smooth trajectories through time, one usually assumes that image sequences are well modeled by smooth transformations of a reference frame [7]. In this context, relevant video features are time-evolving oriented edges that describe concisely the geometric structures of a scene and their temporal evolution [8]. In general, a large variety of geometric structures can be found in a video sequence. A signal representation capable of exploiting video structural properties while keeping generic and flexible enough should then be used. Such properties are introduced into the video feature extraction process, considering spatio-temporal video approximations using redundant codebooks of geometric primitives called *atoms*. Local deformations are then propagated over time by updating the atoms' parameter field in order to approximate the succession of frames.

An algorithm that aims at representing video sequences as a sum of relevant video structures for coding purposes was proposed in [8]. This method decomposes using Matching Pursuit (MP) a reference frame as a sparse sum

The authors acknowledge the support of the Swiss National Science Foundation through the IM.2 National Center of Competence for Research and of the UK Engineering and Physical Sciences Research Council (EPSRC), under grant EP/D033772/1.

of atoms taken from a redundant dictionary [9]. These structures are then tracked through time, decomposing the subsequent frames with a modified MP algorithm that uses *a priori* information inherited from previous frames [8, 10]. Although effective for audiovisual source localization and separation [11, 12], this video MP algorithm is formally and computationally complex. Here we want to formalize the atom tracking problem in a more agile and well grounded fashion, in order to allow an easier and more intuitive understanding of the results. This should allow as well to improve and extend in a natural and elegant fashion the proposed algorithm, as we will discuss in the last section of the manuscript. We also want the tracking method to employ a strategy that allows to reduce the computational load of the algorithm. In addition, we want to underline that the method introduced in [8] was designed as a coding algorithm. This poses some problems from the tracking point of view. First of all, the parameters of the video atoms were coarsely quantized to achieve better compression performances, introducing tracking errors. Secondly, atoms are followed from one frame to the other using a search window of limited size, since, as in most video coding schemes, it is less expensive to code a new object than to encode the difference between two very different entities. This however limits the robustness and flexibility of the tracker.

In this report, we formalize the atom tracking problem to enable a more intuitive interpretation of the decomposition results and we reduce the computational complexity of the atom tracking scheme. The tracker is automatically initialized by representing the first frame of a sequence as a combination of edge-like functions. These functions are retrieved from a redundant dictionary of atoms using MP. In contrast to classical tracking algorithms, the structures to be tracked are implicitly defined by MP that picks the most relevant image contours. Such visual features are then tracked using one of the most popular tracking algorithm, Particle Filter (PF) [13–15]. In this way we put the video atom tracking problem in the well grounded and understood framework of PF, which moreover ensures robustness, flexibility and lower computational complexity than the video MP algorithm [8].

The structure of the report is the following : Section II presents the geometric video representation framework based on MP and the tracking algorithm based on PF. In Section III experimental results of visual edge tracking and of audiovisual source localization are presented. Finally, in Section IV achievements and future research directions are discussed.

II. Tracking of Geometric Video Features

In the next sections the video representation and tracking algorithm is presented. Section II-A introduces the adopted approach to sparse video representation based on the decomposition of the frames over redundant dictionaries of geometric primitives. Section II-B introduces the tracking strategy adopted to follow the video structures across time based on Particle Filter.

A. Geometric Video Representation

Assuming that an image $I(x, y)$ can be approximated with a linear combination of atoms retrieved from a redundant dictionary $\mathcal{D}_\mathcal{V}$ of 2D atoms, we can write :

$$I(x,y) \approx \sum_{\mathbf{x}[n]\in\Omega} c_{\mathbf{x}[n]} G_{\mathbf{x}[n]}(x,y)\,, \tag{1}$$

where n is the summation index, $c_\mathbf{x}$ corresponds to the coefficient for every atom $G_\mathbf{x}(x, y)$ and Ω is the subset of selected atom indexes from dictionary $\mathcal{D}_\mathcal{V}$. We also require that the representation is *sparse*, i.e. the cardinality of Ω is much smaller than the dimension of the signal. The decomposition of $I(x, y)$ on an overcomplete dictionary is not unique and several decomposition approaches have been proposed, like the method of frames [16], Matching Pursuit [9] or Basis Pursuit [17]. We use here Matching Pursuit, an iterative greedy algorithm that selects the element of the dictionary that best matches the signal at each iteration.

Each video frame is decomposed into a low-pass part, that takes into account the smooth components of images, and a high-pass part, where most of the energy of edge discontinuities lays. The low frequency component is obtained by low-pass filtering and downsampling the images in the sequence, using the Laplacian-pyramid scheme [18]. We employ here the FIR low-pass filter proposed in [19]. The high-pass frames are obtained by subtracting the low frequency parts from the original frames. These high frequency residual frames which contain the geometric structures of images, are represented using MP.

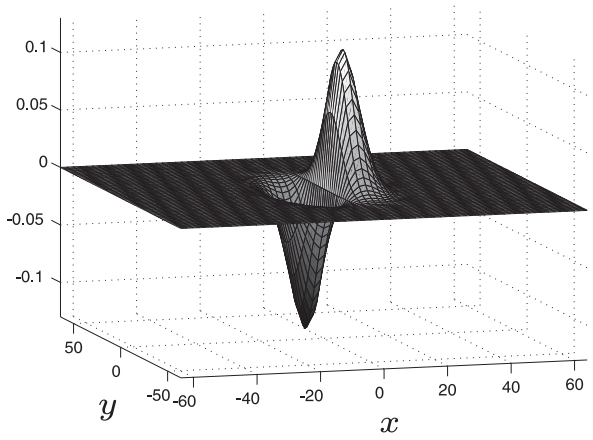

Fig. 1. The generating function $G(x, y)$ described by Eq. 5.

The approach we consider here consists of decomposing a reference frame in terms of geometric 2D primitives and tracking them through time. Thus, starting from the first frame of the sequence, I_1, MP iteratively picks up the function belonging to $\mathcal{D}_\mathcal{V}$ that best approximates the image I_1. The first step of the MP algorithm decomposes I_1 as

$$I_1 = \langle I_1, G_{\mathbf{x}[0]} \rangle G_{\mathbf{x}[0]} + R^1 I_1 \,, \tag{2}$$

where $R^1 I_1$ is the residual component after approximating I_1 in the subspace described by $G_{\mathbf{x}[0]}$. The function $G_{\mathbf{x}[0]}$ is chosen such that the projection $|\langle I_1, G_{\mathbf{x}[0]} \rangle|$ is maximal. At the next step, we simply apply the same procedure to $R^1 I_1$, which yields :

$$R^1 I_1 = \langle R^1 I_1, G_{\mathbf{x}[1]} \rangle G_{\mathbf{x}[1]} + R^2 I_1 \,. \tag{3}$$

This procedure is recursively applied, and after N iterations we approximate I_1 as

$$I_1 \approx \sum_{n=0}^{N-1} c_{\mathbf{x}[n]} G_{\mathbf{x}[n]} \,, \tag{4}$$

where $c_{\mathbf{x}[n]} = \langle R^n I_1, G_{\mathbf{x}[n]} \rangle$.

The dictionary $\mathcal{D}_\mathcal{V}$ is built by varying the parameters of a mother function, in such a way that it generates an overcomplete set of functions spanning the input image space. The choice of the generating function $G(x, y)$ is driven by the observation that it should be able to represent well edges on the 2D plane. Thus, it should behave like a smooth scaling function in one direction and should approximate the edge along the orthogonal one. We use here 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. 1). The generating function $G(x, y)$ is thus expressed as

$$G(x, y) = 2x \cdot e^{-(x^2+y^2)} \,. \tag{5}$$

The codebook of functions $\mathcal{D}_\mathcal{V}$ can be defined as $\mathcal{D}_\mathcal{V} = \{G_\mathbf{x} : \mathbf{x} \in \Gamma\}$. Each atom $G_\mathbf{x} = U_\mathbf{x} g$ is built by applying a set of geometrical transformation $U_\mathbf{x}$ to the mother function $G(x, y)$. Basically, this set has to contain three transformations :

- Translations $\vec{t} = (t_x, t_y)$ all over the image plane.
- Rotations θ to locally orient the function along the edge.
- Anisotropic scaling $\vec{s} = (s_x, s_y)$ to adapt the atom to the considered image structure.

Any atom $G_\mathbf{x}$ in the dictionary rotated by θ, translated by t_x and t_y, and anisotropically scaled by s_x and s_y can thus be written as :

$$G_\mathbf{x}(x, y) = \frac{C}{\sqrt{s_x s_y}} \cdot 2u \cdot e^{-(u^2+v^2)} \,, \tag{6}$$

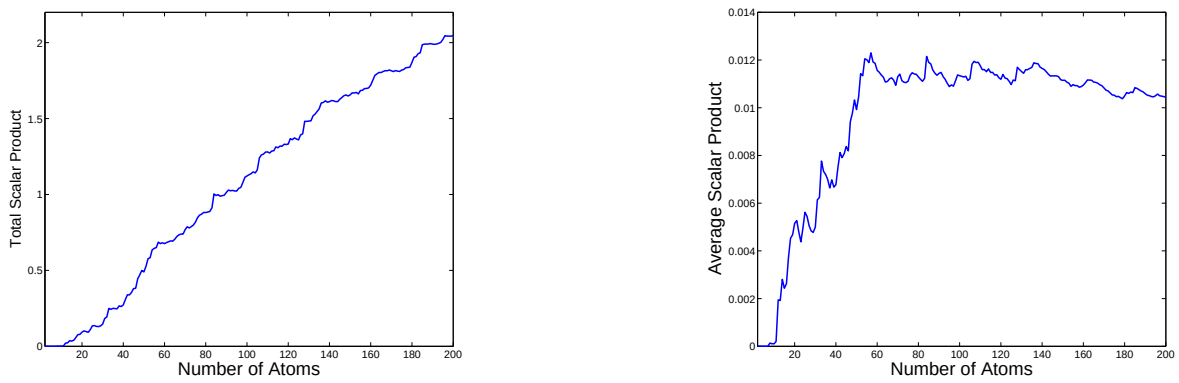

Fig. 2. Sum of scalar products between the atoms representing the first frame of a sequence [Left], and average scalar product [Right], plotted as a function of the number of considered functions.

where C is a normalization constant and

$$u = \frac{\cos\theta(x - t_x) + \sin\theta(y - t_y)}{s_x}\,, \tag{7}$$

and

$$v = \frac{-\sin\theta(x - t_x) + \cos\theta(y - t_y)}{s_y}\,. \tag{8}$$

The reference frame I_1 is thus decomposed into a set of N geometric atoms $G_{\mathbf{x}[n]}(x, y)$ that are tracked through time.

B. Tracking Video Atoms Using Particle Filter

The tracking is performed using Particle Filter (PF), a parametric method which solves non-linear and non-Gaussian state estimation problems [13–15] and can deal with multi-modal *pdf*s. Its robustness and flexibility makes PF one of the most used tracking algorithm.

The reference image is represented with N atoms and the first M atoms are *independently* tracked. This is mainly motivated by the fact that we are interested to the main structures present in the video (i.e., the first functions of the MP decomposition). If few atoms are considered, than their interactions are likely to be weak. One can measure such interactions by computing the scalar products between the atoms. If two atoms exhibit a large scalar product (the atoms have unit norm, thus the maximum scalar product is 1), their interaction is strong, while if it is small (i.e. close to 0), their interaction is weak. Figure 2 shows the sum of the scalar products between the atoms representing the first frame of a sequence [Left], and the average scalar product between atoms [Right], plotted as a function of the number of considered functions. The total scalar product clearly increases with the number of atoms, since there are more interactions between the structures. The average scalar product increases rapidly until when the atoms added to the decomposition become very small since they represent small image details, giving low scalar products with the other functions. In our experiments we will consider the first $M = 30$ atoms selected by MP : as a first approximation, it seems reasonable to consider the atoms independently since the interactions between them are still limited. However, as highlighted in [20], neighboring functions can mutually influence each other and one of the main future research directions will be the design of a method that can account for the interactions between atoms.

Each atom $G_{\mathbf{x}[n]}(x, y)$ is fully characterized by the set of five parameters $\mathbf{x}[n]$, i.e. the position, scale and rotation parameters that describe its shape. Thus each atom to track is an object in a five-dimensional state space. PF solves the tracking problem based on the state equation

$$\mathbf{x}_t[n] = \mathbf{f}_t(\mathbf{x}_{t-1}[n], \mathbf{v}_t)\,, \tag{9}$$

and on the measurement equation

$$\mathbf{z}_t[n] = \mathbf{h}_t(\mathbf{x}_t[n], \mathbf{n}_t)\,, \tag{10}$$

where f_t and h_t are non-linear and time-varying functions. The state variable $\mathbf{x}_t$ describes the characteristics of target n at time t, and thus it defines the n^{th} atom at frame t. To simplify the notation, from now on the atom index n will be omitted, since anyway the atoms are tracked independently. $\{\mathbf{v}_t\}_{t=1,\dots}$ and $\{\mathbf{n}_t\}_{t=1,\dots}$ are assumed to be independent and identically distributed stochastic processes. The problem consists in calculating the *pdf* $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ at each time instant t. This *pdf* can be obtained recursively in two steps, namely prediction and update. The *prediction step* uses the state equation (9) to obtain the prior *pdf* as

$$p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})d\mathbf{x}_{t-1}\,, \tag{11}$$

with $p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$ known from the previous iteration and $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ determined by (9). When the measurement $\mathbf{z}_t$ is available, it is possible to perform the *update step* using the Bayes' rule

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{z}_{1:t-1})}{\int p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{z}_{1:t-1})d\mathbf{x}_t}\,. \tag{12}$$

PF approximates the densities $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ with a sum of N_s Dirac functions centered in $\left\{\mathbf{x}_t^i\right\}_{i=1,\dots,N_s}$ as

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{i=1}^{N_s} \omega_t^i \delta\left(\mathbf{x}_t - \mathbf{x}_t^i\right)\,, \tag{13}$$

where ω_t^i are the weights associated to the particles and they are calculated as

$$\omega_t^i \propto \omega_{t-1}^i \frac{p(\mathbf{z}_t|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i,\mathbf{z}_t)}\,. \tag{14}$$

The function $q(\cdot)$ is the importance density function which is often chosen to be $p(\mathbf{x}_t|\mathbf{x}_{t-1}^i)$, as it is done here. This leads to $\omega_t^i \propto \omega_{t-1}^i p(\mathbf{z}_t|\mathbf{x}_t^i)$.

A re-sampling algorithm can then be applied to avoid the degeneracy problem [13]. In this case the weights are set to $\omega_{t-1}^i = 1/N_s \;\forall\, i$, and therefore

$$\omega_t^i \propto p(\mathbf{z}_t|\mathbf{x}_t^i)\,. \tag{15}$$

The weights are thus proportional to the *likelihood* of the measurement $\mathbf{z}_t$ given the particles. Here the natural choice for the likelihood function is the projection of the candidate atom over the image, since we want to track important video structures, i.e. video atoms exhibiting high projection on the image. This is also coherent with the representational framework formulated in the previous section. The likelihood of a candidate particle is defined as the absolute value of the scalar product between the residual frame and the atom represented by the particle. In order to favor candidates with high likelihood, this quantity is filtered with a Gaussian kernel centered in the maximum likelihood value and with variance $\sigma_{\mathcal{L}}$, obtaining :

$$\mathcal{L}(\mathbf{x}_t^i[n]) = \exp\left(-\frac{(\mathcal{L}_t^M[n] - |\langle R^n I_t, G_{\mathbf{x}_t^i[n]}\rangle|)^2}{2\cdot(\sigma_{\mathcal{L}}\mathcal{L}_t^M[n])^2}\right)\,, \tag{16}$$

with $\mathcal{L}_t^M[n] = \max(|\langle R^n I_t, G_{\mathbf{x}_t^i[n]}\rangle|)\,, i = 1,\dots,N_s$. We want to underline that the atom $G_{\mathbf{x}_t^i[n]}$ is not projected over the frame I_t but over the residual at step n of the decomposition, $R^n I_t$ (see (3)). We will use the function $\mathcal{L}$ to compute the weights ω_t^i. Figure 3 shows the likelihood function of a candidate atom computed on a region extracted from one of the analyzed clips. The re-sampling step derives the particles depending on the weights of the previous step, then all the new particles receive a starting weight equal to $1/N_s$ which will be updated by the next frame filtered likelihood function.

The best state at the time t, $\hat{\mathbf{x}}_t$, is the particle $\mathbf{x}_t^i$ with biggest weight, pondered by a factor that takes into account the similarity of the particle with the corresponding best state at time $t-1$:

$$\hat{\mathbf{x}}_t = \mathbf{x}_t^M \quad \text{s.t.} \quad \omega_t^M = \max(s(\mathbf{x}_t^i, \hat{\mathbf{x}}_{t-1})\cdot\omega_t^i)\,. \tag{17}$$

The function s is a Gaussian in the 5D parameters space. The value of $s(\mathbf{x},\mathbf{y})$ is maximum when the particles $\mathbf{x}$ and $\mathbf{y}$ coincide and it decreases exponentially as the distance between $\mathbf{x}$ and $\mathbf{y}$ in the parameters space increases.

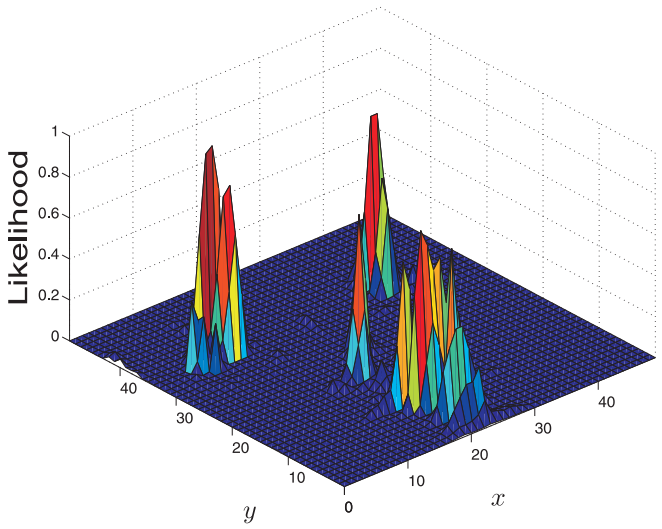

Fig. 3. Likelihood function of a candidate atom computed on a region extracted from one of the analyzed clips. The function is clearly multimodal, exhibiting peaks that have similar amplitude and that are spatially close.

Alternative strategies to compute the best state would be to take the particle with highest weight or to consider the Monte Carlo approximation of equation (13) consisting in estimating the best state as the weighted sum of the particles, as in [13]. However, it was observed that unstable, noisy atom trajectories were generated considering simply the particles with largest weights, due to the multimodality of the posterior *pdf*s, as can be seen in Fig. 3. The Monte Carlo solution would produce more stable atom trajectories. However, in this case there is no guarantee that the best state corresponds to an atom that matches a *real* visual structure, since several local maxima can be present in the likelihood function (Fig. 3). This causes errors due to the fact that when the n^{th} atom is found, it is subtracted, multiplied by its coefficient, to the residual image $R^{n-1}I_t$ to generate the new residual R^nI_t which is used to calculate the successive atoms (see (3)). If the n-th atom is not matching an image structure, its coefficient (i.e. its projection over the residual image) will be very small and thus its contribution to the MP decomposition will not be taken into account, inducing errors in the computation of the successive atoms.

The introduction of the weighting factor $s(\mathbf{x}, \mathbf{y})$ results in a stabilization of the atoms tracks since the algorithm tends to prefer states that are as similar as possible to the previous ones, except if relevant modifications of the structures occur. At the same time, the representation of the scene is kept coherent. An example of PF with re-sampling is shown in Fig. 4.

III. Experiments

In this section we present the results of the atoms tracking algorithm with PF (MP-PF). We test the algorithm on sequences representing one or two persons speaking and moving in front of a camera. The clips used for the tests have been taken from the CUAVE database [21][1]. The video data was recorded at 29.97 fps and at a resolution of 480 $\times$ 720 pixels. The size of the clips has been then reduced to 120 $\times$ 176 pixels. We use a 5-dimensional state model for PF composed of the target position, (x, y), the target size s_x and s_y and the orientation θ. In all experiments a zero-order motion model with fixed $\sigma_{t_x} = \sigma_{t_y} = 2$, $\sigma_{s_x} = \sigma_{s_y} = 0.03$ and $\sigma_\theta = 3.5$. Note that the position change is in pixels while the scale is in percentage and the orientation in degrees. The Gaussian function filtering the likelihood function has $\sigma_{\mathcal{L}} = 0.05$. PF tracker uses 150 samples.

A. Tracking of Video Atoms

In the first experiment, the proposed MP-PF approach is tested on four sequences representing one person speaking and moving in front of the camera and it is compared with the video MP algorithm [8] (3D-MP). Sample frames of two clips are shown in Fig. 5.

[1]Only the luminance component of the video sequences has been considered.

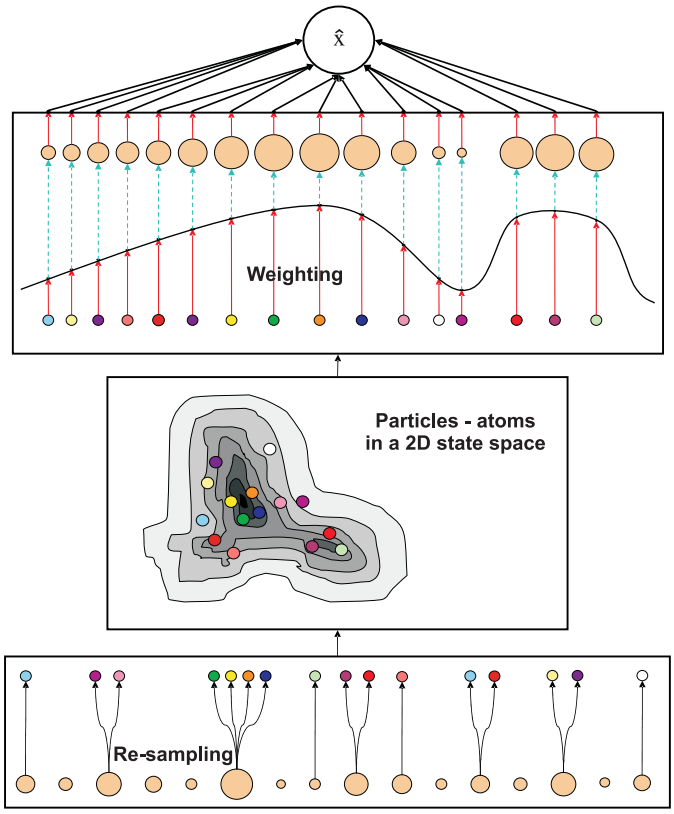

Fig. 4. Schematic representation of the Particle Filter algorithm.

Both trackers are initialized with the same video atoms using MP as described in Sec. II-A. The edges are then tracked using a video MP approach in 3D-MP, while our proposed method tracks the video structures using PF as detailed in Sec. II-B. In Fig. 5 the tracking results using the two algorithms are compared. The first and third rows show the results obtained with the 3D-MP approach and the second and forth rows show the results for the proposed MP-PF method. In the second part of the sequence (second and third frames) the subjects rapidly move towards the left. The 3D-MP tracker looses the track of two edges in the first case and of one in the second, while the MP-PF tracker does not. The same behavior has been observed in the other test sequences. While the 3D-MP algorithm easily loose the track of fast moving edges, the MP-PF approach results more robust, even if errors can be observed. In both sequences for example it happens that the yellow atom associated with the upper lip is temporarily associated with the lower lip or the chin.

In the next section the proposed tracking method is integrated in the audiovisual fusion algorithm presented in [11] to perform a cross-modal source localization task.

B. Audiovisual Source Localization

The analysis of audiovisual signals has received an increased interest in the last years. Each signal typically brings some information about the others and their simultaneous processing can uncover relationships that are otherwise unavailable when considering the sources separately. In their pioneering work, Hershey and Movellan [22] design a simple algorithm to locate sounds using audio-video synchrony. The correlation between audio and video was measured using the correlation coefficient between the energy of an audio track and the value of single pixels. Successive studies in the field [23–27] focused on the statistical modeling of relationships between audio and video features, proposing audiovisual fusion strategies based on Canonical Correlation Analysis [23, 27], Independent Subspace Projections [25] and Mutual Information maximization [24, 26].

While research efforts appear to be concentrated in the development of audiovisual fusion strategies, it seems that the features employed to represent the different signals are often basic and barely connected with the physics

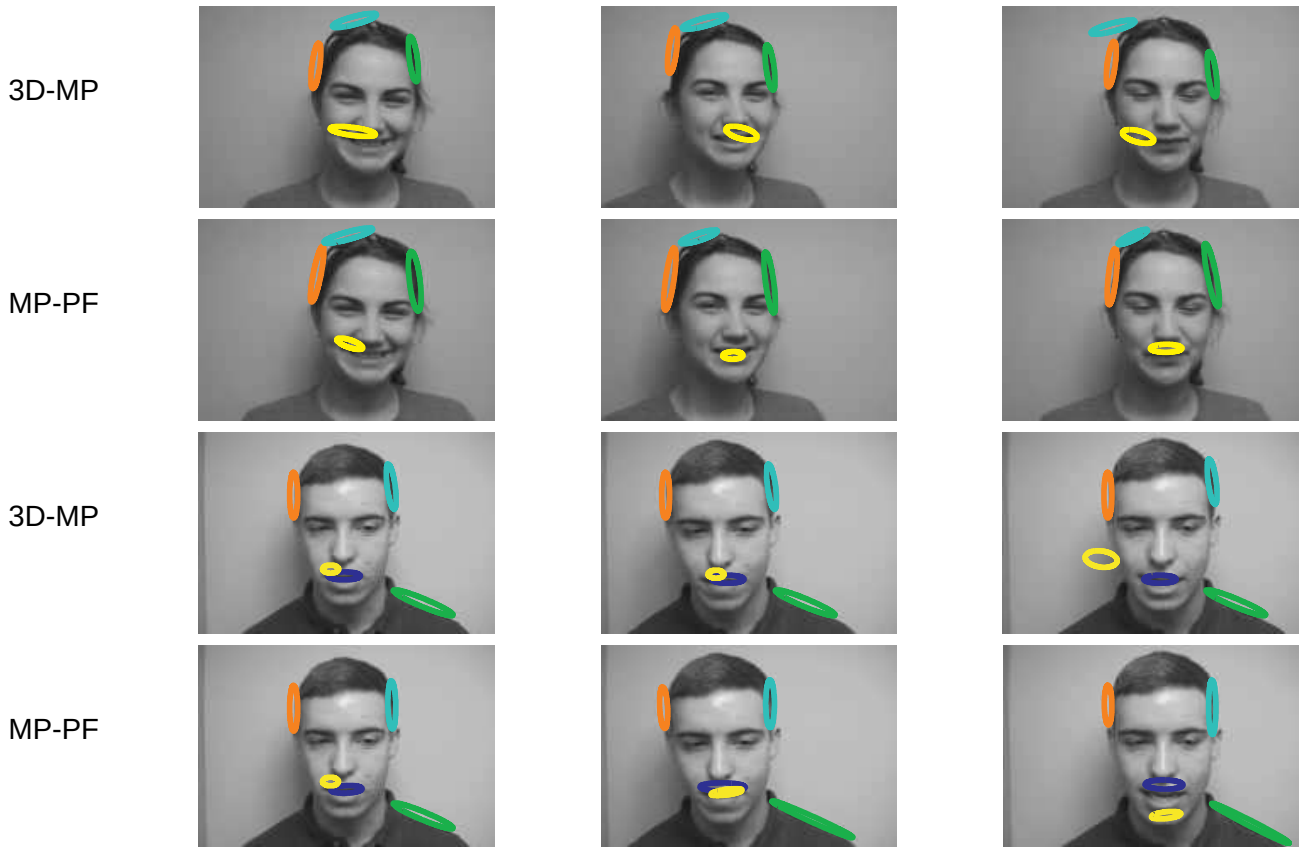

Fig. 5. Video atoms tracking. The footprints of different atoms are depicted with different colors. Results for the 3D-MP approach are on the first and third rows and those for the MP-PF method are on the second and forth rows. From the second to the third frame the subjects rapidly move towards their left : the 3D-MP tracker looses the track of some edges, while the MP-PF tracker does not.

of the observed phenomena (e.g. video sequences are typically represented using time series of pixel intensities). A representation of video signals based on visual geometric features tracked through time has been proposed in [11, 12, 28]. In these works, the trajectories of video structures have been successfully used to correlate audio-video data and localize the sound source in the video exploiting cross-modal correlation.

In the second experiment, MP-PF is integrated in the audiovisual fusion algorithm [11] to perform a source localization task. The audio-video features that are considered here are the same used in [11, 12]. The audio signal is represented by a mono-dimensional feature that estimates the average acoustic energy. The video signal instead is represented using $M = 30$ video atoms and each atom has a feature associated describing its displacement. Peaks are extracted from audio and video features and *synchronization vectors* are built [11]. The video atoms exhibiting the highest degree of correlation with the audio are detected using a simple relevance criterion and the sound source location over the image sequence is estimated. A sliding window of 70 frames length is used to compute the synchronization vectors and to detect the video atoms that are more correlated with the audio. The observation window is then shifted by 20 samples and the procedure iterated.

We have tested the algorithm on four sequences of the CUAVE database (g19, g20, g21, g22) involving two persons taking turn in reading series of digits in English. Figure 6 shows the results of the described approach detecting the mouth of the speaker in two sequences where two persons speak in turns in front of the camera. In white are highlighted the footprints of the video atoms found to be correlated with the soundtrack. The mouth of the correct speaker is detected.

In order to quantify the accuracy of the proposed method the center of the speaker's mouth in the test sequences has been manually labelled. The active speaker's mouth is considered to be correctly detected if the position of the most correlated video atom falls within a circle of diameter D centered in the labelled mouth center. If several atoms are chosen, an atoms' centroid is estimated whose position on the image plane is given by the average of the single atoms coordinates. Correlated atoms are detected every 20 frames, thus mouth labels are placed with

Fig. 6. Frames from clips g19 [Top] and g21 [Bottom]. The footprints of the most correlated atoms are highlighted. The mouths of the correct speakers are detected.

Clip	Nock[24]*	Monaci[11]	**Proposed**
g19	41	87	**94**
g20	93	93	**93**
g21	79	81	**78**
g22	79	87	**80**

TABLE I

RESULTS EXPRESSED IN PERCENTAGE OF CORRECT DETECTIONS. *VALUES SHOULD BE CONSIDERED AS INDICATIVE (SEE TEXT).

this same frequency throughout each sequence, and performances are evaluated at test points distant 20 samples one from the other. The value of the diameter D is set to 50 pixels. This value has been chosen so that we can compare the results with those presented in [24] and [11].

Nock and colleagues [24] propose a method to detect the mouth of the speaker founding the image zone over which the mutual information between audio and video features is maximized. As in our algorithm, in [24] mutual information values are estimated using a sliding time window of 60 frames that is shifted in time with steps of 30 frames. The goodness of the detection is assessed using the criterion that we use here, with the only difference that in [24] the speaker's mouth is considered to be correctly located if it is placed within a *square* of 200×200 pixels centered on the manually labelled mouth center. Thus, taking into account a downsampling factor of 4 that we have applied to the video sequences, the areas of correct mouth detection are comparable. However, we must note that the test clips used in [24] could not exactly coincide with those used in this paper, since the original sequences have been cropped in both cases. In contrast, the results presented in [11] are obtained using exactly the same test sequences. The main differences between the algorithm presented here and the one in [11] basically consist in the video edge tracking approach (here we use MP-PF, while in [11] the 3D-MP approach is used) and in the different number of atoms considered. We consider 30 atoms here and not 40 as in [11] because, as already underlined in the previous section, we track the atoms independently : the higher the number of atoms, the stronger are their interactions, as exemplified by Fig. 2. In [11] interactions between atoms are taken into account and thus this aspect is not an issue.

Table I summarizes the results obtained for the three methods in term of percentage of test points at which the speaker's mouth is correctly detected. Note that there could be no perfect coincidence between the test sequences used in [24] and those used in [11] and here, thus the results for Nock's algorithm should be considered only as indicative. As already shown in [11], the use of geometric video decompositions combined with an audio-video event detector in general improves the results obtained by Nock and colleagues. The proposed method obtains detection performances similar to those of Monaci's algorithm, slightly improving previous results for sequence g19 but obtaining inferior performances on clip g22.

The MP-PF method improves the tracking performances of the 3D-MP tracking algorithm, as shown by the results in Fig. 5. This is indeed interesting considering that the 3D-MP algorithm, even without jointly tracking

groups of structures, takes into account atoms' interactions, which was demonstrated to increase the accuracy of the 3D-MP approach [20]. We argue that a MP-PF algorithm that takes into account atoms' dependencies would correct tracking errors due to atoms' interactions (Fig. 5) and would allow to improve the audiovisual localization results, that by now are essentially equivalent to those obtained using 3D-MP (Table I). Concerning the computational complexity, we have tested the two methods on a video sequence whose 30 principal video atoms were tracked through time. The MP-PF algorithm clearly outperforms the 3D-MP approach, resulting approximately 7 times faster.

IV. Discussion

We presented a new framework and an efficient algorithm to represent and track relevant video structures. The proposed method improves the 3D-MP video representation algorithm presented in [8], which is designed as a coding algorithm and poses problems from the tracking point of view. The parameters of the video atoms are in fact coarsely quantized to achieve better compression performances, introducing tracking errors. Moreover, atoms are tracked using a search window of reduced size, which limits the robustness and accuracy of the tracker. These limitations are overcame by defining the video atom tracking problem in the well grounded and understood framework of PF, which ensures robustness, flexibility and lower computational complexity than the 3D-MP algorithm.

Experiments show that the proposed tracker is more robust and accurate than the 3D-MP one, while being considerably less time consuming. The audiovisual source localization algorithm, however, does not improve accordingly. This is mainly due to the fact that while in [11] the 3D-MP algorithm takes into account atoms' interactions, the current MP-PF method does not. This in certain situations produces less stable atoms trajectories because of interferences between atoms, as shown in Fig. 5. However these results show that there is room for further improvements by designing a mechanism that accounts for the interactions between video atoms. The tracking framework developed in this paper seems to be appropriate to continue the evolution of our system.

References

[1] Y.-S. Yao and R. Chellappa;, "Tracking a dynamic set of feature points," *IEEE Trans. on Image Proc.*, vol. 4, no. 10, pp. 1382–1395, 1995.

[2] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

[3] G. Edwards, C. Taylor, and T. Cootes, "Interpreting face images using active appearance models," in *Proc. of the 3rd Int. Conf. on Automatic Face and Gesture Recognition*, 1998, pp. 300–305.

[4] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of IJCAI*, 1981, pp. 674–679.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[6] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

[7] Y. Wang, J. Ostermann, and Y.-Q. Zhang, *Digital Video Processing and Communications*, Prentice Hall, 2001.

[8] Ò. Divorra Escoda, *Toward Sparse and Geometry Adapted Video Approximations*, Ph.D. thesis, EPFL, Lausanne, June 2005, [Online] Available: `http://lts2www.epfl.ch/`.

[9] S. Mallat and Z. Zhang, "Matching Pursuits with time-frequency dictionaries," *IEEE Trans. on Signal Proc.*, vol. 41, no. 12, pp. 3397–3415, 1993.

[10] O. Divorra Escoda, L. Granai, and Vandergheynst P., "On the use of a priori information for sparse signal approximations," *IEEE Trans. on Signal Proc., in press*, 2006.

[11] G. Monaci and P. Vandergheynst, "Audiovisual gestalts," in *Proc. of CVPR Workshop on Perceptual Organization in Computer Vision*, June 2006.

[12] G. Monaci, Ò. Divorra Escoda, and P. Vandergheynst, "Analysis of multimodal sequences using geometric video representations," *Signal Processing, in press*, 2006, [Online] Available: `http://lts2www.epfl.ch/`.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Trans. on Signal Proc.*, vol. 50, no. 2, pp. 174–188, 2002.

[14] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "A color-based particle filter," in *Proc. of the 1st Workshop on Generative-Model-Based Vision*, June 2002, pp. 53–60.

[15] S. Zhou, R. Chellappa, and B. Moghaddam, "Appearance tracking using adaptive models in a particle filter," in *Proc. of Asian Conf. on Computer Vision*, June 2002.

[16] I. Daubechies, "Time-frequency localization operators: A geometric phase space approach," *IEEE Trans. on Information Theory*, vol. 34, no. 4, pp. 605–612, 1988.

[17] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal of Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[18] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. on Communications*, vol. 31, no. 4, pp. 532–540, 1983.
[19] L. Peotta, L. Granai, and P. Vandergheynst, "Very low bit rate image coding using redundant dictionaries," in *Proc. of the SPIE, Wavelets: Applications in Signal and Image Processing X*, November 2003, vol. 5207, pp. 228–239.
[20] Ò. Divorra Escoda and P. Vandergheynst, "A Bayesian approach to video expansions on parametric over-complete 2-D dictionaries," in *Proc. of IEEE MMSP*, September 2004, pp. 490–493.
[21] 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.
[22] J. Hershey and J. Movellan, "Audio-vision: Using audio-visual synchrony to locate sounds," in *Proc. of NIPS*, 1999, vol. 12.
[23] M. Slaney and M. Covell, "FaceSync: A linear operator for measuring synchronization of video facial images and audio tracks," in *Proc. of NIPS*, 2000, vol. 13.
[24] H. J. Nock, G. Iyengar, and C. Neti, "Speaker localisation using audio-visual synchrony: an empirical study," in *Proc. of the 10^{th} ACM International Conference on Multimedia*, 2002.
[25] P. Smaragdis and M. Casey, "Audio/visual independent components," in *Proc. of ICA*, 2003, pp. 709–714.
[26] J. W. Fisher III and T. Darrell, "Speaker association with signal-level audiovisual fusion," *IEEE Trans. on Multimedia*, vol. 6, no. 3, pp. 406–413, June 2004.
[27] E. Kidron, Y. Schechner, and M. Elad, "Pixels that sound," in *Proc. of CVPR*, 2005, pp. 88–95.
[28] G. Monaci, Ò. Divorra Escoda, and P. Vandergheynst, "Analysis of multimodal signals using redundant representations," in *Proc. of IEEE ICIP*, September 2005.