# Chapter 5

# A Polynomial Fiber Description of Motion for Video Mosaicing

This chapter is about two topics. First, given a video sequence, we introduce a Bayesian framework, where background and moving objects are segmented into layers. The model that describes the different layer evolutions in a sequence of images uses the results of a multi-frame optical flow estimation (MFOFE); we present a new technique based on the fact that each pixel in the frame of reference produces a trajectory in the mosaic absolute coordinate system. The second topic is video summarization. A general moisaicing method is presented for describing the background and the trajectories of moving objects in a sequence of frames. Combining layer segmentation and mosaicing, we show different manners of encoding and visualizing temporal information, where the key point is the selection of a certain object in the images as reference in the evolution.

## 5.1 Introduction

Summarizing the contents of a video into a single image is a very useful tool for video indexing and compression purposes [46, 102]. Indeed, browsing and retrieval by content in video data-bases are becoming a relevant field in Computer Vision and Multimedia computing. This fact goes in accordance with the increasing developments in digital storage and data transmission. In addition to this, the wide range of applications in this framework, such as advertising, publishing, news and video clips, points out the necessity for more efficient organizing techniques. In this chapter, we focus on two important subjects in this area, movement segmentation and video mosaicing as a summarization technique. These subjects make feasible a quick intuition of the evolution of higher level perceptual structures, such as scenes, short stories and panoramic view sequences [98, 93]. Algorithms for image mosaicing con-

sist of two main steps: registration, i.e. estimating the transformation that occurred across consecutive frames in the sequence, and mosaic construction, which implies utilizing the previously estimated transformations in conjunction with the images to be summarized. These two steps are intrinsically related. A good performance of the resulting final mosaic is strongly dependent on the variety of techniques which are applied in both steps.

The first step, *transformation estimation*, is usually based on optical flow estimation in pairs of consecutive frames. However, when images are put in correspondence, rather than an estimation of the relative transformation between consecutive images, a estimation of the absolute transformation between each frame and a selected frame of reference is more necessary. This goes in accordance with the fact that a mosaic coordinate system has to be selected in order to locate the images of a sequence in a mosaic structure. For this reason, in this chapter, we apply the Multi Frame Optical Flow estimation (MFOFE) introduced in [45], since it provides a global estimation of images transformation respect to a previously selected frame of reference. Moreover, this technique offers a good solution to the aperture problem without assuming an a-priori restricted model of the world or of the camera motion.

The second step, *mosaic's structure construction*, utilizes the world coordinate transformations to put images in correspondence, i.e., to estimate the overlapped regions that are common in the images of a sequence. The presence of moving objects in a scene gives rise to a more complex situation. In this case, background and object motions must be separated in order to obtain an accurate registration. To this end, we present a new technique where each pixel in the frame of reference produces a trajectory in the mosaic absolute coordinate system. Unlike techniques which are restricted to pairs of frames [105, 6, 88], this method performs simultaneously a global motion segmentation across all the frames. Clustering is based on the fact that similar trajectories will correspond to the same sort of motion (and camera operation). Thus, we introduce a description of these paths in terms of polynomial fibers, and a probabilistic model is developed in order to rely on a measure of similarity as well as to have a classification mechanism which extracts the possible different classes of motions.

The outline of this chapter is as follows: first, we introduce a description of the features (fibers) that are used to the analysis of the relative movements among different objects. Subsequently, a probablisitic model and an EM algorithm are presented in order to establish the settings for classification. Section 3 shows the development used for mosaic's structure construction and the application of the information of the fibers model. Finally, the chapter is concluded with the summary and conclusions.

## 5.2 Motion in terms of Fiber-Like Structures

The MFOFE, that is presented in [45], gives a technique to describe the absolute transformations of each pixel in a certain frame of reference across a sequence of images. In this section, this estimation is used to detect different relative movements

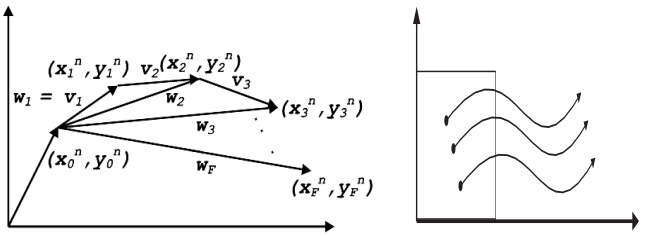



**Figure 5.1:** (a) Description of a fibre in terms of absolute coordinates; it starts in $(x_0^n, y_0^n)$, and, by successive applications of the corresponding absolute velocities $\vec{\omega}_f$, the points in the fiber $(x_f^n, y_f^n)$ are obtained. (b) Fiber picture in the mosaic coordinates.

employing all the information from the sequence.

### 5.2.1 Settings

The starting point is to define the features that are used to the analysis of relative movements among different objects in a sequence of $F$ images $\{I_0, \ldots, I_F\}$, where each image $I_f$ consists of $N$ pixels. Let $I_0$ be the absolute frame of reference and $(x_0^n, y_0^n)$ the *n-th* pixel in this frame, thus, $(x_f^n, y_f^n)$ is the resulting pixel in the $f$*th*-frame of applying the absolute velocity vector $\vec{\omega}_f$ to $(x_0^n, y_0^n)$. MFOFE allows to estimate these absolute coordinates that correspond to this pixel in the following frames. Following the scheme in fig. 5.1, we define a *fiber* $S(n)$ for each pixel $(x_0^n, y_0^n)$ in $I_0$ as:

$$S(n) = [(x_0^n, y_0^n), \ldots, (x_F^n, y_F^n); \vec{v}_1, \ldots, \vec{v}_F]$$

where each $\vec{v}_i$ is a relative velocity vector obtained through the difference of the absolute velocities which are provided by MFOFE, i.e., $\vec{v}_f = \vec{\omega}_f - \vec{\omega}_{f-1}$. In this way, rather than analyzing point pixel structures, it turns out to be more robust the analysis of fibers which are associated to each pixel in the selected frame of reference.

### 5.2.2 A Polynomial Surface Model

We consider the whole sequence of images as a fiber bundle. Each fiber can be described in terms of a polynomial model as follows: let $S(n)$ be the fiber associated to the *n-th* pixel in $I_0$, therefore, the components of the velocity vectors can be fitted to a polynomial of degree[1] $d$:

$$\begin{aligned} u_f^n &= a_{00} + a_{10}x_f^n + a_{01}y_f^n + a_{ij}(x_f^n)^i(y_f^n)^j + \ldots + a_{0d}(y_f^n)^d \\ v_f^n &= b_{00} + b_{10}x_f^n + b_{01}y_f^n + a_{ij}(x_f^n)^i(y_f^n)^j + \ldots + b_{0d}(y_f^n)^d \end{aligned}$$

[1]When $d = 0$, the polynomial corresponds to a translation, $d = 1$ to an affine model and $d = 2$ to a projective camera model of a moving plane of the fiber $S(n)$.

where $\vec{v}_f^n = (u_f^n, v_f^n)$ (relative velocity) is analyzed in terms of its components, and the coordinates in the $f$-$th$ frame are represented by $(x_f^n, y_f^n)$ [2]. The number of unknown coefficients in each polynomial is $r = \frac{(d+1)(d+2)}{2}$. Besides, the previous forms can be written in terms of an inner product of the following vectors:

$$\begin{aligned} \vec{p}_f^n &= \left(1, x_f^n, y_f^n, \ldots, (x_f^n)^i (y_f^n)^j, \ldots, (x_f^n)^d, (y_f^n)^d\right)^T \\ \vec{\alpha} &= (a_{00}, a_{10}, a_{01}, \ldots, a_{d0}, a_{0d})^T \\ \vec{\beta} &= (b_{00}, b_{10}, b_{01}, \ldots, b_{d0}, b_{0d})^T \end{aligned}$$

hence, the velocities fitting can be re-written as follows:

$$u_f^n = [\vec{p}_f^n]^T \vec{\alpha} \tag{5.1}$$

$$v_f^n = [\vec{p}_f^n]^T \vec{\beta} \tag{5.2}$$

Let $U^n$ and $V^n$ be the column vector $F \times 1$ of the velocity components in the fiber $S(n)$, and $P^n$ a matrix $r \times F$ of its corresponding point [absolute] coordinates, therefore, eq. (5.1) and eq. (5.2) are extended in a matrix form:

$$U^n = [P^n]^T \vec{\alpha} \tag{5.3}$$

$$V^n = [P^n]^T \vec{\beta} \tag{5.4}$$

where $\vec{\alpha}$ and $\vec{\beta}$ are $r \times 1$ vectors. In the following section, we introduce a probabilistic formulation to estimate a mixture of surfaces that describe the behavior of the different sort of movements which are present at the video sequence.

### 5.2.3 Probabilistic Mixture Model

Describing a fiber $S(n)$ in terms of a set of coefficients $\Omega = \{\vec{\alpha}, \vec{\beta}\}$ gives us a starting point to develop a probabilistic formulation. The idea behind this, is to provide a model that permits a classification of different types of fibers. These classes are related with the sort of diferent movements that are produced in the sequence. Consider eqs. (5.3) and (5.4) as the generative functions of the velocities along a fiber. Under the assumption of independent zero-mean Gaussian distributed noise in the velocity components, the likelihood function of a fiber, for a given instance of the model $\Omega$, is[3]:

$$P(U^n, V^n \mid P^n, \Omega) = P(U^n \mid P^n, \Omega) P(V^n \mid P^n, \Omega) \tag{5.5}$$

where,

$$P(U^n \mid P^n, \Omega) = \frac{1}{\sqrt{2\pi\sigma_u^2}} \exp\left[-\frac{1}{2\sigma_u^2} \mid U^n - [P^n]^T \vec{\alpha} \mid^2\right]$$

$$P(V^n \mid P^n, \Omega) = \frac{1}{\sqrt{2\pi\sigma_v^2}} \exp\left[-\frac{1}{2\sigma_v^2} \mid V^n - [P^n]^T \vec{\beta} \mid^2\right]$$

[2] Recall superscript $n$ is related to $(x_0^n, y_0^n)$ in the frame of reference

[3] This model assumes independence between the velocity components. Due to this assumption, eq. (5.5) can be factorized. However, the formulation of this model permits a straight forward inclusion of correlation between the velocity components. This means a larger number of unknown parameters to be estimated $\sim \mathcal{O}(F^2/2)$. The examples shown in this chapter work well under this assumption.

When different types of movement are present in an image sequence, their fiber representation leads to take into account more than one model. Consider that a fiber can be explained by a set of $Q$ models $\mathcal{M} = \{\Omega_1, \ldots, \Omega_Q\}$. In this case, the likelihood function (5.5) corresponds to a probability that is conditioned to a certain sub-model $\Omega_i$, $P(U^n, V^n \mid P^n, \Omega_i)$. Therefore, the global likelihood function is:

$$P(U^n, V^n \mid P^n, \mathcal{M}) = \sum_{i=1}^{Q} P(\Omega_i) P(U^n, V^n \mid P^n, \Omega_i) \tag{5.6}$$

where $P(\Omega_i)$ is a prior distribution over the sub-model $\Omega_i$. Given a instance of these sub-models, the classification of each fiber in a sequence of frames is a matter of Maximum a Posteriori (MAP), $\Omega_i = \text{argmax}_{i'} P(\Omega_{i'} \mid U^n, V^n, P^n)$ which is given by the Bayes rule:

$$P(\Omega_i \mid U^n, V^n, P^n) = \frac{P(\Omega_i) P(U^n, V^n \mid P^n, \Omega_i)}{\sum_{i'=1}^{Q} P(\Omega_{i'}) P(U^n, V^n \mid P^n, \Omega_{i'})} \tag{5.7}$$

### 5.2.4 EM Algorithm

The Expectation-Maximization approach is applied to this problem in order to estimate the models that explain the different fiber categories. This is based on the maximization of a likelihood function of the set of fibers that are obtained from the image sequence. Considering the approximation of independent observations among fibers, the global likelihood function of a set of fibers $S(1), \ldots, S(N)$ (N is the number of pixels in the frame of reference $I_0$) is written as a product of single likelihoods. Equivalently, this maximization can be done through maximizing the log-likelihood:

$$\mathcal{L} = \sum_{n=1}^{N} \log \left\{ \sum_{i=1}^{Q} P(\Omega_i) P(U^n, V^n \mid P^n, \Omega_i) \right\} \tag{5.8}$$

When assuming that each fiber is explained by only one sub-model $\Omega_i$, eq. (7.4) can be written in terms of binary variables $z_{ni}$:

$$\mathcal{L} = \sum_{n=1}^{N} \sum_{i=1}^{Q} z_{ni} \log \{P(\Omega_i) P(U^n, V^n \mid P^n, \Omega_i)\} \tag{5.9}$$

The EM algorithm consist of a two-step iterative procedure that converges to a local maximum of (7.4):

- E-step: set $R_{ni} = P(\Omega_i \mid U^n, V^n, P^n)$ and compute the posterior expectation of (5.9).
- M-step: for each sub-model $\Omega_i$
  1. $P(\Omega_i) = \sum_{n=1}^{N} R_{ni}/N$
  2. $\vec{\alpha}_i = \left[\sum_{n=1}^{N} R_{ni} P^n [P^n]^T\right]^{-1} \left[\sum_{n=1}^{N} R_{ni} P^n U^n\right]$

3. $\vec{\beta}_i = \left[\sum_{n=1}^{N} R_{ni} P^n [P^n]^T\right]^{-1} \left[\sum_{n=1}^{N} R_{ni} P^n V^n\right]$
4. $\sigma_{ui}^{-2} = \frac{1}{\sum_{n=1}^{N} R_{ni}} \sum_{n=1}^{N} R_{ni} \mid U^n - [P^n]^T \vec{\alpha}_i \mid^2$
5. $\sigma_{vi}^{-2} = \frac{1}{\sum_{n=1}^{N} R_{ni}} \sum_{n=1}^{N} R_{ni} \mid V^n - [P^n]^T \vec{\beta}_i \mid^2$

These steps are iterated until a certain convergence criteria.

## 5.3 From video to Mosaic Representations

The previous representation of motion in terms of fibers has two straight forward applications: 1) a motion clustering of the different regions in the reference image, which is extended to all the images of the sequence, and 2) permits an estimation of the global transformation that is produced in each segmented region. This second issue allows to a priori estimate the size of the mosaic, since those transformations are computed taking as reference a common frame for all the images. Thus, the translation contribution of global transformation of the "furthest"[4] frame will determine such a size.

Besides, in order to build the mosaic we have to select which layer is taken as reference as well. The selection of a layer as reference will completely determine the action that is summarized in the final mosaic. For instance, fig. 5.4 (a) shows a mosaic, where the reference layer is the segmented background, and fig. 5.4 (b) is given by choosing as reference the segmented foreground. The layer selection procedure is carried out thought using the maximum a posteriori probability in eq. (5.7), which is assigned to each fiber of the sequence. In this way, a sequence of labeled images is produced, where each label indicates the ownership of each fiber to a certain layer.

Once the reference layer is selected, all the images in the sequence are put in correspondence by means of the estimation of the global transformations of each image with respect to the reference frame. This can be performed easily by fitting the optical flow in each image (separately) using: either a 1-degree polynomial function (affine model), or 2-degree polynomial (projective model). Figure 5.3 show a picture of the mentioned data structure. When images are put in correspondence, each position in such a data structure indicates those pixel that will contribute to the formation of the final mosaic. At this point, given a instance of the data structure, the final mosaic can be presented in two manners. First, when the labeling, which was obtained using the fiber analysis, is utilized, the final mosaic will only take into account those pixels that belong to the selected layer. This result is shown figures 5.4 (c) and (d), where (c) is a mosaic that takes a reference layer the background, and (d) the mosaic that only considers the segmented foreground. In these two mosaic appear black areas that indicate that the layer that has been taken as reference had no information in these regions. In (c) the black region corresponds to a zone where the tree-layer was present in the whole sequence. The second option is to consider all the pixels in the images;

[4]The term far, here, is taken with respect to the frame of reference.

**Figure 5.2:** Three frames from a sequence of 25.

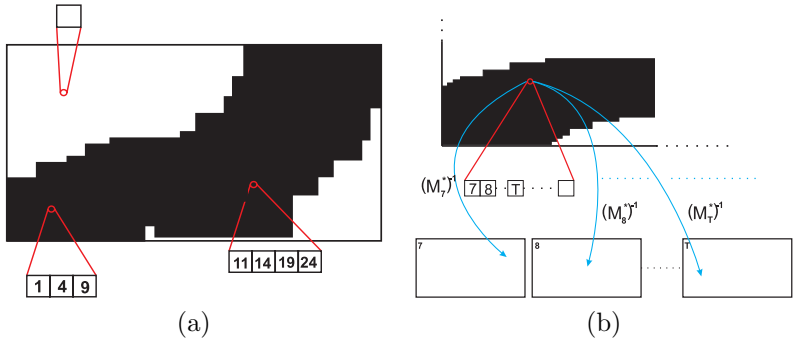


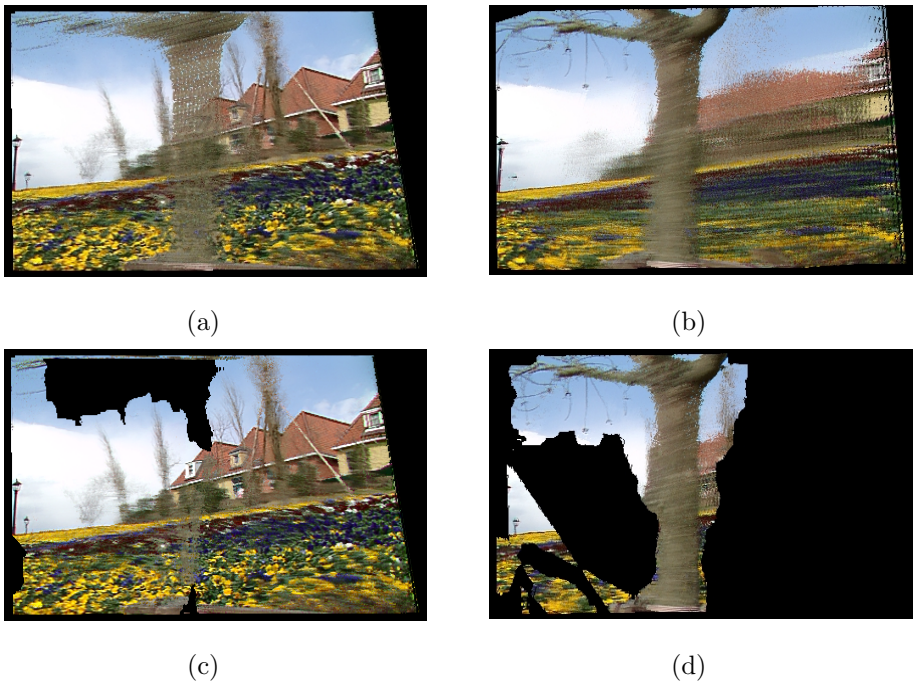

**Figure 5.3:** Scheme of the data structure.

this result is shown in fig. 5.4 (a) and (b). It is worth it to comment that the pixel that appears in the final mosaic (for each position in the data structure) is obtained is these examples by a median computation. Our purpose is to point out that once the introduced data structure is computed, the different resulting mosaics come from this common structure. The only thing that differs is whether a labeling mask is used or not, and the selected operation, such as median or average, is employed in order to determine which pixel from the frames, that contribute to a certain location in the data structure, is shown in that final mosaic.

In figure 5.2 we show three frames of the video sequence that has been used for these examples. Finally, in figure 5.5 we show a mosaic, where the reference layer is he background, which has three masks of the foreground in order to summarize the video sequence.

## 5.4 Conclusions and future work

In this chapter, we presented a Bayesian method for creating mosaics from video sequences with moving objects. Both prior and posterior information, through an analysis in terms of fibers, are exploited in order to distinguish the significantly different sort of relative movements along a whole sequence. In future work, we consider that the relation between polynomial coefficients and $3D$ recovery is an open interesting issue to be analyzed.



(a) (b)

(c) (d)

**Figure 5.4:** (a) Image mosaic with background as reference layer. (b) Image mosaic with foreground as reference layer. (c) Segmented background mosaic.(d) Segmented tree mosaic. Black color regions in (c) and (d) show those regions that the segmented layer had no information contributions to the final mosaic during the whole sequence of frames.

**Figure 5.5:** Summarization of the movement of the tree along the sequence.

# Chapter 6

# Video Summarization through Iconic Data Structures

Representing motion into a single image has been a challenge since the beginnings of Humanity (from cavern hunting paintings to modern art). Even children, when drawing their parents' car in motion, add some oriented blurring effects in order to represent time in a single picture. Such a form of compression, from a temporal sequence of images to a reduced set, is straightforwardly meaningful, since we are able to reconstruct an approximation of the original temporal sequence from our experience. In this chapter, we address the video summarization problem in a Bayesian framework in order to detect and describe the underlying temporal transformation symmetries in a video sequence. Given a set of time correlated frames, we attempt to extract a reduced number of image-like data structures which are semantically meaningful and that have the ability of representing the sequence evolution. To this end, we present a generative model which involves jointly the representation and the evolution of appearance. Applying Linear Dynamical System theory to this problem, we discuss how the temporal information is encoded yielding a manner of grouping the iconic representations of the video sequence in terms of invariance. The formulation of this problem is driven in terms of a probabilistic approach, which affords a measure of perceptual similarity taking both learned appearance and time evolution models into account. This measure provides a setting for assigning boundaries to sequence of frames.

## 6.1 Introduction

Organizing, browsing and retrieval by content in video data-bases is becoming a relevant field in Computer Vision and Multimedia Computing. This fact goes in accordance with the increasing developments in digital storage and transmission. In addition to this, the wide range of applications in this framework,

such as advertising, publishing, news and video clips, points out the necessity for more efficient organizing techniques [30, 76].

In this chapter, we focus on two important subjects in this area that are video preview and summarization, and, which make feasible a quick intuition of the evolution, under a low streaming cost, of higher-level perceptual structures, such as stories, scenes or pieces of news. That fact becomes relevant for low bandwidth communication systems. Expressing a video sequence in terms of a few representative images permits a continuous media to be *seekable*. Besides, the summarizing ability of a story will depend on the specific choice of *key-frames* set. Currently, the standard approach for keyframes selection, as indicators of the content of video, is to choose certain images that belong to the video sequence, which usually correspond to the beginning and the end of clips. However, considering that editors, authors and artists utilize camera operations to communicate some specific intentions, this standard key-frame selection may presents the risk of losing semantic information. Selecting solely one image of the sequence to represent its temporal evolution may lack of expressiveness in terms of summarization purposes. When more than one types of motions (due to different frequencies, velocities, etc) are involved in the sequence, more than one icon are necessary to represent it. For instance, figure 6.1 shows a sequence of images where two main types of motions are involved. On one hand, the arm is rising with a certain velocity, while, on the other, the head is turning with a different rapidity of movement.

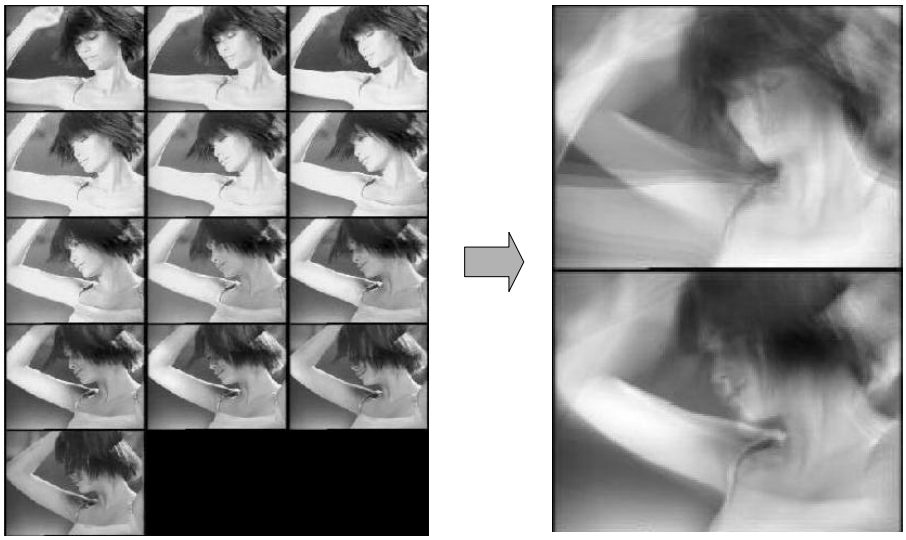


**Figure 6.1:** Semantic keyframes summarization of a sequence of images.

For this reason, our purpose is to present a compact and perceptually meaningful representation that preserves the subjective approach, i.e. the semantics, given by actions and camera operations in the evolution of a video sequence. The model to extract this new set of iconic representative image -like data structures (see fig. 6.1) is

based on an application of Linear Dynamical System and Lie's group theories, which are our support to define temporal symmetries and invariances. In this framework, the temporal information is encoded in an infinitesimal generator matrix, which defines different types of behaviors in the evolution of an image sequence. We use this distinct sort of contributions to give, in addition, a grouping inside the summarized representation.

The formulation of this problem is driven in terms of a probabilistic approach. Appearance representation and time evolution between consecutive frames are introduced in a generative model framework. First, a feature space is built through Probabilistic Principal Component Analysis (PPCA) [15], since this technique allows to codify images as points capturing the *intrinsic degrees of freedom* of the appearance, and at the same time, it yields compact description preserving semantics and perceptual similarities [106, 77, 68]. Subsequently, we present a generative dynamical model for the estimation of the curve's behavior that the sequence of images describe in this subspace of principal features. Authors in [82] introduced previously this dynamical model in a neural network framework. However, we embed it into a latent variable model, providing an EM algorithm for its estimation. This fact avoids undesirable problems such as when it comes to instantiate by hand the update steps of gradient descents techniques. Furthermore, the presented latent variable model allows a conjugation of both semantic and temporal representations. This affords a measure of perceptual similarity taking both learned appearance and time evolution sub models into account. Indeed, this probabilistic framework allows determining whether two consecutive images are in accordance with the learned dynamical model. This fact has an important significance when it comes to assign some boundaries to a sequence of frames.

The outline of this chapter is as follows: first, we introduce a review on Linear Dynamical Systems. The aim of this is to present the key points on the interpretation of the temporal appearance codification and how this information can be extracted. Subsequently, in section 3, an appearance probabilistic framework for time symmetry estimation is introduced in terms of latent variable models. Section 4 shows the experimental results in order to see this framework applied to real image problems. Section 5 presents the summary and conclusions. Finally, the appendix gives a detailed explanation of the developed EM algorithm for the dynamical model estimation.

## 6.2 On underlying symmetries

Consider a sequence of frames $\mathcal{F} = \{\phi_0, ..., \phi_N\}$ that are represented as vectors. Each vector corresponds to an image read in lexicographic order belonging to a subset of real numbers $\mathcal{S} \subset \Re^d$. Since images are obtained from a temporal sequence, order takes significant relevance where transition between two consecutive frames is achieved as a transformation from the previous one to the next one. Suppose that this transformation can be parameterised by a single real number $\theta \in \Re$, which gives us a notion of time over the whole sequence. Therefore, when $\theta$ tends to zero, the associated transformation is the identity, recovering the initial image $\phi_0$. In a first

approximation order, the relation between an image $\phi_0$ and a near one transformed $\phi(\delta\theta)$ can be expressed as: $\phi(\delta\theta) \simeq (1 + \delta\theta G)\phi_0$. So, a macroscopic transformation $T(\theta)$ can be built in terms of concatenating infinitesimal transformations, dividing the parameter $\theta$ in $M$ parts and making $M \to \infty$:

$$\phi(\theta) = \lim_{M\to\infty} \left(1 + \left(\frac{\theta}{M}\right) G\right)^M \phi_0 = e^{\theta G}\phi_0 \tag{6.1}$$

Equation (6.1) is related with to the study of a trajectory near a fixed point in $\mathcal{S}$ described by a linear dynamical system:

$$\dot{\phi} = G\phi \tag{6.2}$$

The basic idea considering a trajectory in $\Re^d$, formed by a sequence of images $\mathcal{F} = \{\phi_0, ..., \phi_N\}$, is to understand as a video sequence with some underlying appearance *invariance*. From a geometric the point of view invariance is defined as follows:

**Definition 6.2.1** *Let $\mathcal{S} \subset \Re^d$ be a set, then $\mathcal{S}$ is said to be invariant under the vector field $\dot{\phi} = T(\phi)$ if for any $\phi_0 \in \mathcal{S}$ we have $\phi(\theta, \phi_0) \in \mathcal{S}$ for all $\theta \in \Re$.*

Furthermore, we can see that the information available in the temporal evolution of a sequence of frames is encoded in the matrix $G$ under this linear model. The goal is to find how this information can be extracted. To this end, in the following section we describe the geometrical meaning of that matrix $G$, as well as, the behavior that follow the solutions of eq. (6.2) from the analysis of the internal structure of the linear system.

### 6.2.1 Geometrical Point of View of Dynamical Systems

In order to give an intuitive idea of the behavior of the solutions in eq. (6.2), we focus on an analysis of the orbit structure near fixed points. In eq. (6.1), a macroscopic transformation was built by considering a continuous process with incremental changes in the evolution parameter $\theta$. This type of transformations form a one-parameter Lie group, which satisfies the following differential equation:

$$\frac{dT(\theta)}{d\theta} = GT(\theta)$$

that corresponds to a generalization of the plane rotation and translation groups. The matrix $G$ is called infinitessimal generator or action of the group. Lie's group theory applied to Computer Vision is not new. In order to get an insight into this framework, we recommend [50], where a comprehensive view of its applications is developed.

Besides, the evolution described by (6.2) is a particular case of dynamical systems. Indeed, it corresponds to consider a linearization a system of differential equations:

$$\dot{\phi} = F(\phi) \to \dot{\phi} = DF(\phi)\mid_{\phi(0)} \phi \to \dot{\phi} = G\phi$$

where $G \equiv DF(\phi) \mid_{\phi(0)}$ with $\phi \in \Re^d$. This development is carried out through the analysis in the vicinity of a certain fixed point $\phi(0)$ at $\theta = 0$. In the following sections we show how this approximation can be assumed embedding the estimation problem into a probabilistic framework. This fact affords a measure of likelihood that determines whether a set of consecutive points $\{\phi(\theta_n)\}$ are as a result of a certain transformation of this type.

The Linear Dynamical Systems theory shows how to extract information of the system by means of an eigenvector analysis of the infinitesimal generator $G$. The starting point is that the $\Re^d$ space can be represented as a direct sum of three subspaces defined in terms of a set of (generalized) eigenvectors: $E^s = \text{span}\{e_1, \ldots, e_s\}$, $E^u = \text{span}\{e_{s+1}, \ldots, e_{s+u}\}$ and $E^c = \text{span}\{e_{s+u+1}, \ldots, e_{s+u+c}\}$. The first set of eigenvectors $\{e_1, \ldots, e_s\}$ corresponds to the eigenvalues of $G$ having negative real part, the second set are the eigenvectors $\{e_{s+1}, \ldots, e_{s+u}\}$ whose corresponding eigenvalues have positive real part, and $\{e_{s+u+1}, \ldots, e_{s+u+c}\}$ correspond to the eigenvalues of $G$ with zero real part. These subspaces are called, *stable* subspace $E^s$, *unstable* subspace $E^u$ and *center* subspace $E^c$ respectively, and $s + c + u = d$.

These spaces are an example of invariant subspaces, since solutions of eq. (6.2) with initial conditions entirely contained in either $E^s$, $E^u$ or $E^c$ must remain in that particular subspace for all values of $\theta$ (time) according to the definition 6.2.1.

In order to see the meaning of the eigenvalues of $G$, let us consider the following example. A curve in $R^4$ is built from two $2D$ quadratic forms, which, in this particular case, are an ellipse and a hyperbola (see fig. 6.2). This may be an example of generating a real space $\Re^4$ from the direct sum of lower dimensional subspaces. Now, we can see that the matrix $G$ that generated the orbit, with some initial condition, has two complex eigenvalues and two real ones. Indeed, under a similarity transformation $T$, $G$ can be written as follows:

$$G = T \begin{pmatrix} 0 & \omega & 0 & 0 \\ -\omega & 0 & 0 & 0 \\ 0 & 0 & \lambda_1 & 0 \\ 0 & 0 & 0 & \lambda_2 \end{pmatrix} T^{-1}$$

where $\{i\omega, -i\omega\}$ are pure imaginary eigenvalues of $G$, and $\{\lambda_1, \lambda_2\}$ are real values. In this particular case, we take $\lambda_1$ to be a real positive number, and $\lambda_2$ to be a negative real number. From this information, we can see that the $\Re^4$ space is decomposed in 3 subspaces, $E^c = [\text{span}\{e_1, e_2\}; \{i\omega, -i\omega\}]$, $E^u = [\text{span}\{e_3\}; \lambda_1]$ and $E^s = [\text{span}\{e_4\}; \lambda_2]$. Taking $\phi(0) = [x_1(0), x_2(0), x_3(0), x_4(0)]$ as inital condition, the parametric equation of the orbit, i.e. the solution of eq. (6.2) in this particular case is:

$$\begin{aligned} \phi(\theta) &= T e^{\Lambda\theta} T^{-1} \phi(0) = \\ &= T \begin{pmatrix} \cos\omega\theta & \sin\omega\theta & 0 & 0 \\ -\sin\omega\theta & \cos\omega\theta & 0 & 0 \\ 0 & 0 & e^{\lambda_1\theta} & 0 \\ 0 & 0 & 0 & e^{\lambda_2\theta} \end{pmatrix} T^{-1} \phi(0) \end{aligned} \tag{6.3}$$

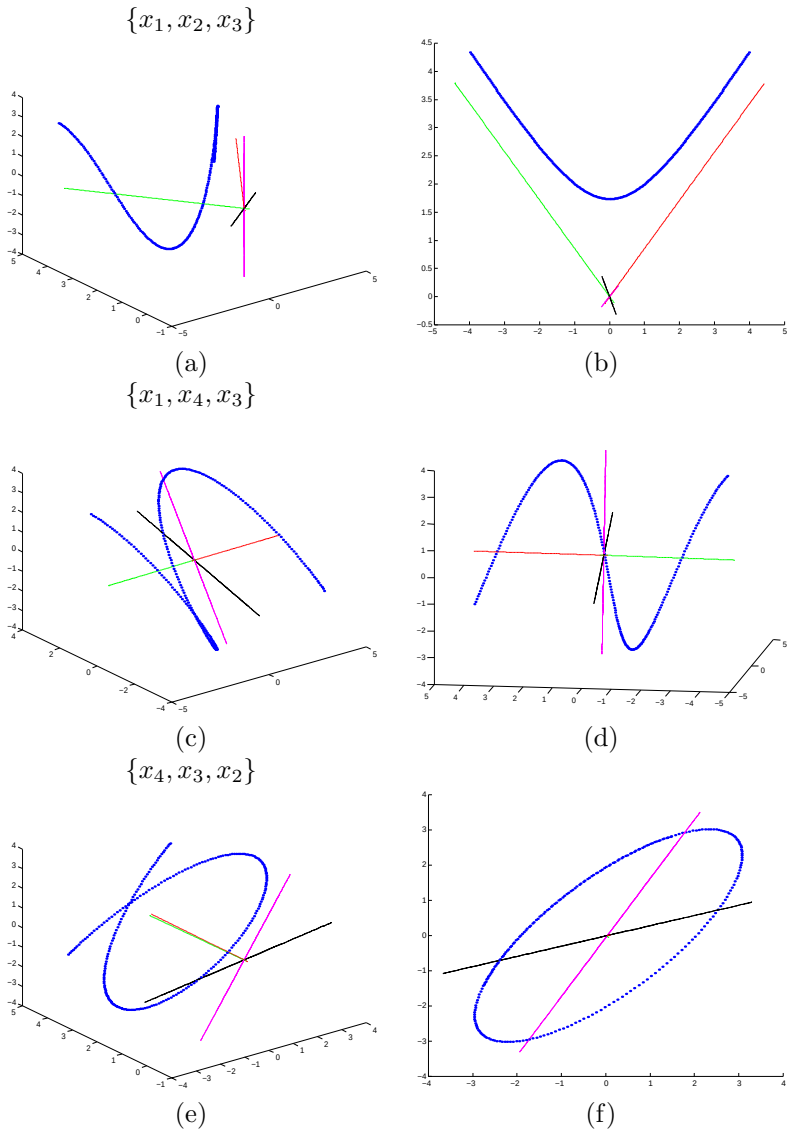



**Figure 6.2:** 3-Dimensional projections of a solution of a linear dynamical system in $4D$. Plot (b) is another view of (a), where the meaning of two (generalized) eigenvectors is interpreted as the asymptotes of $E^u$. Plot (f) shows the axis that span the subspace $E^c$, and it corresponds to the same $3D$ projection as (e).

From this solution, we deduce that $E^c = \text{span}\{e_1, e_2\}$ (purple and black axis directions in fig. 6.2 (f)) is an invariant subspace that generates a closed orbit, $E^u = \text{span}\{e_3\}$ (red axis direction in fig. 6.2(b))is an invariant subspace of solutions that decay to zero as $\theta \to -\infty$, and $E^s = \text{span}\{e_4\}$ (green axis direction in fig. 6.2(d))is the third invariant subspace of solutions that decay to zero as $\theta \to \infty$. For instance, consider an initial condition like the following:

$$T^{-1}\phi(0) = \begin{pmatrix} x_1(0) \\ x_2(0) \\ 0 \\ 0 \end{pmatrix}$$

Therefore, orbit obtained by means of (6.3) remains in $E^c = \text{span}\{e_1, e_2\}$ for all possible values of the time parameter $\theta$, and that fact is in accordance with the definition 6.2.1.

With this reference to the analysis of the solutions of linear dynamical systems, we see that the information, which is encoded in the infinitesimal generator $G$, is straight forward understandable through its eigenvalues and eigenvectors. This internal structure analysis not only allows the selection of a new representation for the images evolution, which is based in the modes of $G$ (eigenvectors), but also yields a manner of grouping the different principal directions of $G$ distinguishing the subspaces that they span in terms of stability, i.e., $E^u$,$E^s$ and $E^c$.

## 6.3 Appearance Based Framework for Time Symmetry Estimation

The previous example was performed in order to illustrate that a sequence of points that follow a certain temporal evolution can be described in terms of some privileged directions which are indicative of the behavior of the curve where they are embedded into. The aim of this is to apply linear dynamical system theory to temporal correlated sequences of images. To this end, we need to define a feature space where the images can be represented as points. Subsequently, the goal is to have an estimation of the evolution process of these images. The aim of this is to extract the temporal information encoded in the infinitesimal generator in order to present a reduced number of images that are able of summarizing semantically the whole sequence.

In this section, we present a generative model which defines appearance representation and time evolution between consecutive frames. This model involves jointly representation and evolution of appearance. In this case, temporal symmetry estimation is based on the fact that images belonging to a coherent sequence are also related by means of appearance representation.

First, the probabilistic formulation for appearance description is developed in terms of linear generative models through Probabilistic Principal Component Analysis (PPCA) [15]. Subsequently, the temporal appearance evolution is developed inside the linear generative model with the purpose of presenting a unified framework, where likelihood measure takes into account both appearance representation and evolution.

### 6.3.1 Appearance Representation Model

First of all, we need to define a space of features where images are represented as points. This problem involves to find a representation as a support for analyzing the temporal evolution. To address the problem of appearance representation, authors in [106, 77, 68] proposed Principal Component Analysis as redundancy reduction technique in order to preserve the semantics, i.e. perceptual similarities, during the codification process of the principal features. The idea is to find a small number of causes that in combination are able to reconstruct the appearance representation. These small numbers of causes are taken as the basis for the feature space. Besides, Tipping et. al. [15] embedded PCA into a Linear Generative Model framework in order to capture the intrinsic *degrees of freedom* of the object category model as well as to give an inherent *likelihood* measure to the learned object category. Generative models are a causal approach to describe the underlying phenomena that generates the complexity of observed data (images).

One of the most common approaches for explaining a data set is to assume that causes in linear combination:

$$t = Wx + \mu + e$$

where $x \in \Re^q$ (our chosen reduced representation, $q < d$) are the causes (latent variables), $W$ is an orthogonal matrix which rotates the data $t$, $\mu$ corresponds to the sample mean and $e$ is some noise. This causal approach leads to define a joint distribution $p(t, x)$ over visible $\{t\}$ and hidden variables $\{x\}$, the corresponding distribution $p(t)$ (similarity measure) for the observed data is obtained by marginalization: $p(t) = \int p(t \mid x)p(x)dx$, where $p(t \mid x)$ defines the *causal connection* between the observations $\{t\}$ and the latent variables $\{x\}$, and it is associated to the noise distribution as follows:

$$p(t \mid x) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left\{-\frac{1}{2\sigma^2} \mid t - \mu - Wx \mid^2\right\}$$

and the corresponding similarity measure given the model:

$$p(t) = \frac{1}{\sqrt{(2\pi)^d \det C}} \exp\left\{-\frac{1}{2}(t-\mu)^T \Sigma^{-1}(t-\mu)\right\}$$

where $\Sigma = WW^T + \sigma^2 I$. The *prior* knowledge on latent variables is expressed in $p(x)$. This density function takes the form of a Gaussian distribution with zero mean and identity covariance matrix: $\mathcal{N}(0, I)$. Therefore, it is said that the causes are mutually independent in terms of a second order statistics. The main goal is to find the parameters that maximize the joint observed data distribution i.e. the best description under the specific generative model. After considering the temporal model, the algorithm to estimate latent variables and parameters is introduced in a unified framework for appearance representation and evolution.

### 6.3.2 Appearance Temporal Evolution Model

Given a suitable basis to describe appearance, temporal symmetry can be analyzed in terms of this representation. An image $t$ corresponds to a point $x$ in the latent space,

$\mathcal{S}$. On the other hand, equation (6.1) can be interpreted in terms of a generative model, where an image description in latent space $x_{n+1}$ is obtained making to evolve with the action $G$, and a certain quantity $\theta_n$, a previous one $x_n$.
Symmetry learning is based on observations, more specifically, in a sequence of ordered images. So, is feasible to consider that observations are obtained with a certain additive noise. The generative equation takes the following form:

$$x(\theta) = e^{\theta G} x(0) + r \tag{6.4}$$

where $r$ is a $\theta$-independent noise process. According to the infinitessimal approximation $e^{\theta G} \sim 1 + \theta G$, eq. (6.4) yields:

$$\begin{aligned} x(\theta) &= (1 + \theta G)x(0) + r \\ \Delta x(\theta) &= \theta G x(0) + r \end{aligned} \tag{6.5}$$

where $\Delta x(\theta) \equiv x(\theta) - x(0)$. For the isotropic noise model case $r \sim \mathcal{N}(0, \beta^2 I)$, the probability distribution over the transformations $\Delta x$-space for a given image $x \in \mathcal{S}$ and step parameter $\theta$ corresponds to:

$$P(\Delta x \mid x, \theta, G, \beta^2) =$$
$$= \frac{1}{(2\pi\beta^2)^{q/2}} \exp\left\{ -\frac{1}{2\beta^2} \mid \Delta x - \theta G x \mid^2 \right\}$$

The prior distribution over the latent variables $\theta$ is assumed to be Gaussian with unit variance, so $\theta \sim \mathcal{N}(0, I)$. Therefore, the corresponding similarity measure for temporal transformations in latent space $\mathcal{S}$ is obtained by marginalization:

$$P(\Delta x \mid x, G, \beta^2) = \int P(\Delta x \mid x, \theta, G, \beta^2) P(\theta) d\theta =$$
$$= \frac{1}{(2\pi)^d \sqrt{\det C}} \exp\left\{ -\frac{1}{2} \Delta x^T C^{-1} \Delta x \right\} \tag{6.6}$$

where $C = G x x^T G^T + \beta^2 I$. The similarity measure eq. (6.6) evaluates the likelihood of a transformation $\Delta x$ between to points,(for a given $x_n$ to a following one $x_{n+1}$), respect to a learned model $\{G, \beta^2\}$. These points $\{x_n, x_{n+1}\}$ are a representation of two images $\{t_n, t_{n+1}\}$ for a certain instance of the appearance model $\{W, \mu, \sigma^2\}$. Indeed, this probabilistic framework allows determining whether two consecutive images are in accordance with the learned dynamical model. This fact has an important significance when it comes to assign some boundaries to a sequence of frames.

### 6.3.3 Maximum Likelihood Estimation

At this point, the problem is centered on parameter estimation, which, in practice, will be given by data observations. This leads to consider the problem of *incomplete data*. For this purpose, Dempster et al.(1977) [31] use the EM algorithm, where each observation $t_n$ (image) is associated to an unobserved state $s_n = \{x_n, \theta_n\}$, and the main goal is to determine which component generates the observation. In this sense, the unobserved states can be seen as *missing data* and therefore the union of observations $t_n$ and $s_n$ is said to be complete data, $y_n = (t_n, s_n)$. In this way, for a

given set of observations $\{t_1, \ldots, t_N\}$ the likelihood measure to be maximized is the *Complete-log-Likelihood*, i.e.:

$$\mathcal{L}(y_1, \ldots, y_N \mid \Omega) = \log \{p(t_1, \ldots, t_N; s_1, \ldots, s_N \mid \Omega)\} \tag{6.7}$$

where $\Omega$ represents the model parameters.
Although both parameters and latent variables are unobserved, the difference is that latent variables are pressumed to be instantiated once for every observation, that is there is a latent $s_n$ for each observation $t_n$. Furthermore, the noise model offers smoothness, then, this approach differs from regression-based methods, in the way that the goal is to estimate the data density, and leading to reduce the overfitting. The following table shows the paramaters $\Omega$ that are involved in the model, and the latent variables related to $s_n$:

| *Model* | *Generative Mapping* | *Parameters,* $\Omega$ | $s_n$ |
|---|---|---|---|
| App, Rep. | $t_n = Wx_n + \mu + e_n$ | $W, \sigma^2, \mu$ | $x_n$ |
| Time Sym, | $\Delta x_n = \theta_n G x_n + r_n$ | $G, \beta^2$ | $\theta_n$ |

Following the assumption that appearance representation depends only on data observations, the ML estimation for the appearance parameters is given in a closed form solution as it is developed in [15]:

$$W = U_q(\Lambda_q - \sigma^2 I)^{\frac{1}{2}} R; \quad \sigma^2 = \frac{1}{d-q} \sum_{j=q+1}^{d} \lambda_j$$

where $U_q$ are the first $q$ eigenvectors of the data set covariance matrix, $\Lambda_q$ is a diagonal matrix with the corresponding first $q$ eigenvalues ($\lambda_i, \quad \forall 1 \leq i \leq N$) and $R$ is an arbitrary rotation matrix.

In order to estimate the appearance dynamics we utilize an EM algorithm, which is detailed in the Appendix A. This is basically a two steps procedure: *Expectation* and *Maximization* of a likelihood function. The *Expectation* step requires a third operation, which in the latent variable model can be added to the pair of learning and model selection; *inference*. This refers to estimation of value of latent variables $s_n$ given known parameters $\Omega$ and observations $t_n$.

The introduced model shows a hierarchical structure between observed images $t_n$ and latent variables $x_n, \theta_n$. First, images are obtained to build the appearance representation, and secondly, taking avantage of a reduced appearance basis, data evolution is estimated. Inference is this framework is a simple matter of the application of Bayes' rule:

1. Inferring latent variables related to apperance:

$$p(x \mid t, \Omega) = \frac{p(x)p(t \mid x, \Omega)}{p(t \mid \Omega)} \tag{6.8}$$

2. Inferring latent variables related to temporal evolution, given appearance latent variables inferred and their corresponding transformations computed:

$$p(\theta \mid x, \Delta x, \Omega) = \frac{p(\theta)p(\Delta x \mid \theta, x, \Omega)}{p(\Delta x \mid x, \Omega)} \tag{6.9}$$

Therefore, for each image $t_n$ the computation of its corresponding coordinates in latent space $x_n$ is given by means of the *Maximum a Posteriori* (MAP) in eq.(6.8):

$$x = \text{argmax}_{x'} p(x' \mid t, \Omega) \tag{6.10}$$

Once, images are expressed in latent space coordinates, the computation of the best estimated transformation parameter $\theta_n$ is also done through the MAP in eq.(6.9):

$$\theta = \text{argmax}_{\theta'} p(\theta' \mid x, \Delta x, \Omega) \tag{6.11}$$

Under gaussian assumptions for noise models and prior knowledges, the *posterior* the posterior means $< x \mid t >$ and $< \theta \mid x, \Delta x >$ correspond to the MAP for each distribution. In the appendix we show the explicit forms for these posterior probabilities, as well as, an EM algorithm for the temporal parameters estimation is introduced.

## 6.4 Experimental Results

In this section, we exemplify the application of the introduced appearance evolution model to the extraction of semantically meaningful image-like data structures. Two types of experiments are developed in this section. First, a study of the invariance subspaces derived from the estimation of the action $G$ of the group is performed for periodic motions. Second, an application to summarization of video sequences is presented.

### 6.4.1 Capturing Local Behaviors through Global Representations

This first experiment has the purpose of studying the subspaces obtained from the eigenvector decomposition of the infinitesimal generator $G$. To this end, a synthetic sequences of images has been constructed. Figure 6.3 shows the temporal volume of this sequence of images. Two types of different periodic motions are present in the sequence. Their corresponding frequencies are significantly different.

**Global Representation.** The synthetic sequence consists of 100 images. Two white squares over a black background have oscillatory motions with two different frequencies. The aim of this experiment is to show how both movements can be locally segmented in the frame of reference thanks to the invariant subspace analysis. While a Principal Component Analysis of the images gives a set of appearance eigenvectors that describe de modes of variations, no segmentation

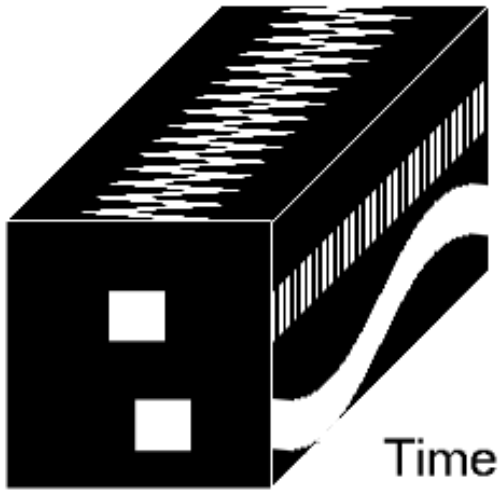


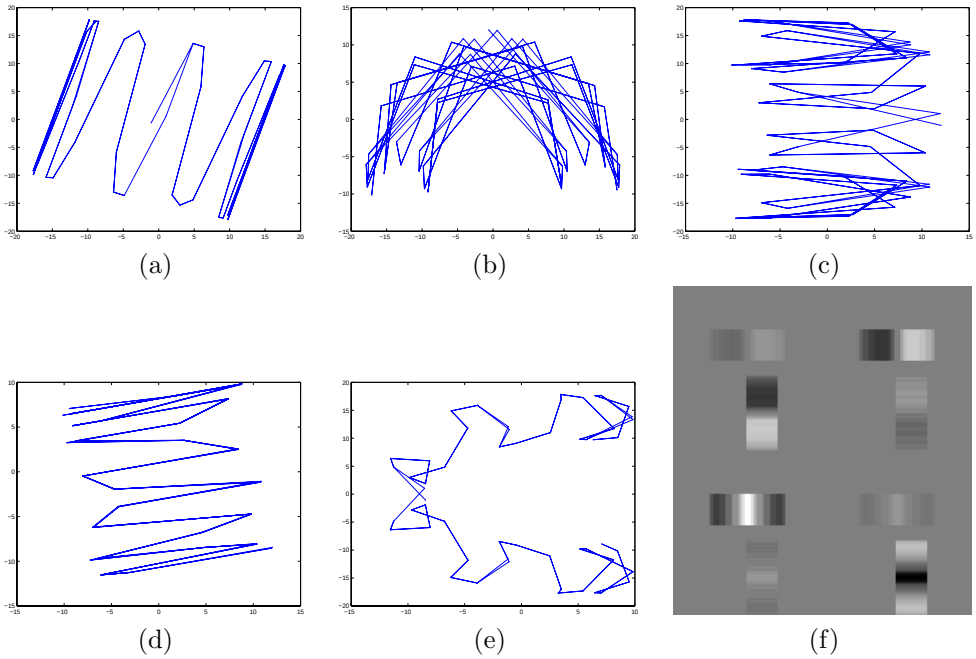

**Figure 6.3:** Temporal volume representation of the synthetic image sequence with two oscillatory motions.

between the two types of motions is performed (see figure 6.4 (f)). They are designed to rank variations in the set of images from low to higher frequencies. In fact, no temporal correlation is taken into account when constructing a PCA representation.

The PCA representation has been build using 4 Principal Components, that cope with the 70% of the total energy. Of course, more than 4 PC could have been selected. However, according to the previous analysis of the invariant subspaces, the aim is to divide the invariant orbits into two groups. Figure 6.4 shows the projected orbits onto pairs of Principal components. All of them, show that there is an inherent periodicity in the evolution of the trajectories.

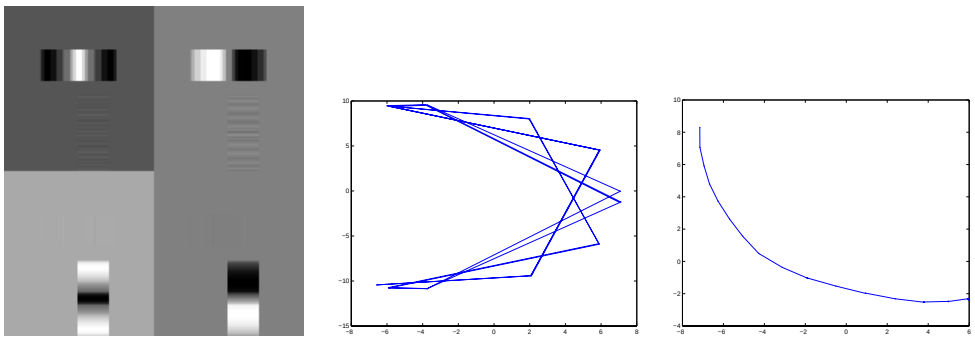
**Invariant Subspaces.** Adding temporal correlation to this analysis, we can see that two subspaces can describe the two different motions. This study has been performed by running the presented EM algorithm for estimating the infinitesimal generator $G$ of the temporal evolution, the corresponding one-dimensional parameter for each image. The eigenvector decomposition of the matrix $G$ gives two invariant subspaces. The corresponding orbits are plotted in figure 6.5. More specifically, figure 6.5 (b) shows many repetitions of the same trajectory, while figure 6.5 (c) shows only one trace. This is not surprising since if we look at the temporal volume in figure 6.3, we can notice that there is one horizontal motion with a higher frequency than the vertical one, which just presents one oscillation.

**Local Behavior Description.** The image representation (through back-projection)



(a) (b) (c) (d) (e) (f)

**Figure 6.4:** Plot (a) shows a projection onto the two first appearance eigenvectors, (b) onto $x_2, x_3$,(c) onto $x_3, x_1$,(d) onto $x_3, x_4$, and (e) onto $x_4, x_1$. Figure (f) shows the 4 eigenvectors obtained from PC Analysis.



(a) (b) (c)

**Figure 6.5:** Representation of the latent space coordinates in terms of the invariant subspaces obtained from diagonalizing the action of the group of transformations $G$. (a) shows the four corresponding back-projected eigendirections onto the image world. They are grouped according to invariance. Two invariant subspaces of dimension 2 each one describe the orbits in (b) and (c).

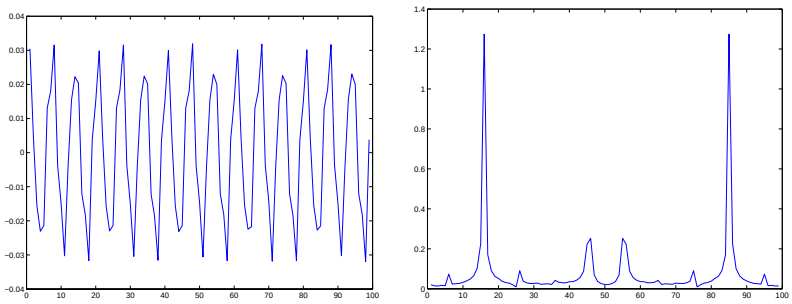



**Figure 6.6:** Representation of the temporal evolution of the one dimensional parameter $\theta$ in (a), and its Fourier Transform power spectrum in (b). Clearly, two frequencies are significantly noticeable.

of the the two sets of eigenvectors in $G$, leads to a meaningful description of the local behaviors of the two different motions in the frame of reference. In fact, they can be seen as masks that locally segment the regions where one of the two motions occurs. The image pair in the top of figure 6.5 (a) shows a segmentation of the area where the faster horizontal motion happens. Analogously, the area in the two images of the bottom of figure 6.5 (a) describe the slower vertical motion.

This local segmentation has been performed through a suitable global representation that takes into account both temporal evolution and invariance at the same time.

**Extracting Information from $\theta$.** The estimation of the one-dimensional parameter of the evolution for image also gives a notion of the two periodicities. Transforming the temporal evolution of $\theta(t)$ into the Fourier power spectrum representation, we can notice that two main frequencies are remarked. The accuracy when determining frequencies by means of this method, or through the imaginary parts of the eigenvalue analysis of the matrix $G$ depends on the reconstruction error of the selected appearance representation. Nevertheless, this technique can be used as an estimator of potential periodic motions in video sequences. This fact makes feasible an enrichment on automatic video annotation systems. Chapter 8 shows another technique to discriminate periodic motions based on a local representation of the pixel values variations.

### 6.4.2 Key-Frames Extraction

In this second block of experiments, we analyze two sequences: one focus on the evolution of an action (fig. 6.7(a)), and the other one corresponding to a camera operation, fig. 6.8(a). The estimation process is common for both cases, first it is

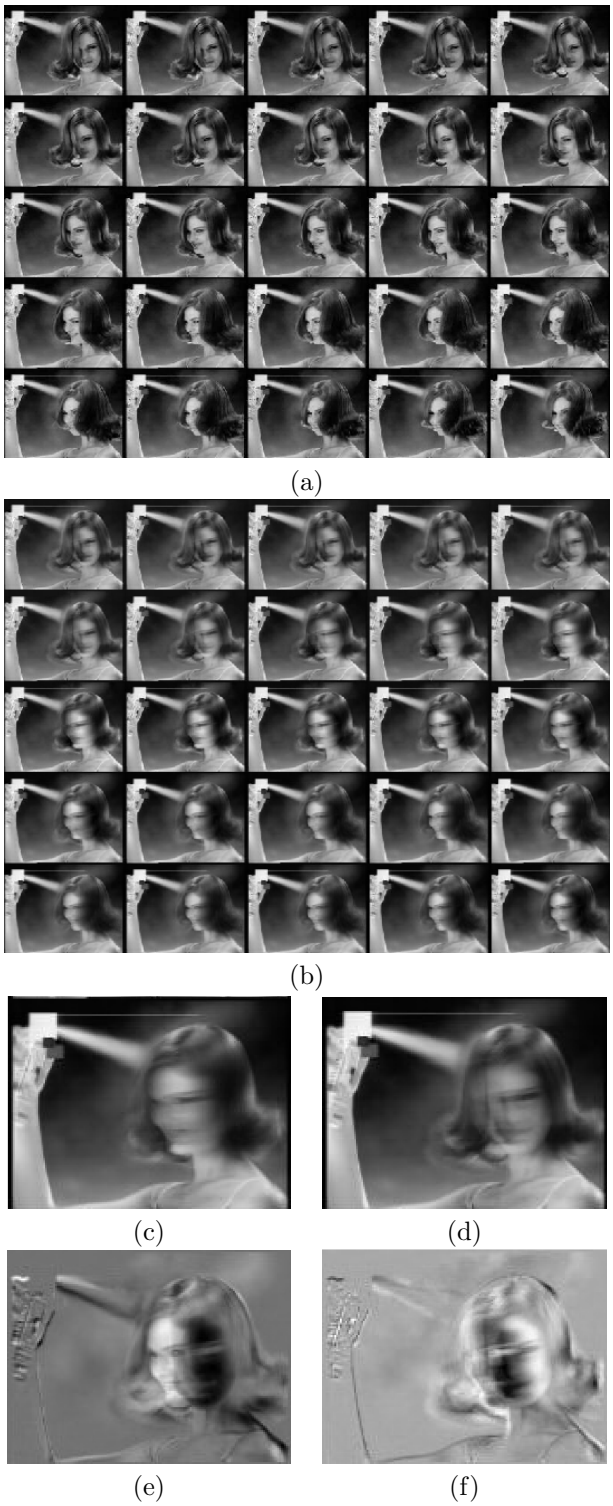


**Figure 6.7:** (a) Original sequence of frames. Reconstructed sequence (b) with 70.45% of reconstruction quality using 2 principal components of appearance (e) and (f) and a $2 \times 2$ $G$ matrix whose eigenvectors correspond to the images (c) and (d).

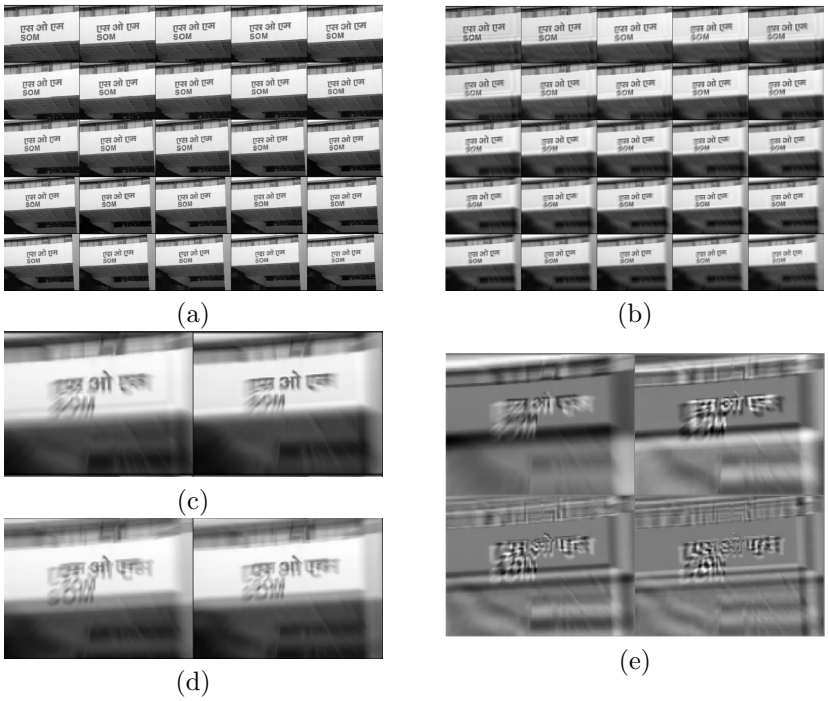



**Figure 6.8:** (a) Original sequence of frames. Reconstructed sequence (b) with 88.17% of reconstruction quality using 4 principal components of appearance (e) and a $4 \times 4$ $G$ matrix whose eigenvectors correspond to the images (c) and (d).

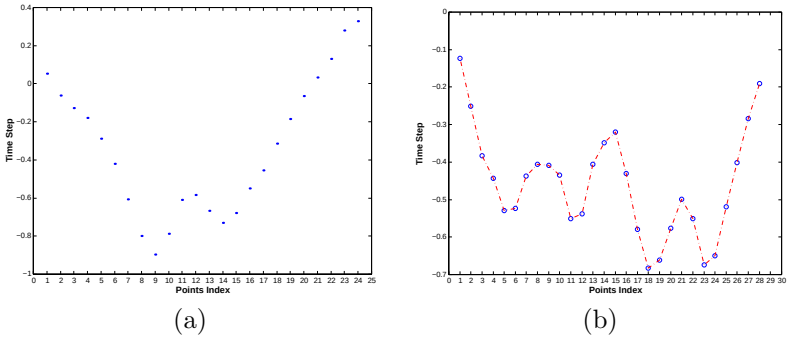



**Figure 6.9:** (a) Time step $\theta_n$ inferred values for each image of the sequence fig. 6.7(a). (b) Inferred $\theta_n$ values for the sequence fig. 6.8(a)

necessary to build the appearance representation by means of the ML for PPCA. Afterwards, utilizing the posterior probability eq. 6.8, the new coordinates for the images are computed though MAP. We use these new appearance representation to estimate the dynamical model by means of the introduced EM algorithm (see appendix). Once $G$ is estimated we compute the iconic representation, by means of an eigenvector analysis the infinitesimal generator. These eigenvectors are back-projected to the image space in order to have an expression of them as images.

The first sequence, fig. 6.7 (a), is represented by a 2-dimensional appearance eigenspace with a 70.45% of reconstruction quality. The appearance eigenvectors represented as images are shown in fig. 6.7(e) and (f). These express the variations between the mean or prototypical appearance of the object. However, such a prototype is not an appropriate representative sample, since the concept of "face" is not as much perceptible as in figs. 6.7(c),(d). Performing the estimation of the dynamical model $\{G, \beta^2\}$, we have the corresponding generator of the transformation os a $2 \times 2$ matrix whose modes correspond to the images fig.6.7 (c),(d). They are presented like real images, however, they are not directly obtained from the sequence, i.e., like selecting the first frame and last one. The significant issue here is that this temporal information can be used to reconstruct the video sequence (fig. 6.7 (b)) by means of eq. (6.5) and using just only the $2 \times 2$ matrix, the appearance basis and the time step $\theta_n$ of the evolution, (see fig. 6.9(a)).

The purpose of this second one sequence (fig. 6.8) is to expose that the appearance evolution model does afford to keep perceptually the camera operation in the new iconic representation. Given that editors, authors and artist communicates some specific intentions with certain camera operations, we notice that this information remains in the summarized representation. This sequence of frames (fig. 6.8(a)) has been represented in a 4-dimensional subspace with a 88.17 % quality of reconstruction. The 4 eigenvectors of appearance fig. 6.8(e) do not give a semantically meaningful idea of the images evolution, actually, we can see that they point out some zones of the image where the variations, at different scales, are produced. The temporal evolution was estimated by a $4 \times 4$ matrix, whose principal directions are represented as images in fig. 6.8 (c) and (d). They correspond to 2 complex (and conjugates) eigenvalues of $G$. So, using the theoretical background in LDS, we deduce that they are grouped in two classes, forming 2 invariant spaces: two vectors in fig. 6.8(c) and the other two in fig. 6.8(d). We can see what each one of them is expressing by focussing in the sign's letters of the images. The two first ones, fig. 6.8(c), are centered on a slow frequency (in time) variation, the sign is clearly readable and some blur only appears in the images' boundaries. The 2 second ones fig. 6.8 (b) are more focussed in high frequency time variations. We notice that the word "SOM" appears at separate different distances. The term *frequency* is applied here to the imaginary part of the complex eigenvalues. Actually, the frequency of the eigenvectors in fig. 6.8(c) is $\omega_1 = 0.36$, and the frequency of the eigenvectors in fig. 6.8 (d) is $\omega_2 = 0.61$. Therefore, we note that there is a direct relation between the imaginary values of $G$-eigenvalues and movement variations in the iconic representations.

It deserves to be commented that this dynamical representation has the ability

of reconstructing the sequence with a small number of parameters: figs. 6.7 (b) and 6.8(b). This fact allows the possibility of performing video preview with a low cost of streaming, and it is very appropriate for low bandwidth communication systems (like a modem at home). To this end, we need the eigenvectors that span the appearance representation, the matrix $G$ ($2 \times 2$ in fig. 6.7 and $4 \times 4$ in fig. 6.8), the reduced coordinate of the first image, and an 1-$D$ array of scalar values that represents the time step evolution from image to images. Thus, utilizing eq. (6.5) as a filtering process, the whole sequence is reconstructed. Note that our model includes an estimation of the time step between consecutive images (see figs. 6.9 (a),(b)), what differs from dynamical models that assume a constant time step evolution.

## 6.5 Summary and Conclusions

As an alternative to standard key-frames selection, in this chapter we propose a Bayesian framework for video summarization. We address the problem of characterizing key-frames basing partitions on appearance visual information criterion and, at the same time, conjugating semantic and temporal representations. This fact, not only allows embedding in a more numerical tractable framework the video *retrieval*, but also yields a new approach to extract underlying information from temporal evolution of sequences. A suitable selection of these basic perceptual units allows the transformation of a continuous temporal data structure into a discrete meaningful one, where the intention is that the semantics remains preserved. The choice of an appropriate representation for the data takes a significant relevance when it comes to deal with symmetries, since these usually imply that the number of intrinsic degrees of freedom in the data distribution is lower than the coordinates used to represent it. Indeed, this means that the problem can be reduced to a lower dimensional one. Therefore, using both topics; the decomposition into basic units and the change of representation, makes that a complex problem is transformed into a manageable one. This simplification of the estimation problem has to rely on a proper mechanism of combination of those primitives (appearance eigenvectors) in order to give an optimal description of the global complex model.

## Appendix: EM algorithm for Maximum Likelihood Estimation

In the Expectation-Maximization (EM) algorithm for symmetry estimation, we consider the latent variables $\theta_n$ to be "missing". If their value were known, the estimation of $G$ would be straightforward from equation (6.5) by applying standard least-squares techniques. This leads to consider a joint distribution over visible and hidden variables, then the corresponding distribution for the observed data is obtained by marginalisation. Thus the goal is to find the parameters that maximize the joint observed data distribution i.e. the best description under a specific generative

model. Assuming that we do not know for a given transformation $\Delta x_n$ which value of $\theta_n$ generated it, the joint distribution can be calculated through the *expectation* of the *complete data log-likelihood*.

From a Bayesian point of view, Maximum Likelihood estimation requires, in general, a two step procedure:

1. *Expectation* of latent variables $\theta_n$ for a given observed $\Delta x$. Posterior expectation given observed data for the complete log-likelihood is a way of computing an aproximation of the predictive density.
2. *Maximisation* of complete log-likelihood from the model parameters $G, \sigma^2$. This is equivalent to minimize the expectation of the loss function under that (approximated) predictive density.

**Expectation** The expected complete-data log likelihood is given by:

$$\begin{aligned} < \mathcal{L} > &= \sum_{n=0}^{N-1} \left\{ \frac{1}{2\sigma^2} < \mid \Delta x_n - \theta_n G x_n \mid^2 > + \right. \\ &+ \left. \frac{q}{2} \log \sigma^2 + \frac{< \theta_n^2 >}{2} \right\} \end{aligned}$$

which expanded yields:

$$\begin{aligned} \mathcal{L} &= \sum_{n=0}^{N-1} \left\{ \frac{1}{2\sigma^2} \mid \Delta x_n \mid^2 + \frac{1}{2\sigma^2} < \theta_n^2 > x_n^T G^T G x_n - \right. \\ &- \frac{1}{2\sigma^2} < \theta_n > x_n^T G^T \Delta x_n - \\ &- \left. \frac{1}{2\sigma^2} < \theta_n > \Delta x_n^T G x_n + \frac{< \theta_n^2 >}{2} \right\} + \frac{Nq}{2} \log \sigma^2 \end{aligned}$$

where the sufficient statistics of the posterior distributions correspond to:

$$< \theta_n > = M^{-1} x_n^T G^T \Delta x_n \tag{6.12}$$

$$< \theta_n^2 > = \sigma^2 M^{-1} + < \theta_n >< \theta_n > \tag{6.13}$$

with $M = \sigma^2 + x_n^T G^T G x_n$.

**Maximization** Differentiating equation (6.12) and setting the derivatives to zero, $G$ and $\sigma^2$ are updated as:

$$\hat{G} = \left[ \sum_n < \theta_n > \Delta x_n x_n^T \right] \left[ \sum_n < \theta_n^2 > x_n x_n^T \right]^{-1}$$

$$\begin{aligned} \sigma^2 &= \frac{1}{Nq} \sum_{n=1}^{N} \left\{ \mid \Delta x_n \mid^2 - 2 < \theta_n > x_n^T \hat{G}^T \Delta x_n + \right. \\ &+ \left. < \theta_n^2 > (\hat{G} x_n)^T (\hat{G} x_n) \right\} \end{aligned}$$

These equations are iterated until the algorithm converges with a certain degree of tolerance.

# Chapter 7

# Online Bayesian Video Summarization and Linking

In this chapter, an online Bayesian formulation is presented to detect and describe the most significant key-frames and shot boundaries of a video sequence. Visual information is encoded in terms of a reduced number of degrees of freedom in order to provide robustness to noise, gradual transitions, flashes, camera motion and illumination changes. We present an online algorithm where images are classified according to their appearance contents -pixel values plus shape information- in order to obtain a structured representation from sequential information. This structured representation is presented on a grid where nodes correspond to the location of the representative image for each cluster. Since the estimation process takes simultaneously into account clustering and nodes' locations in the representation space, key-frames are placed considering visual similarities among neighbors. This fact not only provides a powerful tool for video navigation but also offers an organization for posterior higher-level analysis such as identifying pieces of news, interviews, etc.

## 7.1 Introduction

Rich Media and content management has generated an enormous interest in video analysis within Computer Vision and Pattern Recognition communities. It offers a novel and exciting challenge for applying and developing new techniques in the recognition/classification framework. The addition of time to visual information analysis presents new constraints -a huge amount of information to be dealt with- and specific demands (such as real-time analysis) on the formulation of feasible and reliable techniques.

In this chapter, we focus on two important subjects in this area: video segmentation and summarization, which make feasible a quick intuition of the evolution, under

a low streaming cost, of higher-level perceptual structures, such as stories, scenes or pieces of news. Shot partitioning is considered as the extraction of the basic units for video analysis. Usually, shot boundary detection has been analyzed through feature based techniques, such as: pairwise pixel comparison [117], which are very sensitive to noise, color or grayscale histogram comparison [95, 115], which fails in distinguishing images with very different structures but similar color distributions, analysis of compressed streams [2, 3], and local feature based techniques [116]. However, many problems arise when it comes to dealing with noise, shape information, camera motion, illumination change, fades, and flashes. In these cases, specific problem-oriented techniques are applied: camera compensation [117], illumination reduction [119]. This sort of peculiarities are also treated by combining different measures [27]: dissolve, cut and fade measures. On the other hand, appearance based methods [38] use a representation where visual changes are encoded with a reduced number of degrees of freedom, and which provide more flexibility to the system in order to tolerate camera motions or illumination changes. However, the method presented in [38] requires loading the whole sequence. Offline techniques are not practical when dealing with large pieces of video. In this sense, online solutions, combined with certain robustness to gradual transitions and a reasonable computational complexity, are rather preferable.

To this end, we present a novel algorithm that provides an online treatment of video analysis plus the mentioned advantages of working under a Bayesian appearance-based framework. We address the problems of key-frame extraction and shot partitioning relying on a feature space where not only pixel value distributions (grayscale or color) are encoded but also shape information is taken into account. The algorithm online classifies the different shots of a video sequence and automatically extracts the most significant key-frames. Often, due to postproduction work (in commercials, movies, etc.), there are many sequences that contain the same shot in different time positions, that make standard algorithms to produce repeated key-frames and forcing posterior ad hoc merging/removing techniques in order to avoid unnecessary redundancies. Given that the algorithm is embedded in a Bayesian formulation, questions such as sufficient number of key-frames to represent a video sequences or avoiding extra key-frame detection due to flashes, are automatically solved.

The problems of key-frame extraction and shot partitioning are treated in terms of a probabilistic unsupervised learning approach. Each frame of a video sequence is assumed to belong to a cluster of images that are related in terms of their appearance contents, and, each cluster has a representative image that will be used for summarization purposes, i.e., a key-frame. The algorithm's process is controlled by a tuning parameter whose range embraces appearance representation [97, 67, 106] (such as PCA, FA or NNMF techniques) and hard clustering (competitive learning). The goal of the learning is to identify the latent variables (weights) and the unknown mapping parameters (key-frames). For this purpose, we present the estimation process in the context of the Expectation-Maximization algorithm.

The outline of the chapter is as follows: first, we describe the basis of our approach in section 2. There, the model that connects a video sequence with a $2D$ graph representation is presented. We provide an EM algorithm for estimating the

parameters of the model in two versions: offline and online. The experimental results analyze: (i) the effects of the tuning parameter that makes the algorithm embracing techniques compressed representation of appearance and clustering techniques, (ii) the effectiveness of dealing with prior information in order to automatically select the number of necessary key-frames to represent a video sequence.

## 7.2 The Model: Bayesian Framework

In order to describe our model, we first define the latent space were observations (images) are represented. We consider this latent space to be a grid of $M$ nodes represented by a set of vectors $\{\mathbf{c}_1, \ldots, \mathbf{c}_M\}$. Each node represents a class of images that are similar in terms of their appearance contents. Consider a set of $N$ images $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ in a vector form read in lexicographic order, and each image representation (location) on the $2D$ grid (latent variables) as $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. For each latent variable $\mathbf{x_n}$ we like to know the contribution to its generation for each node $\mathbf{c}_m$ in the grid. A measure that quantifies such a contribution can be expressed in terms of a Bayesian framework by the posterior probability $P(\mathbf{c}_m \mid \mathbf{x}_n)$. The model will provide the similarity measure that relates the ownership of an image under a specific class. Consider each node $\mathbf{c}_m$ has a representative image $\mathbf{w}_m$ that summarizes the images' appearance contents that belong to the $m$-class. In this sense, we can consider each image to be a weighed combination of the summarizing representative images $\mathbf{w}_m$, where the weights correspond to the posterior probabilities $P(\mathbf{c}_m \mid \mathbf{x}_n)$.Therefore, we can construct a model where images and latent variables ($2D$ points on the grid) are related by the following mapping:

$$\mathbf{y}_n = \sum_{m=1}^{M} \mathbf{w}_m P(\mathbf{c}_m \mid \mathbf{x}_n) + \mathbf{e}_n \tag{7.1}$$

where $\mathbf{e}_n$ is gaussian independent identically distributed (idd) noise $\mathcal{N}(0, \sigma^2 I)$. Although equation (7.1) has the form of well known linear decompositions, PCA or FA, it is worth noting that the posterior probabilities $P(\mathbf{c}_m \mid \mathbf{x}_n)$. are restricted to be non-negative and the sum to be the unity. In the following sections, we show that the nature of these posterior probabilities determines whether the model can be used for performing hard clustering (key-frame extraction and shot partitioning applications) or a compressed representation of images (dimensionality reduction). The model has two main issues: on one hand, the estimation of the representative images $\mathbf{w}_m$ and the posterior probabilities $P(\mathbf{c}_m \mid \mathbf{x}_n)$, and on the other, the inference for the images' location in the latent space (grid).

### Noise model

First, the noise model is expressed through a Gaussian distribution for the data: $\mathcal{N}(\mathbf{y}_n - W\mathbf{h}_n, \sigma^2)$, where W is a matrix whose columns are the vectors $\mathbf{w}_m$ and $\mathbf{h}_n$ is an array of the posterior probabilities for the image $\mathbf{y}_n$: $\mathbf{h}_n = (\, P(\mathbf{c}_1 \mid \mathbf{x}_n), \ldots, P(\mathbf{c}_M \mid \mathbf{x}_n))$.

A Bayesian treatment of this model is obtained by introducing a prior distribution over the components $\{\mathbf{w}_1, \ldots, \mathbf{w}_M\}$. The key point is to control the effective number of sufficient parameters (number of classes). This is achieved by introducing a prior distribution $P(W \mid \alpha)$, where $\alpha$ is a $M$-dimensional vector of hyper-parameters $\{\alpha_1, \ldots, \alpha_M\}$. Each hyper-parameter $\alpha_m$ controls one of the cluster representative vector $\mathbf{w}_m$ by means of the following distribution: $P(W \mid \alpha) = \prod_{m=1}^{M} \mathcal{N}(\mathbf{w}_m, \alpha_m^{-1})$ Each hyper-parameter $\alpha_m$ corresponds to an inverse variance. For large values of $\alpha$ the corresponding $\mathbf{w}_m$ will tend to be small, and therefore, such component will be neglected. These hyper-parameters behave as switchers, activating or deactivating each component $\mathbf{w}_m$. Typically, the selection for the distribution of $\alpha_m$ corresponds to Laplace distributions or Gamma distributions due to their properties on "pruning". We select a gamma distribution for the hyper-parameters, since it offers a tractable analytical treatment for the estimation process: $P(\alpha) = \prod_{m=1}^{M} \Gamma(\alpha_m \mid a, b)$, where

$$\Gamma(\alpha_m \mid a, b) = \frac{b^a (\alpha_m)^{a-1} e^{-b\alpha_m}}{\Gamma(a)} \tag{7.2}$$

We select $a = 10^{-3}$ and $b = 10^{-3}$, which are the magnitude orders typically selected in this framework.

**Modeling the latent space**

The distribution on the grid is modeled as a mixture of unimodal distributions. For instance, one might select a mixture of Gaussians, however, the algorithm that we present can be easily modified with another type of density functions (laplacians, gamma, etc.). The likelihood measure for a single point in the latent space is given by: $P(\mathbf{x}_n) = \sum_{m=1}^{M} P(\mathbf{c}_m) P(\mathbf{x}_n \mid \mathbf{c}_m)$, where we select the conditional distributions to have the form of Gaussian distributions: $P(\mathbf{x}_n \mid \mathbf{c}_m) = \mathcal{N}(x_n - \mathbf{c}_m, \tau_m^2)$.

### 7.2.1 Parameter Estimation

In this section, we present the framework where the parameters of this model are estimated. The estimation process has to take into account two issues at the same time: the noise model and the density distribution in the latent space. In the estimation procedure, there are two main steps communicated by a feedback process. Given a set of images, the cluster representative images $\mathbf{w}_m$ and the posterior probability vectors are computed in order to infer the location of each image on the grid. This is done through posterior expectation of the nodes, i.e.:

$$< x_n >= \sum_{m=1}^{M} \mathbf{c}_m P(\mathbf{c_m} \mid \mathbf{x}_n) = \sum_{m=1}^{M} \mathbf{c}_m h_n^m \tag{7.3}$$

These expected locations on the grid are used as data for estimating the grid's density distribution parameters, i.e., the variances $\tau_m^2$, which play an important role as scale factors in the clusters distribution topography. In addition to this, the posterior

probabilities are then recomputed using the Bayes' rule. Since these probabilities contribute to the re-estimation of the components $\mathbf{w}_m$, now, the estimation of the noise model parameters takes into account the topographical distribution of the clusters in the latent space.The following table summarizes the parameters to be estimated according to these two steps:

| Model | Parameters | |
|---|---|---|
| Noise | $\mathbf{w}_m$ | Cluster representative images |
| | $\mathbf{h}_n$ | Posterior Probabilities |
| | $\sigma^2$ | Noise variance |
| | $\alpha_m$ | Switchers |
| Latent Space | $< \mathbf{x}_n >$ | Expected location |
| | $\tau_m$ | Node variance |

We embed the estimation process in the framework of the Expectation-Maximization (EM) algorithm, which is useful to find maximum likelihood parameter estimates in problems where some variables are unobserved. In our case, posterior probabilities and posterior location points are unobserved. The M step maximizes w.r.t. the model parameters $(\mathbf{w}_m, \sigma^2, \alpha_m, \tau_m)$ and the E step maximizes it w.r.t. the distribution over the unobserved variables $(\mathbf{h}_n, < \mathbf{x}_n >)$. Typically, the algorithm consists of a set of fixed-point type equations that are iterated until convergence. In the following section, we show the procedure to estimate both sets of parameters and latent variables.

### EM Algorithm

The maximum likelihood estimation for the noise model parameters can be equivalently performed in terms of maximizing the logarithm of the joint distribution:

$$\mathcal{L} = -\frac{1}{2}\sum_{n=1}^{N} \mid \mathbf{y}_n - W\mathbf{h}_n \mid^2 - \frac{1}{2}\sum_{m=1}^{M}\left\{\alpha_m|\mathbf{w}_m|^2 + \frac{d}{2}\log\alpha_m - \log\Gamma(\alpha_m \mid a, b)\right\} \tag{7.4}$$

Given an initial guess for the parameters and unobserved variables iterate:

- **Expectation:** Find $\mathbf{h}_n$ that maximizes (7.4). Given the constraints for $\mathbf{h}_n$ -non-negativity and normalization- we need to apply a exponentiated gradient method [55] to ensure that the new estimates are always positive. This is done by introducing an auxiliary function as in [31]. In order to derive this update rule, we make use of an auxiliary function $G(\mathbf{h}_n, \mathbf{h^t}_n)$ such that $G(\mathbf{h}_n, \mathbf{h}_n) = \mathcal{L}(\mathbf{h}_n)$ and $G(\mathbf{h}_n, \mathbf{h^t}_n) \leq \mathcal{L}(\mathbf{h}_n)$ for all $\mathbf{h^t}_n$. For this auxiliary function, it can be seen that $F$ is nondecreasing after the update $\mathbf{h}_n^{t+1} = \arg\max_{\mathbf{h}_n} G(\mathbf{h}_n, \mathbf{h^t}_n)$. So the update rule is given by making $\partial G(\mathbf{h}_n, \mathbf{h}_n^t)/\partial \mathbf{h}_n = 0$ on each step. An auxiliary function for $\mathcal{L}(\mathbf{h}_n)$ is constructed as,

$$G(\mathbf{h}_n, \mathbf{h}_n^{t+1}) = -\frac{1}{2}\sum_{n=1}^{N} \mid \mathbf{y}_n - W\mathbf{h}_n^{t+1} \mid^2 - \nu\sum_{k=1}^{M} h_{kn}^{t+1}\log\frac{h_{kn}}{h_{kn}^{t+1}} \tag{7.5}$$

  which leads to the following update rule:

$$h_{kn}^{t+1} = h_{kn}\frac{\exp\left\{\frac{\nu}{2\sigma^2}\left[W'(\mathbf{y}_n - W\mathbf{h}_n)\right]_k\right\}}{\sum_{i=1}^{M} h_{in}\exp\left\{\frac{\nu}{2\sigma^2}\left[W'(\mathbf{y}_n - W\mathbf{h}_n)\right]_i\right\}} \tag{7.6}$$

Note that there has been introduced a scale parameter $\nu \in [0, \infty)$, which controls the degree of change from the old estimate to the new one. This parameter plays an important role, since its value determines whether the algorithm is performing vector coding (clustering when $\nu \to \infty$) or appearance encoding when $\nu \to 0$ (such as PCA, FA techniques). After estimating the posterior probabilities $\mathbf{h}_n$, the estimation of the images positions on the grid in done by means of eq. (7.3).

- **Maximization:** Given the inference of the positions on the grid $\mathbf{x}_n$, we can compute the node variances: $\tau_m = \frac{1}{2N} \sum_{n=1}^{N} h_{nm} \mid \mathbf{x}_n - \mathbf{c}_m \mid^2$. And therefore, the posterior probabilities under the grid model making use of the Bayes' rule. We define the new computed posterior probabilities as vectors $\varphi_n = [P(\mathbf{c}_1 \mid \mathbf{x}_n), \ldots, \mathbf{c}_M \mid \mathbf{x}_n)]$. These are used for computing the cluster representative images:

$$W = \left[\sum_{n=1}^{N} \mathbf{y}_n \varphi_n'\right] \left[\sum_{n=1}^{N} \varphi_n \varphi_n' + diag(\alpha_1, \ldots, \alpha_M)\right]^{-1} \tag{7.7}$$

and the noise variance: $\sigma^2 = \frac{1}{ND} \sum_{n=1}^{N} \mid \mathbf{y}_n - W\varphi_n \mid^2$, where $D$ is the images' dimension ($\mathbf{y}_n$). Finally, the switchers are computed as follows:

$$\alpha_m = \frac{a + \frac{D}{2}}{b + \mid \mathbf{w}_m \mid^2} \tag{7.8}$$

### Incremental Learning

The manner the algorithm has been formulated implies that all data is necessary in the M step. When dealing with large data sets, this sort of batch algorithms may incur in computational memory problems. It could be more useful updating the parameters incrementally using data points one at a time. To this end, it is shown in [69] that, under some specific conditions, an EM algorithm can be performed incrementally converging to a local maximum as well. First condition is that the joint probability distribution over the observed data factorizes, and, another condition is that the sufficient statistics can be expressed as a sum over the contribution of each single sufficient statistics' point. In our case, the problem is consistent with these conditions. Therefore the update rules for each new point are:

**Expectation:** Compute the posterior probabilities as in eq.(7.5) an (7.3). This step does not change.

**Maximization:**

$$\tau_m = \tau_m^{new} + \frac{1}{2N} h_{N+1,m}^{new} \mid \mathbf{x}_{N+1}^{new} - \mathbf{c}_m \mid^2 - \frac{1}{2N} h_{N+1,m} \mid \mathbf{x}_{N+1} - \mathbf{c}_m \mid^2$$

For a given $A = \left[\sum_{n=1}^{N} \mathbf{y}_n \varphi_n'\right]$ and $B = \left[\sum_{n=1}^{N} \varphi_n \varphi_n'\right]$ set:

$$A^{new} = A + \left[\mathbf{y}_{N+1} \varphi_{N+1}^{new'} - \mathbf{y}_{N+1} \varphi_{N+1}'\right]$$

$$\begin{aligned} B^{new} &= B + \left[\varphi_{N+1}^{new'}\varphi_{N+1}^{new'} - \varphi_{N+1}\varphi'_{N+1}\right] \\ W^{new} &= A^{new}[B^{new} + diag(\alpha_1, \ldots, \alpha_M)]^{-1} \end{aligned}$$

and

$$[\sigma^2]^{new} = \sigma^2 + \frac{1}{ND} \mid \mathbf{y}_{N+1} - W^{new}\varphi_{N+1}^{new} \mid^2 - \frac{1}{ND} \mid \mathbf{y}_{N+1} - W\varphi_{N+1} \mid^2$$

Finally, the switchers $\alpha_m$ are computed as in eq.(7.8). These two steps are iterated for each new image $\mathbf{y}_{N+1}$.

## 7.3 Experiments

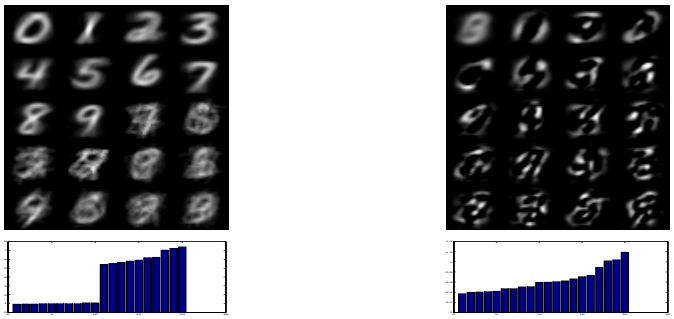
In this section, we analyze the introduced algorithm. We specially emphasize two facts: the effects of introducing switchers and the consequences of selecting a specific value for the tuning parameter $\nu$. In the first case, we obtain from the switchers the necessary number of classes to represent a video sequence. With regard to the selection of the tuning parameter, we show how the algorithm embraces vector coding (clustering) and appearance learning encoding.

### 7.3.1 Features of the tuning parameter

The aim of this experiment is to show the effects of selecting a specific value for the tuning parameter. For this purpose, we chose the MNIST[39] data set of handwritten digits. In this case, we select 10 $\mathbf{w}_m$ components. Figure 7.1 shows two sets of images: left one corresponds to perform the learning process using a relatively high tuning parameter $\nu = 3.5$, and figure 7.1(b) is the result of using a low tuning parameter $\nu = 1$. Notice that, in the first case and according to equation 7.5, for each sample $\mathbf{y}_n$ the posterior probabilities $\mathbf{h}_n$ are forced to be close to zero except one which corresponds to the biggest one and its value is assigned to be close to the unity. This fact forces the learning process to perform hard clustering. On the other hand, when the tuning parameter is set to $\nu = 1$ the learning process is finding a compressed representation for the observed data through a sparse basis. Finally, the extreme case of $\nu = 0$ makes all the posterior probabilities to have the same value, which means that all the basis $\mathbf{w}_m$ components will be the same.

### 7.3.2 Applications to Video Analysis

The purpose of this experiment is twofold, we address the problem of characterizing key-frames basing partitions on appearance visual information criterion, and in addition to this, we show the use of the switchers that provide a measure for deciding how many components (key-frames) are necessary for a given video sequence. To this end, we chose an interview of 20 minutes (30000 frames), where there are mainly three camera shots. Note that a priori we are not assumed to know how many shots will be



(a) $\nu = 3.5$ (b) $\nu = 1$

**Figure 7.1:** Basis components $\mathbf{w}_m$ resulting of running our algorithm on the MNIST data set. (a) Hard clustering corresponds to perform the learning process using a relatively high tuning parameter $\nu = 3.5$ and (b) is the result of using a low tuning parameter $\nu = 1$.

on the interview. Therefore, we selected 5 possible classes. Since we are focused on clustering we selected a relatively high value for the tuning parameter $\nu = 3.5$. Figure 7.2 shows the results of the algorithm's performance under these conditions. First, the resulting key frames are shown in fig. 7.2(a), where one of them corresponds to a meaningless image. The switcher value corresponding this particular "key-frame" is the highest one (figure 7.2(b)); therefore as mentioned in section 2.2.1 it can be neglected. In this sense, we describe this interview in terms of four key-frames that corresponds to four different camera shots (fig. 7.2(a)). Adding time as a third dimension to the grid, we can have in figure 7.2(c) notion of the evolution of the sequence in terms of linking in time the four clusters. For visualization purposes we have skimmed the total contents of sample images in this last figure 7.2(c). It took 1 hour and 24 minutes to do the computation in MATLAB; however, we believe that the same algorithm under C++ will speed up the process at least in a factor of 3, providing a real time solution for video indexing and annotation. In figure 7.3, we analyze our algorithm when it comes to dealing with professionally produced material such as commercials. The one[1] we analyze contains flashes, illumination changes, and fast camera movements. Performing pixelwise comparison between consecutive frames flashes, many images are detected as key-frames because of rapid camera or objects movements, flashes and illumination changes. This makes the resulting storyboard to be redundant in these specific types of scenes. Adding flexibility to the system by means of codifying appearance in terms of a few degrees of freedom, we can detect representative images and determine their corresponding relevance (bottom of fig 7.3(a)).

[1] All material -videos and images- can be found at the following URL : `http://www.cvc.uab.es/ xevi/videos/`

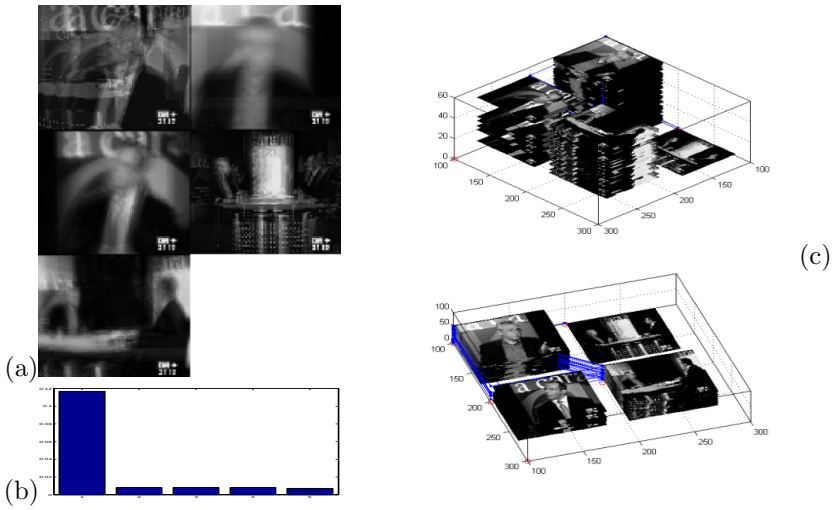



**Figure 7.2:** (a) Cluster means, where key-frame selection can consist of picking up the closest image to each of them.(b) Switchers. (c) Adding time to the grid space.

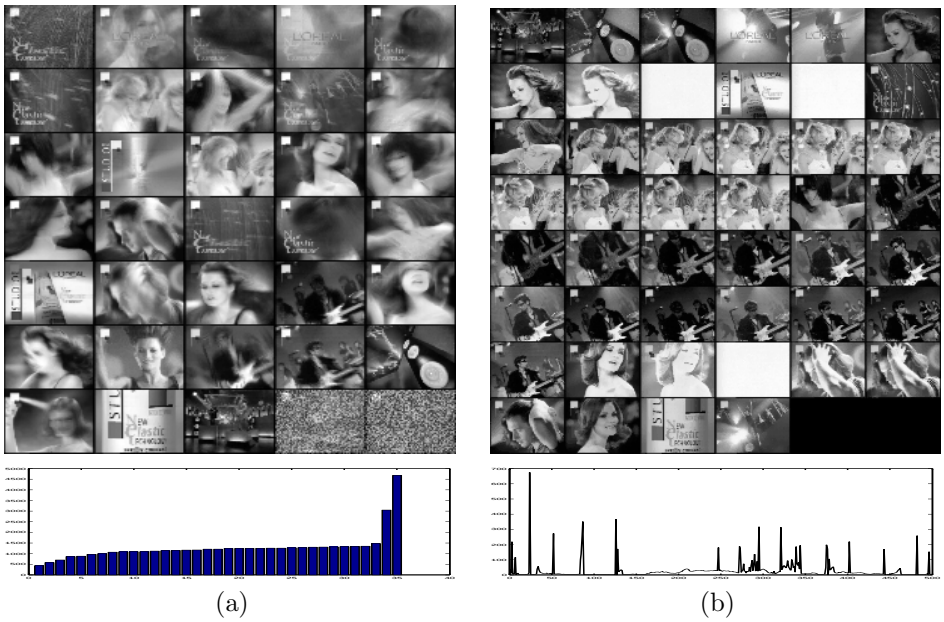



**Figure 7.3:** (a) Cluster means extracted from a commercial using our algorithm and using pixelwise comparison (b). Note that in (a) key-frames are presented in this case by criteria of key-frames' relevance. The figure is read in lexicographic order.

## 7.4 Conclusions

As an alternative to standard feature-based key-frames selection, in this chapter we propose a Bayesian framework for video summarization. We address the problem of characterizing key-frames basing partitions on appearance visual information criterion. This fact, not only allows embedding in a more numerical tractable framework the video *retrieval*, but also yields a new approach to extract underlying information from temporal evolution of sequences. A suitable selection of these basic perceptual units allows the transformation of a continuous temporal data structure into a discrete meaningful one, where the intention is that the semantics remains preserved.

# Chapter 8

# Analyzing Periodic Motions in Video Sequences

In this chapter, we present a new technique for separating different types of periodic motions in a video sequence. We consider different motions those that have different periodic patterns with one or many fundamental frequencies. We select the temporal Fourier Transform for each pixel to be the representation space for a sequence of images. The classification is performed using Non-Negative Matrix Factorization (NNMF) over the power spectra data set. The chapter we present can be applied on a wide range of applications for video sequences analysis, such as: background subtraction on non-static backgrounds framework, object segmentation and classification. We point out the fact that no registration technique is applied in the method that we introduce. Nevertheless, this method can be used as a cooperative tool for the existing techniques based on camera motion models (motion segmentation, layer classification, tracking of moving objects, etc).

## 8.1 Introduction

The aim of finding periodicities in image sequences goes back to the beginnings of Computer Vision. Many biological reasons support the idea of dealing with this specific issue. Periodic motion detection is a strong cue for object and action recognition in human motion perception [37, 51]. Actually, studies on recognizing moving light displays show the ability of human perception for recognizing biological motion [37, 51]. Even when dealing with very low resolution image sequences, humans are capable of recognizing periodic movements [28].

As it has been pointed out in [92, 64, 79], periodicity is striking in that it can be detected without taking into account the structure of objects in a scene (rigid and non-rigid objects are accounted for), and, at the same time, techniques for periodic motion

detection, segmentation and classification can assist in many applications requiring object and activity recognition and representation [63, 62, 70, 22].

Recent analyses that categorize the existing methods for periodic motion recognition and segmentation can be found in [28, 47], and can be summarized into: *Fourier based* methods [79, 64, 28], *Point correspondences* based methods [92], *Linear Dynamic models* methods [22], fitting spatio-temporal surfaces [70], and *Flow-based* methods [78, 62]. Many of them use spatio-temporal alignment, background subtraction and tracking techniques for targeting periodic patterns. The work we present in this chapter is certainly compatible with these techniques, even though, for the purpose of this chapter, they are not the main point of discussion.

### 8.1.1 Contribution

We present a novel technique to deal with a new and interesting problem, which can be stated as follows: How many different types of periodic movements are in a specific scene? Is it possible segmenting different objects from their motion when: a) there are occlusions in the scene across time and b) the same object has disconnected parts? Both questions have an answer when studying the global behavior of a sequence that contains different objects moving with different periodic movements. The algorithm we propose yields a manner for detecting in each frame : i) which pixels correspond to a specific object? and ii) which are the fundamental frequencies that contribute to its motion? The referenced works were about detecting periodicities and segmenting a particular region where periodic movements occur, however no classification for different periodic movements in the same scene was proposed.

Sequences of images with periodic motions are characterized by having some of their pixels with a certain pattern of repetition on their values (gray, color) across time. If the temporal length of the sequence is larger than the period of the object's motion, there are pixels that show a pattern of transitions with a certain frequency. The temporal Fourier representation of each pixel location ($i$-row, $j$-column) evolution across time is used for the analysis of this specific periodic pattern. However, there is a certain amount of variability among temporal Fourier spectra due to two main factors: a) the different periodic movements that occur in the scene and b) the object's shape (even more variability when dealing with non-rigid objects).

The technique we present can be used when dealing simultaneously with moving objects plus moving backgrounds such as: waterfalls, waves, smoke, etc. Typically, this sort of backgrounds are considered video textures. The main problem, in this case, is when approaching them with either background subtraction or spatio-temporal alignment techniques, since the collection of pixels belonging to the background do not correspond to a pure camera transformation and they are not static in their pixel value (gray, color). We treat the background with no particular distinction among the rest of pixels.

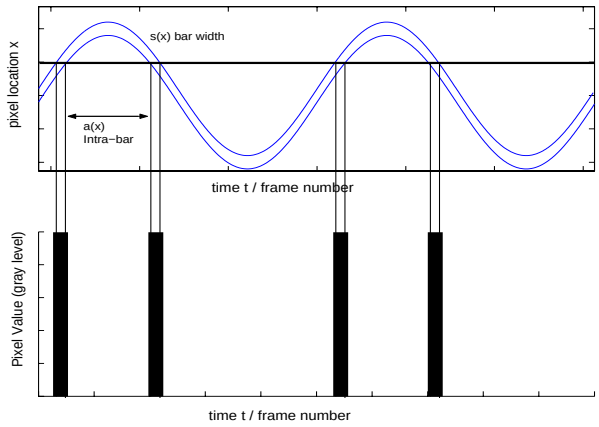



**Figure 8.1:** One-dimensional sequence of images and temporal signal for a specific pixel location

### 8.1.2 Outline

First, we build a model for one-dimensional images in order to analyze the different contributions of shape, motion and frame-rate to the Fourier power spectra. In section 3, we present a brief study on the reliability of periodic motion classification and power spectrum factorization. A extended version of this analysis can be found in [73]. The segmentation of multiple periodic moving objects in video sequences is based on the formulation presented in section 4. The algorithm is shown in section 5. Section 6 presents a set of experiments in order to show the algorithm's performance. Finally, the conclusions are presented in section 7.

## 8.2 Periodic Motion Analysis

In this section, we justify how the fundamental frequency can be extracted from a set of observed images. To this end, we show an example that deals with one dimensional images. It can be directly extended to 2-$D$ images.

Let $I(x)$ be an one dimensional image with $d$ pixels, where $x$ indicates the pixel position. The following example shows an oscillating spot of length $L$ and amplitude $A$ across time. Therefore, the observation is a set of $N$ images with the spot at different positions (see fig.8.1). The frequency of oscillation is $\omega_0 = 2\pi/T_0$, and we consider those cases where $T_0 < N$. For each pixel position, there is a 1-$D$ periodic signal which consists of a pattern of bars with amplitude $s$ and intra-bar separation $a$.

The size of the object $L$ and the frequency of oscillation determine the behavior of $s$ and $a$ at each height $x$ location. The oscilation model corresponds to the domain

defined by the two following boundary signals:

$$f_1(t) = A\cos(\omega_0 t) + \frac{L}{2} \tag{8.1}$$

$$f_2(t) = A\cos(\omega_0 t) - \frac{L}{2} \tag{8.2}$$

We can see that the length of the object for image in the sequence is $f_1(t) - f_2(t) = L$, with $L \geq 0$. For a specific pixel position $x$ whithin the oscillation amplitude interval, there is a 1-$D$ signal that corresponds to the intersection of $x$ with each of the two boundaries (eqs. (8.1) and (8.2)): $x = A\cos(\omega_0 t_1) + \frac{L}{2}$ and $x = A\cos(\omega_0 t_1) - \frac{L}{2}$. Therefore, $x$ intersects at $t_1$ and $t_2$ yielding the bar width $s = \mid t_2 - t_1 \mid$:

$$s(x, A, L, \omega_0) = \frac{1}{\omega_0}\left|\cos^{-1}\left(\frac{x + \frac{L}{2}}{A}\right) - \cos^{-1}\left(\frac{x - \frac{L}{2}}{A}\right)\right| \tag{8.3}$$

The intra-bar separation $a$ can be writtem in terms of $(x, A, L, \omega_0)$:

$$a(x) = \frac{1}{\omega_0}\left|2\pi - 2\cos^{-1}\left(\frac{x + \frac{L}{2}}{A}\right) - \cos^{-1}\left(\frac{x - \frac{L}{2}}{A}\right)\right| \tag{8.4}$$

For each pixel location $x$, there is a temporal periodic signal with $(a, s, T_0 = \frac{2\pi}{\omega_0})$. Let $Q_a(t)$ be defined as a step function defined as follows:

$$Q_s(t) = \begin{cases} 1 & \text{if} \quad 0 \leq t \leq s(x) \\ 0 & \text{else} \end{cases} \tag{8.5}$$

The temporal signal for a specific pixel location $x$ can be defined as follows:

$$f(t) = \sum_n Q_{s(x)}(t - nT_0) + \sum_n Q_{s(x)}(t - nT_0 - a(x)) \tag{8.6}$$

Assuming a frame-rate sufficiently fast to capture the periodicity, the power spectrum at $x$ can be written as follows:

$$S(\omega, x) = 16\sum_n \delta(\omega - \omega_0 n)\frac{1}{\omega_0^2 n^2} U_n(\omega_0 a(x)) U_n(\omega_0 s(x)) \tag{8.7}$$

where $U_n(z) = [1+\cos(nz)]^2$ has been defined for notation simplicity. From equations (8.4) and (8.3) we note that $U_n(\omega_0 a(x))$ and $U_n(\omega_0 s(x))$ do not actually depend on the fundamental frequency $\omega_0$. This implies that the power spectrum consist of the contribution of two terms of different nature: i) one corresponding to the sampling effect due to the fundamental frequency $\omega_0$, and ii) another term corresponding to the contribution of the pixel location $x$ and the object's shape parameters:

$$S(\omega, x) = \sum_n \delta(\omega - \omega_0 n) H_n(A, L, x) \tag{8.8}$$

Moreover, the discretization effect due to the number of frames in the sequence makes equation (8.8) to be re-written approximately [73] as follows:

$$S(\omega, x) \approx \sum_{k=0}^{N-1} \delta(\omega - k) \sum_{n=0}^{T_0} \delta(\omega - \frac{N}{T_0} n) H_n(x) \tag{8.9}$$

This is just an approximation of a sum of exponentials, however, for our range analysis we will further show that is very useful since it allows to study the influence of $H_n(x)$ on the spectra obtained from the observations. It is worth to note that when $T_0$ approaches to $N$, i.e., the ratio $\frac{N}{T_0} \to 1$, no information about the fundamental frequency can be extracted from the observations. The corresponding spectra miss the common property that allows to identify them as the result of the same motion origin.

## 8.3 Variance

The purpose of this section is to study the possibility of classifying different spectra belonging to different types of periodic motions. Consider a sequence of two moving objects with fundamental frequencies $\omega_1$ and $\omega_2$. The resulting spectra for each pixel position are:

$$\begin{aligned} S_1(\omega, x) &\approx \sum_{k=0}^{N-1} \delta(\omega - k) \sum_{n=0}^{T_1} \delta(\omega - \frac{N}{T_1} n) H_n^1(x) \\ S_2(\omega, x) &\approx \sum_{k=0}^{N-1} \delta(\omega - k) \sum_{n=0}^{T_2} \delta(\omega - \frac{N}{T_2} n) H_n^2(x) \end{aligned}$$

The aim of this example is to analyze the variance due to the pixel position in comparison with the average difference between the two types of spectra. Let us call *intra-class* difference to the variance due to the pixel position (and object's shape), and *inter-class* difference to the average difference between the two spectra.

A symmetrical measure that express the inter/intra-class ratio variance can be the geometrical mean of the ratios: $d(S_1, S_2)/ < \Delta S_1 >$ and $d(S_1, S_2)/ < \Delta S_2 >$, which is expressed in terms of the periods $T_1$ and $T_2$ as follows [73]:

$$R_{S_1,S_2} = 4\sqrt{\left(1 + \frac{T_1^4}{T_2^4}\right)\left(1 + \frac{T_2^4}{T_1^4}\right)} \tag{8.10}$$

The bigger is the ratio between the two periods $T_1/T_2$ the bigger is the variance $R_{S_1,S_2}$, and therefore, the most distinguishable are the two types of motion. For instance, consider $T_1$ being twice $T_2$, therefore, $R_{S_1,S_2} \approx 16$ times between intra-class and inter-class. This yields to study the possibility of applying statistical techniques to segment the different types of periodic movements that occur in an image sequence.

## 8.4 Segmenting Different Periodic Motions

The fact that two different motions are sampled different, the distance between them is much bigger, than the differences due to shape and pixel location of different spectra originated by the same periodic motion as it is shown in equation (8.10). Therefore, the fact of factoring the power spectra does not block the possibility of segmenting different types of periodic motions:

$$S(\omega, x) = \sum_{n=1}^{T_0} W_n(\omega) H_n(x) \approx \bar{W}(\omega) \bar{H}(x) \tag{8.11}$$

This approximation is the central point for analyzing the segmentation of periodic motions. Since the contribution of shape has been minimized through this approximation, we can see that the method can deal with non rigid objects (no assumption based on rigidity has been made).

An similar procedure can be performed for modelling movements with multiple periods corresponding to the same object. In this case, factorization is assumed to follow the approximation in equation (8.11) as well, where the discrete sampling effect will affect to different frequencies in the spectra. The goal is to distinguish different

types of motion, and it is ensured by the upper bounds described before. When dealing with multiple motions, we assume that the spectrum providing from a specific pixel location $x$ will contain a linear superposition of the generative individual spectra. If a sequence of images, contains more than one moving object, it is expected that at different time values a pixel location will have information about some of them. This fact includes occlusions among different objects. Of course, the definition of object here embraces both types rigid and non-rigid, with means that its definition is based on motion particularly.

For a model that assumes different periodic moving objects, the idea is that the power spectrum at each pixel location $x$ factorizes as follows:

$$S(\omega, x) = \sum_{k=1}^{q} \bar{W}_k(\omega)\bar{H}_k(x) \tag{8.12}$$

where $q$ different types of movements have been assumed. Further, we embed this model into a Bayesian framework in order to select from data the number of possible different types of motion automatically. In order to analyze the linear superposition assumption in the spectra, we refer to the linear property of the Fourier transform, and the fact the interferences when computing the power spectra are taken into account in the variance analysis (intra and inter class).

The parameter estimation has to take into account the fact that $S(\omega, x)$,$\bar{W}_k(\omega)$ and $\bar{H}_k(x)$ are non negative. To this end, we base our method on the technique presented in [60]. The error function to minimize takes into account both the reconstruction error and a prior function over $\bar{H}_k(x)$ in order to automatically control the effective number of sufficient parameters (number of possible moving objects $q$). Therefore, a set of hyper-parameters $\{\alpha_1, \ldots, \alpha_q\}$ is introduced in order to behave as switchers; activating or deactivating the components $\bar{H}_k(x)$. For large values of $\alpha_k$ the corresponding component will tend to be small, and therefore, such component will be neglected.

$$\mathcal{E} = \frac{1}{\sigma^2} \sum_x \sum_\omega \left| S(\omega, x) - \sum_{k=1}^{q} \bar{W}_k(\omega)\bar{H}_k(x) \right|^2 + \sum_{k=1}^{q} \alpha_k \sum_x \bar{H}_k(x) \tag{8.13}$$

The update rules that take into account non-negativity are:

$$\bar{H}_k(x)^{t+1} = \bar{H}_k(x)^t \left\{ \frac{\sum_\omega S(\omega, x)\bar{W}_k(\omega)}{B^t} \right\} \tag{8.14}$$

where

$$B^t = \sum_\omega \bar{W}_k(\omega) \sum_{i=1}^{q} \bar{W}_i(\omega)\bar{H}_i(x)^t + \sigma^2 \sum_{i=1}^{q} \delta_{ik}\alpha_i \bar{H}_i(x)^t \tag{8.15}$$

with the unity constraint:

$$\bar{H}_k(x)^{t+1} \leftarrow \frac{\bar{H}_k(x)^{t+1}}{\sum_{i=1}^{q} \bar{H}_i(x)^{t+1}} \tag{8.16}$$

$$\bar{W}_k(\omega)^{t+1} = \bar{W}_k(\omega)^t \frac{\sum_x S(\omega, x)\bar{H}_k(x)}{\sum_{i=1}^{q} \bar{W}_i(\omega) \sum_x \bar{H}_i(x)\bar{H}_k(x)} \tag{8.17}$$

The computation for the noise variance $\sigma^2$ and the model selectors $\alpha_k$ can be performed as follows:

$$\sigma^2 = \frac{1}{V_x V_\omega} \sum_x \sum_\omega \left| S(\omega, x) - \sum_{k=1}^{q} \bar{W}_k(\omega)\bar{H}_k(x) \right|^2 \tag{8.18}$$

and

$$\alpha_k = \frac{V_x}{\sum_x \bar{H}_k(x)} \tag{8.19}$$

where $V_x$ and $V_\omega$ are the number of pixel locations and frequencies respectively. The idea, here, was to show the manner the factors are estimated avoiding masking the procedure with extra mathematical formalisms.

## 8.5 Algorithm and Examples

Two sequences of images are used in order to show the performance of the algorithm. The first one is a synthetic generated sequence with the purpose of studying the manner the algorithm deals with occlusions. Three moving objects are in the scene: two of them evolving according to a translational motion, and a third one according to a zoom operation. The three moving objects have different frequencies. Figure 8.3 shows 25 frames of a 100 frames sequence - in 1 out of 4 order-. Optical flow-based techniques are often used to estimate the motion of objects in image sequences. The main weakness of those techniques is their reliance on texture. In this specific sequence, we selected the objects to have no texture. Flow-based techniques, here, can only rely on the objects' edges since the gradient is zero almost everywhere. Moreover, this sequence contains occlusions between two objects. This is an important point to be taken into account, since a global approach is necessary to distinguish the different motions. Local approaches, here, fail when trying to distinguish different objects, since they will offer many interpretations, such as: "just one object which is stretching", instead of "two objects crossing one in front of the other". A global approach means, here, the fact that all the frames in the sequence are considered in order to estimate motion and segment objects. A proper study of the different trajectories across time yields a reliable manner of labelling the different pixels in the scene according to the different objects that are present in the sequence.

A second sequence consists of natural images with two main periodic motions: a moving face with lower frequency than a moving hand. In this case, occlusion is also a notable factor to be concerned when tackling this problem with local approaches. Moreover, in this sequence, there is a third issue to be analyzed: "non-rigid objects". We can consider the hand to be a non-rigid object, or, more properly, an articulated object - for the purpose it does not matter -. The fact is that, techniques based on parametric motion estimation lack of enough flexibility to deal with the segmentation of this moving hand (see fig. 8.2). Parametric techniques are either too restrictive or not enough general to be applied in a variety of situations. The technique we present is able, instead, to deal with non-rigid objects with the same approach applied in the first sequence.

In both sequences, few first compute the time Fourier transform for each pixel location $x$, and therefore, their corresponding power spectra. After, this first step, its necessary to assign an initial guess to the number of different periodic motions, which are supposed to be in the scene. It is recommendable to assume that there are many

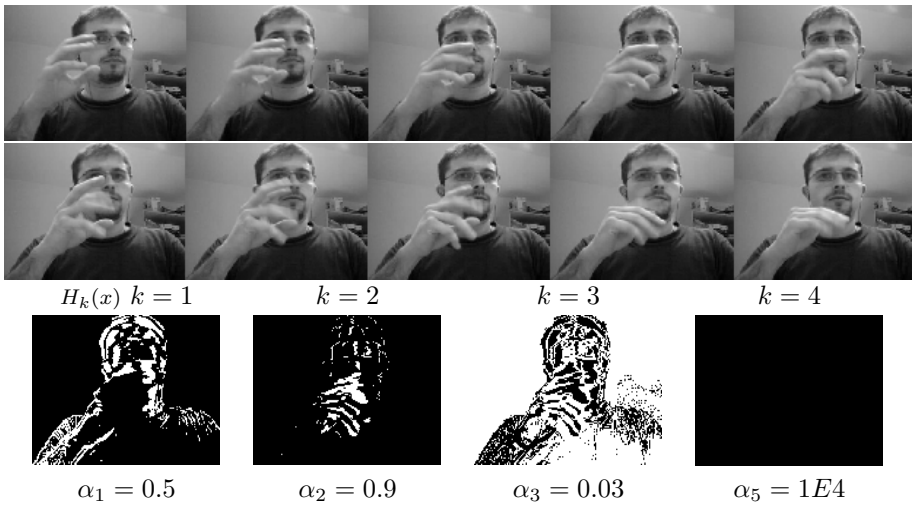



**Figure 8.2:** Top rows: Some frames (one out of five) of the face sequence. Bottom row: Components indicating the contribution of each type of periodic movement to each pixel location $x$. A large value of $\alpha_k$ indicates negligible component.

motions in the scene, since the Bayesian approach of this algorithm will explain how many true different motions are in the scene. For the synthetic sequence we chose 5 different motions as initial guess. In the other one we select 4 as initial guess. Having the power spectra for each sequence, we run the estimation process described in the previous section. After convergence, the hyper parameters $\alpha_k$ will explain in each case the number of sufficient different motions to be considered as shown in figures 8.3 and 8.2.

The algorithm provides a "location mask" $H_k(x)$ for each different detected motion. Since for each pixel location $\bar{H}_k(x)$ is normalized to the unity -eq. (8.16)- with respect to the motion model components $k$, the $\bar{H}_k(x)$ values indicate the contribution in terms of probabilities of each single segmented motion in each pixel location. This will allow labeling the different regions in the image frame in terms of the motions that occurred across the sequence. The algorithm also provides the power spectra $W_k(\omega)$ for each different detected movement. Using the components separately, we can generate synthetic video sequences with the different segmented moving objects [73]. This segmentation is performed in space and time at the same time.

## 8.6 Conclusions

We have presented a technique that classifies the different periodic motions that can be present in a video sequence. We have firstly built a model in order to show the different effects that contribute to the temporal Fourier spectra, such as: shape, motion and frame-rate. Moreover, we have shown a reliability analysis that justifies the power

spectra factorization, which is the key point for a spectral classification of the different pixels in an image sequence. This analysis permits dealing with occlusions, non-rigid objects and quasi-periodic moving backgrounds such as video textures (waterfalls, smoke, sea, waves etc.).

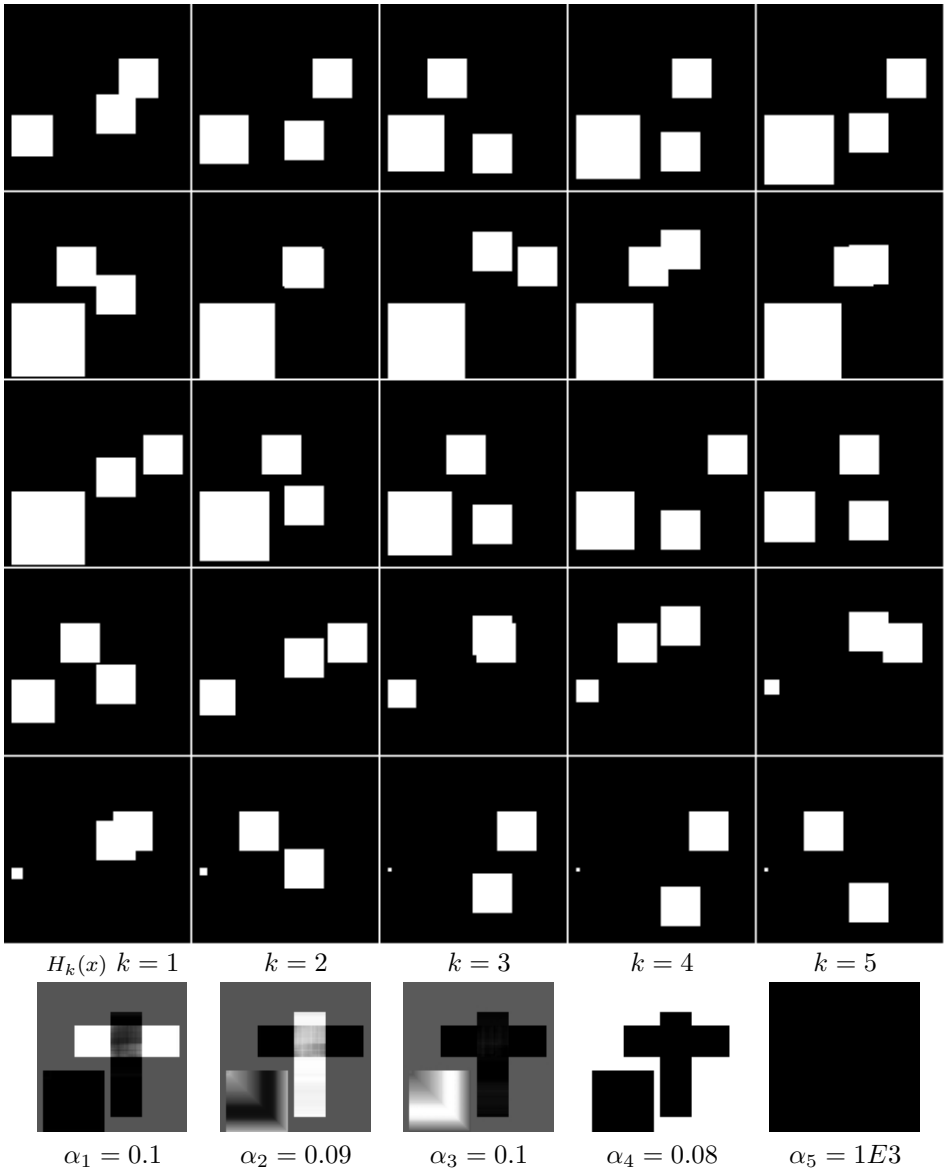



**Figure 8.3:** Top rows: Some frames (one out of four) of the synthetic sequence. Bottom row: Components indicating the contribution of each type of periodic movement to each pixel location $x$. A large value of $\alpha_k$ indicates negligible component.

# Chapter 9

# Concluding Remarks and Future Work

We conclude the thesis with some comments about our general approach, a discussion of the contributions of the thesis and a suggestion for future work.

## 9.1 Conclusions

The contributions of this thesis consist in a set of publications of new algorithms applied to two different areas in Computer Vision: i) video analysis and summarization, and ii) 3D range data. Both areas have been approached through the Latent Variables framework, where similar protocols for representing data have been employed.

**Probabilistic Approach.** The applications introduced in this thesis are based on a probabilistic formulation in order to model noise and including the *knowledge of the domain* in terms of *a priori information* through prior distributions. In addition to this, a standard methodology for estimating the parameters of the model is carried out thanks to a probabilistic formulation of the presented problems by means of EM algorithm.

**Intrinsic Degrees of Freedom, Invariance and Symmetries.** All the presented problems have in common the way how these three concepts are combined playing a fundamental role in the latent variable framework. From a 3D cylindrical geometry problem, where intuition on symmetries arises significantly straightforward, to video summarization problems, the same formulation has been applied for modelling the underlying phenomena from observations. The connection between Lie's group theory, the structural equation and the internal symmetries present in the observed data has been also pointed out.

**Local and Global Information.** The way the different levels of information can be combined from a set of observations relies on the feature selection process. The degree of optimality and reliability of the potential solutions for a given problem mainly depend on this combination of information. Moreover, from the presented applications, we can conclude that the feature selection process is guided by the constraints of each specific problem, such as non-rigidity, spatio-temporal smoothness, continuity, etc...

**Prediction.** The process that reduces dimensionality tanking into account te internal symmetries of a problem, provides a manner of dealing with missing data and it makes possible predicting new observations.

The discussed ideas (global and local information, mixtures of linear models and representation) are exemplified through different applications, which have been collected into the following topics: *Dimensionality Reduction* and *Video Analysis and Summarization*.

### 9.1.1 Dimensionality Reduction

This first part begins with an explanation of the different types of dimensionality reduction. It goes from applications based on simple linear models to problems that require a representation through non-linear models. More specifically, these latter problems are tackled by means of Mixture Models and the Divide-and-Conquer idea.

As a first step to introduce the potential applications of latent variable models in Computer Vision, this part starts with one section dedicated to medical image analysis using PCA under a probabilistic approach, which permits dealing with a measure of similarity for detecting vessel structures in angiographies.

**Detecting elongated structures using statistical snakes.**

A priori global information on shape (deformable models) is combined with local information in the image framework. In this case, the algorithm learns the gray level profile of a vessel in the perpendicular direction to its elongated structure. Thanks to the likelihood measure provided by the generative model, it is possible building a likelihood map that indicates which pixels are potential parts of a vessel structure. This map is used as a potential field for a deformable snake model in order to track the ridge of a vessel. The challenge offered by this sort of images lies in the significant level of impulsive noise due to x-rays. In addition to this, local variation of brightness and contrast make the detection problem a tough task. This chapter can be seen as an example of the combination of global and local information, in both pixel vicinity and distribution senses. The concept of basic units as relevant information extraction is also pointed out.

### 3$D$ Range Data

In order to approximate the idea of mixture models to our 3-dimensional intuition, one section is dedicated to the analysis of 3D range data and its applications to archaeology pot reconstruction. The motivation here is to avoid the standard local triangle-based techniques, since neighboring-based operations are highly expensive in terms of computational cost. In fact, local approaches are either based on computations that require a previous data ordering -triangular meshes- [61, 49, 36] or *boundary following*-like algorithms [74]. Techniques based on local computations -such as partial differential equations- suffer from extreme sensitiveness to noise. We propose new methods that are based on global computations, and, which : *i)* is fast in very large data sets, *ii)* is robust to noise, and *iii)* does not need data to be organized.

1. **Reconstruction of 3D axially symmetric surfaces from partial data.**

   Given a 3$D$ data set which represents a partial surface patch of a larger axially symmetric object, we attempt to estimate the axis of symmetry and the associated profile curve. Prior assumptions on the object's shape, constrain the presented model in terms of global information. It is assumed that data has been generated through the evolution of a profile curve around an axis of symmetry. The noise model is the result of considering a model that locally approximates the surface through cylinders, which are globally constrained by the axis of symmetry, while, at the same time, data has been corrupted during the scanning process.

2. **Finding breaking curves in 3D surfaces.**

   Particularly, the chapter dedicated to finding breaking curves in 3D open surfaces a new algorithm for Mixture Modelling, which is **initialization independent**. The manner it is formulated avoids implicitly ill-conditioning problems without being forced to add ad hoc numerical treatments. Moreover, the number of mixture components is **online automatically determined** from data, i.e., the criteria for deciding how many planes are sufficient when approximating a surface is developed in terms of the noise level in the observed data. Of course, the introduced techniques can be applied to higher dimensional data sets.

## 9.1.2 Video Analysis and Summarization

The addition of **time** to visual information analysis presents new constraints -a huge amount of information to be dealt with and specific demands (such as real-time analysis) on the formulation of feasible and reliable techniques.

The applications introduced in this thesis are focussed on dealing with: i) a much larger amount of data (sometimes batch algorithms may not be appropriate), ii) searches in a continuous media, iii) the extraction of higher level structures such as scenes, stories and pieces of news, and, iv) making feasible a quick intuition of the contents of a video under low streaming cost.

Concerning the video summarization framework, two main approaches are considered in this part: key-frame selection and video mosaicing.

### Video Mosaicing

Algorithms for image mosaicing consist of two main steps: registration, i.e. estimating the transformation that occurred across consecutive frames in the sequence, and mosaic construction, which implies utilizing the previously estimated transformations in conjunction with the images to be summarized. These two steps are intrinsically related. A good performance of the resulting final mosaic is strongly dependent on the variety of techniques, which are applied in both steps.

In our combination of global-local information approach, this subject is significantly appropriate. In this sense, space and time continuity of the different motion layers are a strong cue. Two contributions are presented in this area:

1. **An appearance-based method for video registration.**

   We frame the multiple-image registration in a two-step iterative algorithm, where one step takes a global representation for the image data set, and the other one refers to locally spatial distributed information. We have address the problem of characterizing the different types of motions that occur across a sequence based on a visual appearance information criterion and, at the same time, conjugating local and global representations. Linear subspace constraints have been based on the assumption of constancy in the appearance subspace. One of the main contributions of the appearance subspace encoding is that the appropriate scale in each problem is captured from the images themselves. Image spatio-temporal derivatives are computed by coupling linear combinations of the PC basis. The choice of an appropriate representation for the data becomes significant when dealing with image transformations, since these usually imply that the number of intrinsic degrees of freedom in the data distribution is lower than the coordinates used to represent it. This fact, not only allows embedding the video registration in a more numerical tractable framework, but also yields a new approach to extracting underlying information from temporal evolution of sequences. In this case, the noise model connects the global coordinates in the subspace representation and the parametric optical flow estimates.

2. **A polynomial fiber description of motion for video mosaicing.**

   We present a new technique based on the fact that each pixel in the frame of reference produces a trajectory in the mosaic absolute coordinate system. The model that we present describes the different layer evolutions in a sequence of images uses the results of a multi-frame optical flow estimation. Clustering is based on the fact that similar trajectories will correspond to the same sort of motion (and camera operation). Thus, we introduce a description of these paths in terms of polynomial fibers, and a probabilistic model is developed in order to rely on a measure of similarity as well as to have a classification mechanism which extracts the possible different classes of motions.

### Key-Frame Selection

This thesis presents two contributions related to key-frame selection and online piecewise video partitioning and hyper linking:

1. **An EM algorithm for video summarization through iconic image-like data structures.** Our purpose is to present a compact and perceptually meaningful representation that preserves the subjective approach, i.e. the semantics, given by actions and camera operations in the evolution of a video sequence. The model to extract this new set of iconic representative image -like data structures is based on an application of Linear Dynamical System and Lie's group theories, which are our support to define temporal symmetries and invariance. In this framework, the temporal information is encoded in an infinitesimal generator matrix, which defines different types of behaviors in the evolution of an image sequence. We use this distinct sort of contributions to give, in addition, a grouping inside the summarized representation.

2. **Online Bayesian video summarization and linking.** As an alternative to standard feature-based key-frames selection, in this chapter we propose a Bayesian framework for video summarization. We address the problem of characterizing key-frames basing partitions on appearance visual information criterion. This fact, not only allows embedding in a more numerical tractable framework the video *retrieval*, but also yields a new approach to extract underlying information from temporal evolution of sequences. We present a novel algorithm that provides an online treatment of video analysis plus the advantages of working under a Bayesian appearance-based framework. We address the problems of key-frame extraction and shot partitioning relying on a feature space where not only pixel value distributions (gray-scale or color) are encoded but also shape information is taken into account. The algorithm online classifies the different shots of a video sequence and automatically extracts the most significant key-frames. Often, due to postproduction work (in commercials, movies, etc.), there are many sequences that contain the same shot in different time positions, that make standard algorithms to produce repeated key-frames and forcing posterior ad hoc merging/removing techniques in order to avoid unnecessary redundancies. Given that the algorithm is embedded in a Bayesian formulation, questions such as sufficient number of key-frames to represent a video sequences or avoiding extra key-frame detection due to flashes, are automatically solved.

### Periodic Motion Detection

Apart from affine/projective transformations there is another relevant type of motion; periodic motion, which is often suitable for segmenting objects in video sequences. Periodic motion detection is a strong cue for object and action recognition in human motion perception. In this case the contribution is:

1. **Analyzing periodic motions in video sequences.**

   We present a novel technique to deal with a new and interesting problem, which can be stated as follows: How many different types of periodic movements are in a specific scene? Is it possible segmenting different objects from their motion when: a) there are occlusions in the scene across time and b) the same object has disconnected parts? Both questions have an answer when studying the global behavior of a sequence that contains different objects moving with different periodic movements. The algorithm we propose yields a manner for detecting in each frame : i) which pixels correspond to a specific object? and ii) which are the fundamental frequencies that contribute to its motion? The referenced works were about detecting periodicities and segmenting a particular region where periodic movements occur, however no classification for different periodic movements in the same scene was proposed.

Concerning the Generative Modelling methodology, several interesting issues are studied in each contribution; from the use of constraints on the model parameters and the use of lower bound functions, to the construction of a noise model that suitably connects different scales of information extracted from data. Moreover, the use of Mixtures of Linear Models is also pointed out. In some cases, constraints on the model are directly related to some assumptions on shape. In other cases, continuity/constancy hypotheses (in space, brightness or time, etc) are employed as prior information from external knowledge sources. This is possible thanks to the probabilistic framework that embeds this formulation.

## 9.2 Future Work

The lines of research of this thesis are directed to the fusion of multiple data sources, with straightforward applications within the MPEG 4 framework: 3D Television, web 3D, and the corresponding 3D databases.

To this end, the role of extracting descriptors of 3D pieces according to their symmetries is considered significantly relevant in this thesis, since compression and a posterior fast retrieval are possible. It is a matter of future work dealing with the different types of features that can be extracted from the Divide-and-Conquer algorithm presented in chapter 3 in order to characterize and classify 3D fragments from unorganized data sets. The main advantage is that the algorithm captures the complexity of the distribution of points in terms of curvatures, it is very fast, robust to noise and independent from initialization. For this reason, the research on algorithm that extracts a signature from the distribution of curvatures and symmetries is a very relevant issue for posterior storage and retrieval in 3D databases.

Following the same direction, the coefficients obtained for describing the polynomial fibers in chapter 5 can provide 3D information from a video sequence. This connection between the polynomial parameters and the 3D scene would give manner of simultaneously segmenting and locating in the 3D world moving objects due to

parallax.

The research on multi-parametric Lie groups for video and 3D range data is a significant issue as well. The idea of extending the one-parameter group of continuous transformations to a mixture model would permit analyzing symmetries at a local level while capturing global non-linear behaviors of the transformations. Compression of complex 3D pieces and multiple motions in video sequences would be carried out through a local symmetry analysis.

Regarding symmetries, another interesting line of research is studying Latent Spaces for periodic observations. This is desirable because modelling a periodic observed variable with a non-periodic latent variable results in a discontinuous dimensionality reduction mapping. The relation of the symmetry analysis through Lie's group theory and multiple-periodic motion would provide a efficient representation for compressing video textures with quasi-periodic motions plus camera operations and moving objects.

The ideas introduced in chapter 7 can be extended through a deeper analysis of the resulting graphs in the latent space. Detecting periodicities and closed subgraphs would automatically determine the structure of a video sequence for posterior classification purposes, i.e., pieces of news, sport events, etc.

# Appendix A

# Proof of Convergence of Variational EM Algorithm

This appendix shows the use of the log-concavity requirement for the log-likelihood in the EM algorithm. Assuming that the log-likelihood $p(D|\theta)$ of a data set $D$ has at least one finite maximum value, there is only a remaining question:

$$\mathcal{F}(Q^{k+1},\theta^{k+1}) \geq \mathcal{F}(Q^{k+1},\theta^{k}) \geq \mathcal{F}(Q^{k},\theta^{k})?$$

in other words, is $\mathcal{F}(Q^{k+1},\theta)$ monotonically increasing at each step? To answer this question, log-concavity requirement on the log-likelihood $\log p(D|\theta)$ must be employed.

1. $\mathcal{F}(Q^{k+1},\theta^{k}) \geq \mathcal{F}(Q^{k},\theta^{k})$: Assume that $Q^{k}(x) \approx Q^{k+1}(x) + \delta G(x)$

$$\begin{aligned}
\mathcal{F}(Q^{k+1},\theta^{k}) - \mathcal{F}(Q^{k},\theta^{k}) &= \mathcal{F}(Q^{k+1},\theta^{k}) - \mathcal{F}(Q^{k+1}+\delta G(x),\theta^{k}) = \\
&= \mathcal{F}(Q^{k+1},\theta^{k}) - \mathcal{F}(Q^{k+1},\theta^{k}) - \delta G(x)\left[\frac{\delta \mathcal{F}}{\delta Q(x)}\right]_{Q^{k+1}} - \\
&- \frac{1}{2}(\delta G(x))^2\left[\frac{\delta^2 \mathcal{F}}{\delta Q(x)^2}\right]_{Q^{k+1}} = \\
&= -\frac{1}{2}(\delta G(x))^2\left[\frac{\delta^2 \mathcal{F}}{\delta Q(x)^2}\right]_{Q^{k+1}}
\end{aligned}$$

The problem is focussed on computing the second functional derivative of $\mathcal{F}(Q,\theta^{k})$:

$$\begin{aligned}
\left[\frac{\delta^2 \mathcal{F}}{\delta Q(x')^2}\right] &= \frac{\delta^2}{\delta Q(x')}\left\{\int [dx] Q(x)\log\frac{P(x,D\mid\theta^{k})}{Q(x)}\right\} = \\
&= \frac{\delta^2}{\delta Q(x')}\left\{\int [dx] Q(x)\log P(x,D\mid\theta^{k}) - Q(x)\log Q(x)\right\} =
\end{aligned}$$

$$= -\frac{1}{Q(x')} < 0$$
$$\text{since } Q(x') \geq 0 \forall x'$$

Therefore $\mathcal{F}(Q^{k+1}, \theta^k) \geq \mathcal{F}(Q^k, \theta^k)$.

2. $\mathcal{F}(Q^{k+1}, \theta^{k+1}) \geq \mathcal{F}(Q^{k+1}, \theta^k)$: Assume that $\theta^k = \theta^{k+1} + \delta\theta$

$$\begin{aligned}
\mathcal{F}(Q^{k+1}, \theta^{k+1}) - \mathcal{F}(Q^{k+1}, \theta^k) &= \mathcal{F}(Q^{k+1}, \theta^{k+1}) - \mathcal{F}(Q^{k+1}, \theta^{k+1} + \delta\theta) = \\
&= \mathcal{F}(Q^{k+1}, \theta^{k+1}) - \mathcal{F}(Q^{k+1}, \theta^{k+1}) - \delta\theta \left[\frac{\partial \mathcal{F}}{\partial \theta}\right]_{\theta^{k+1}} - \\
&- \frac{1}{2}(\delta\theta)^2 \left[\frac{\partial^2 \mathcal{F}}{\partial \theta^2}\right]_{\theta^{k+1}} = \\
&= -\frac{1}{2}(\delta\theta)^2 \left[\frac{\partial^2 \mathcal{F}}{\partial \theta^2}\right]_{\theta^{k+1}}
\end{aligned}$$

$$\begin{aligned}
\left[\frac{\partial^2 \mathcal{F}}{\partial \theta^2}\right]_{\theta^{k+1}} &= \frac{\partial^2}{\partial \theta^2}\left\{\int dx Q(x) \log \frac{P(x, D \mid \theta)}{Q(x)}\right\} = \\
&= \int dx Q(x) \frac{\partial^2 \log P(x, D \mid \theta)}{\partial \theta^2}
\end{aligned}$$

Since $\log P(x, D \mid \theta)$ is concave :

$$\frac{\partial^2 \log P(x, D \mid \theta)}{\partial \theta^2} \leq 0$$

and $Q(x) \geq 0$

Therefore:

$$\mathcal{F}(Q^{k+1}, \theta^{k+1}) \geq \mathcal{F}(Q^{k+1}, \theta^k) \geq \mathcal{F}(Q^k, \theta^k)$$

It can be shown that for $\log P(x, D \mid \theta)$ corresponding to quadratic forms the convergence goes as an exponential decay function of time (number of iterations).

# Appendix B

# Publications

The contribution of this thesis consists in a set of publications of new algorithms applied to two different areas in Computer Vision: i) video analysis and summarization, and ii) 3D range data. Both areas have been approached through the Latent Variable framework, where similar protocols for representing data have been employed.

As a first step to introduce the potential applications of latent variable models in Computer Vision, the following publications introduce and exploit the use of PCA and mixtures of PCAs.

- X.Orriols. **Models Locals Lineals per a l'Anàlisi d'Imatges**. *CVC Technical Report 31, 1999*
- X. Orriols and X. Binefa. **Local Linear Models for Image Analysis**. *2on Seminari de Treball en Automàtica, Robòtica i Percepció. Octubre,1999.*
- Ricardo Toledo, Xavier Orriols, Petia Radeva, Xavier Binefa, Jordi Vitrià, Juan J. Villanueva. **Eigensnakes for Vessel Segmentation in Angiography**. *Proc. Intl. Conf. on Pattern Recognition (ICPR'2000). Barcelona, Spain, September 2000.*
- X. Orriols, R. Toledo, X. Binefa, P. Radeva, J. Vitrià, J.J. Villanueva. **Probabilistic Saliency Approach for Elongated Structure Detection using Deformable Models**. *Proc. Intl. Conf. on Pattern Recognition (ICPR'2000). Barcelona, Spain*
- R. Toledo, X. Orriols, X. Binefa, P. Radeva, J. Vitrià, J. Villanueva. **Tracking Elongated Structures using Statistical Snakes**. *In Computer Vision and Pattern Recognition, 2000*

Novel algorithms and applications of latent variable models to archaeology pot reconstruction through 3D range data can be found:

- X.Orriols,A. Willis, X. Binefa, D. Cooper. **Bayesian Estimation of Axial Symmetries from Partial Data, a Generative Model Approach**. *CVC Tech. Rep. 49 November 2000.*

- X.Orriols, X. Binefa. **Finding Breaking Curves in 3D Surfaces**. *1st Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA2003), June 2003, Mallorca, Spain*

- Andrew Willis , Xavier Orriols , Senem Velipasalar, David B. Cooper, Xavier Binefa. **Extracting Axially Symmetric Geometry From Limited 3D Range Data**. *LEMS Technical Report 192, December 2000.*

- David B. Cooper, Andrew Willis, Stuart Andrews, Jill Baker, Yan Cao, Dongjin Han, Kongbin Kang, Weixin Kong, Frederic F. Leymarie, Xavier Orriols, Eileen L. Vote, Martha S. Joukowsky, Benjamin B. Kimia, David H. Laidlaw, David Mumford, Senem Velipasalar. **Assembling Virtual Pots from 3D Measurements of their Fragments**. *Virtual Reality, Archaeology and Cultural Heritage Symposium, VAST 2001, Athens, Greece.*

- David B. Cooper, Andrew Willis, Stuart Andrews, Jill Baker, Yan Cao, Dongjin Han, Kongbin Kang, Weixin Kong, Frederic F. Leymarie, Xavier Orriols, Eileen L. Vote, Martha S. Joukowsky, Benjamin B. Kimia, David H. Laidlaw, David Mumford, Senem Velipasalar. **Bayesian Virtual Pot-Assembly from Fragments as Problems in Perceptual-Grouping and Geometric-Learning**. *Proc. Intl. Conf. on Pattern Recognition (ICPR'2002). Québec City, Canada.*

- Andrew Willis, Xavier Orriols, David B. Cooper. **Accurately Estimating Sherd 3D Surface Geometry with Application to Pot Reconstruction**. *CVPR Workshop on applications of Computer Vision in Archaeology (ACVA'03), Wisconsin, USA, June 2003*

Concerning the video analysis and summarization framework, two main approaches are considered in this part: key-frame selection and video mosaicing.

- X.Orriols, Ll. Barceló, X. Binefa. **Polynomial Fiber Description of Motion for Video Mosaicing**. *Int. Conf on Image Processing (ICIP 2001). Thessalonica, 2001*

- X. Orriols, X. Binefa. **An EM Algorithm for Video Summarization, Generative Model Approach**. *Int. Conference on Computer Vision (ICCV 2001), Vancouver, 2001*

- X.Orriols, Ll. Barceló, X. Binefa. **An Appearance-Based Method for Parametric Video Registration**. P*remières Rencontres des Sciences et Technologies de l'Information (ASTI 2001). Paris, April 2001*

- X.Orriols, X. Binefa. **Online Bayesian Video Summarization and Linking**. *International Conference on Image and Video Retrieval (CIVR 2002) July 18-19, 2002, The Brunei Gallery, SOAS, Russell Square, London, UK*

- X.Orriols, X. Binefa. **Analyzing Periodic Motion Classification**. *1st Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA2003), June 2003, Mallorca, Spain*
- X.Orriols, X. Binefa. **Classifying Periodic Motions in Video Sequences**. *International Conference on Image Processing 2003 (ICIP2003), September 2003, Barcelona, Spain*

# Bibliography


[1] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis.* Wiley, 1958.

[2] Farshid Arman, Arding Hsu, and Ming-Yee Chiu. Feature management for large video databases. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 2–12, 1993.

[3] Farshid Arman, Arding Hsu, and Ming-Yee Chiu. Image processing on compressed data for large video databases. In *Computer Graphics (Multimedia '93 Proceedings)*, pages 267–272. Addison-Wesley, 1993.

[4] H. Attias. Independent factor analysis. *Neural Computation*, 11(4):803–851, 1999.

[5] S. Ayer and H. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and mdl encoding. *ICCV*, pages 777–784, 1995.

[6] S. Ayer, P. Schroeter, and J. Bigun. Segmentation of moving objects by robust motion parameter estimation over multiple frames. In *ECCV94*, pages 316–327, 1994.

[7] David J. Bartholomew. *Latent Variable Models and Factor Analysis.* Charles Griffin & Company Ltd., London, 1987.

[8] Richard Bellman. *Adaptive Control Processes: A Guided Tour.* Princeton University Press, Princeton, 1961.

[9] J. L. Bentley, D. Haken, and J.B. Saxe. A general method for solving divide-and-conquer recurrences. *ACM SIGACT News*, 12(3):36–44, 1980.

[10] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. *ECCV*, pages 237–252, 1992.

[11] P. Bhattacharya and D. Wild. A new edge detector for gray volumetric data. *Comput. Biol. Med.*, 26:315–328, 1996.

[12] C. Bishop. Bayesian PCA. In *Advances in Neural Information Processing Systems*, volume 11, pages 382–388. MIT Press, 1999.

[13] C. M. Bishop, M. Svensen, and C. K. I. Williams. GTM: The generative topographic mapping. *Neural Computation*, 10:215–234, 1998.

[14] Christopher M. Bishop and Michael E. Tipping. Probabilistic principal component analysis. *TR NCGR*, 1997.

[15] Christopher M. Bishop and Michael E. Tipping. A hierarchical latent variable model for data visualization. *IEEE Trans. on P.A.M.I*, 20(3), 1998.

[16] C.M. Bishop and N.D. Lawrence. Variational bayesian independent component analysis. Technical report, Computer Laboratory, University of Cambridge, 2000.

[17] M.J. Black and P. Anandan. Robust dynamic motion estimation over time. *CVPR*, pages 296–302, 1991.

[18] M.J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *CVIU*, 63:75–104, 1996.

[19] M.J. Black and A. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *ECCV*, pages 329–342, 1996.

[20] R. Bolle and D. Cooper. On optimally combining pieces of information, with applications to estimating 3d complex-object position from range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(5):619–638*, 1986.

[21] J.-F. Cardoso. Source separation using higher order moments. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'89)*, pages 2109–2112, Glasgow, UK, 1989.

[22] C. Cohen, L. Conway, and D. Koditschek. Dynamical system representation, generation, and recognition of basic oscillatory motion gestures. In *Int. Conf. Auto. Face and Gesture Recognition*, volume 1, pages 60–65, 1996.

[23] P. Comon. Separation of stochastic processes. In *Proc. Workshop on Higher-Order Specral Analysis*, pages 174 – 179, Vail, Colorado, 1989.

[24] D. Cooper, A. Willis, Y. Cao, D. Han, F. Leymarie, X. Orriols, D. Mumford, et al. Assembling virtual pots from 3D measurements of their fragments. In *VAST International Symposium on Virtual Reality Archaeology and Cultural Heritage*, pages pp. 241–253, 2001.

[25] D. Cooper, A. Willis, Y. Cao, D. Han, F. Leymarie, X. Orriols, D. Mumford, et al. Bayesian pot-assembly from fragments as problems in perceptual-grouping and geometric-learning. In *ICPR*, volume III, pages pp. 297–302, August 2002.

[26] D.B. Cooper. et al. Bayesian virtual pot-assembly from fragments as problems in perceptual-grouping and geometric-learning. In *Proceedings of the International Conference on Pattern Recognition (ICPR'02), Quebec City, Canada IEEE Computer Society Press.*, 2002.

[27] J.M. Corridoni and A. Del Bimbo. Structured representation and automatic indexing of movie information contents. *Pattern Recognition*, 31:2027–2045, 1998.

[28] R. Cutler and L. Davis. Real-time periodic motion detection, analysis, and applications. In *Computer Vision and Pattern Recognition*, volume 1, pages 326–332, 1999.

[29] D.B.Cooper. et al. Assembling virtual pots from 3d measurements of their fragments. In *Proceedings of the International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST2001)*, 2001.

[30] A. Del Bimbo. *Visual Information Retrieval*. Morgan Kaufmann, 1999.

[31] A.P. Dempster, N.P. Lair, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. of the Royal Statistical Society Series B*, 39:1–38, 1977.

[32] G. Doretto, Y. Wu, and S. Soatto. Dynamic textures. *ICCV*, 2001.

[33] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.

[34] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. Wiley-Interscience, 1973.

[35] Andrew W. Fitzgibbon. Stochastic rigidity: Image registration for nowhere-static scenes. *ICCV*, 2001.

[36] M. Garcia and L. Basanez. Fast extraction of surface primitives from range images. *13th IAPR International Conference on Pattern Recognition, Vienna, Austria*, 1996.

[37] N. Goddard. The interpretation of visual motion: Recognizing moving lights. In *In IEEE Worshop on Motion*, volume 1, pages 212–220, 1989.

[38] Yihong Gong and Xin Liu. Generating optimal video summaries. In *IEEE International Conference on Multimedia and Expo (III)*, pages 1559–1562, 2000.

[39] Geoffrey E. Hinton, Peter Dayan, and Michael Revow. Modeling the manifolds of images of handwritten digits. *IEEE trans. on Neural Networks*, 8(1):65–74, 1997.

[40] Adam Hoover, Gillian Jean-Baptiste, Xiaoyi Jiang, Patrick J. Flynn, Horst Bunke, Dmitry B. Goldgof, Kevin K. Bowyer, David W. Eggert, Andrew W. Fitzgibbon, and Robert B. Fisher. An experimental comparison of range image

segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673–689, 1996.

[41] H. Hotelling. Analysis of a complex of statistical variables into principal components. *J. of Educational Psychology*, 24:417 – 441, 1933.

[42] A. Hyvärinen. Survey on independent component analysis. *Neural Computing Surveys*, 2:94–128, 1999.

[43] A. Hyvärinen. Fast ICA by a fixed-point algorithm that maximizes non-Gaussianity. In S. Roberts and R. Everson, editors, *Independent Component Analysis: Principles and Practice*, pages 71–94. Cambridge University Press, 2001.

[44] A. Hyvärinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[45] M. Irani. Multi-frame optical flow estimation using subspace constraints. *ICCV*, pages 626–633, 1999.

[46] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. *ICCV*, pages 605–611, 1995.

[47] J.Davis, A. Bobick, and W. Richards. Categorical representation and recognition of oscillatory motion patterns. In *Computer Vision and Pattern Recognition*, volume -, pages –, 2000.

[48] J.F.Canny. A computational approach to edge detection. *IEEE trans. Pattern Analisys and Machine Intell.*, 8(6):679–698, 1986.

[49] X. Jiang and H. Bunke. Edge detection in range images based on scan line approximation. *Computer Vision and Image Understanding*, 73(2):183–199, 1998.

[50] Mundy J.L. and Zisserman A:. *Geometric Invariance in Computer Vision*. The M.I.T. Press.

[51] G. Johansson. Visual motion perception. *Scientific American*, 232:75–88, 1976.

[52] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

[53] S. Ju, M.J. Black, and A. Jepson. Multilayer, locally affine optical flow, and regularization with transparency. *CVPR*, pages 307–314, 1996.

[54] K. Karhunen. Zur Spektraltheorie stochastischer Prozesse. *Ann. Acad. Sci. Fennicae*, 34, 1946.

[55] J. Kivinen and M. Warmuth. Additive versus exponentiated gradient updates for linear prediction. *Journal of Information and Computation*, 132:1–64, 1997.

[56] Jyrki Kivinen and Manfred K. Warmuth. Additive versus exponentiated gradient updates for linear prediction. In *Proceedings of the 27th Annual ACM Symposium on Theory of Computing*, New York, 209–218 1995. The Association for Computing Machinery.

[57] Jyrki Kivinen and Manfred K. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–63, 1997.

[58] N. Lawrence. *Variational Inference in Probabilistic Models*. PhD thesis, Computer Laboratory, University of Cambridge, New Museums Site, Pembroke Street, Cambridge, CB2 3QG, U.K., 2000.

[59] Daniel D. Lee and H. Sebastian Seung. A problem of dimensionality: A simple example. *Advances in Neural and Information Processing Systems*, 13:–, July 2001.

[60] T.-W. Lee, M. Girolami, and T. J. Sejnowski. Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, 11(2):417–441, 1999.

[61] André Lejeune and Frank P. Ferrie. Finding the parts of objects in range images. *Computer Vision and Image Understanding: CVIU*, 64(2):230–247, 1996.

[62] J. Little and J. Boyd. Describing motion for recognition. In *Proc. Symp. Comp. Vis. IEEE*, volume 1, pages 235–240, 1995.

[63] J. Little and J. Boyd. Recognizing people by their gate: the shape of motion. In *Videre*, volume 1, pages –, 1998.

[64] F. Liu and R. Picard. Finding periodicity in space and time. In *International Conference on Computer Vision*, volume 1, pages 376–383, 1998.

[65] M. Loève. Fonctions aléatoires du second ordre. In P. Lévy, editor, *Processus stochastiques et mouvement Brownien*, page 299. Gauthier - Villars, Paris, 1948.

[66] G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley-Interscience, 1997.

[67] Baback Moghaddam and Alex Pentland. Probabilistic visual learning for object representation. *IEEE Trans. on P.A.M.I*, 19(7):696–710, 1997.

[68] H. Murase and S.K. Nayar. Visual learning and recognition of 3d objects from appearance. *IJVC*, 14(5):5–24, 1995.

[69] R.M. Neal and G.E. Hinton. *A new view of the EM algorithm that justifies incremental and other variants*. Kluwer, 1998.

[70] S. Niyogi and E. Adelson. Analyzing and recognizing walking figures in xyt. In *Computer Vision and Pattern Recognition*, volume 1, pages 469–474, 1994.

[71] Erkki Oja. *Subspace Methods of Pattern Recognition*. Research Studies Press, Letchworth, Hertfordshire UK, 1983.

[72] X. Orriols and X. Binefa. Finding breaking curves in 3d surfaces. In *1st Iberian Conference on Pattern Recognition and Image Analysis*, Mallorca, Spain, 2003. Lecture Notes in Computer Science, Springer-Verlag.

[73] Xavier Orriols and Xavier Binefa. Reliability analisys of periodic motions classification in video sequences. In *CVC. Technical Report*, volume 132.

[74] T. Pavlidis. Algorithms for graphics and image processing. *New York:Springer*, 1982.

[75] Karl Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

[76] A. Pentland, R.W. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases. *IJCV*, 18:233–254, 1996.

[77] Alex Pentland, Baback Moghaddam, and Thad Starner. View-based and modular eigenspaces for face recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[78] R. Polana and R. Nelson. Low level recognition of human motion. In *In IEEE Worshop on Motion of Non-rigid and Articulated Objects*, volume 1, pages 77–82, 1994.

[79] R. Polana and R. Nelson. Detection and recognition of periodic, non-rigid motion. *International Journal of Computer Vision*, 23:261–282, 1997.

[80] H. Pottmann, H. Chen, and I. Lee. Approximation by profile surfaces, 1998.

[81] H. Pottmann, M. Peternell, and B. Ravani. An introduction to line geometry with applications, 1999.

[82] R.P.N Rao and D.L. Ruderman. Learning lie groups for invariant visual perception. *N.I.P.S*, 11:810–816, 1999.

[83] Jorma Rissanen. Modelling by shortest data description. *Automatica*, 14:465–471, 1978.

[84] S. Roweis. EM algorithms for PCA and SPCA. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10, pages 626 – 632. MIT Press, 1998.

[85] Sam Roweis and Zoubin Gharhamani. A unifying review of linear gaussian models. *Neural Computation*, 11(2), 1999.

[86] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 22 2000.

[87] H. Sawhney and S. Ayer. Compact representation of videos through dominant and multiple motion estimation. *IEEE Trans. on PAMI*, 18:814–829, 1996.

[88] H. Sawhney and S. Ayer. Compact representation of videos through dominant motion estimation. *IEEE PAMI*, 1996.

[89] A. Schodl, R. Szeliski, D. Salesin, and I. Essa. Video textures. *ACM SIGGRAPH*, 2000.

[90] David W. Scott. *Multivariate Density Estimation. Theory, Practice, and Visualization.* Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, New York, London, Sydney, 1992.

[91] David W. Scott and J. R. Thompson. Probability density estimation in higher dimensions. In J. E. Gentle, editor, *Computer Science and Statistics: Proceedings of the Fifteenth Symposium on the Interface*, pages 173–179, Amsterdam, New York, Oxford, 1983. North Holland-Elsevier Science Publishers.

[92] S.M. Seitz and C.R. Dyer. View-invariant analysis of cyclic motion. *International Journal of Computer Vision*, 25:1–23, 1997.

[93] H-Y. Shum and R. Szeliski. Panoramic image mosaics. *Microsoft Research TR.*

[94] Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis.* Number 26 in Monographs on Statistics and Applied Probability. Chapman & Hall, London, New York, 1986.

[95] M.A. Smith and M.G. Christel. Automating video database indexing and retrieval. *ACM Int. Conf. on Multimedia*, pages 357–358, 1995.

[96] C. Spearman. General intelligence, objectively determined and measured. *American J. of Psychology*, 15:201–293, 1904.

[97] K. Sung and Tomaso Poggio. Example-based learning for view-based human face detection. *A.I. Memo 1521, C.B.C.L. Paper 112*, 1994.

[98] R. Szeliski. Video mosaics for virtual environments. *IEEE Computer Graphics and Applications*, pages 22–30, 1996.

[99] M. Szummer and R. Picard. Temporal texture modeling. *ICIP*, 1996.

[100] Joshua B. Tenenbaum. Mapping a manifold of perceptual observations. pages 682–688.

[101] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 22 2000.

[102] L.A. Teodosio and W. Weber. Salient video stills: Content and context preserved. *ACM Int. Conf. on Multimedia*, 1993.

[103] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11, 1999.

[104] R. Toledo, X. Orriols, X. Binefa, P. Radeva, J. Vitria, and J. Villanueva. Tracking of elongated structures using statistical snakes. In *In Computer Vision and Pattern Recognition (CVPR 2000)*, pages 157–162, Hilton Head Island, USA, 2000.

[105] P. Torr. Geometric motion segmentation and model selection. *Phil. Trans.*, 1998.

[106] M. Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[107] J. Wang and E. Adelson. Layered representation for motion analysis. *CVPR*, pages 361–366, 1993.

[108] S. Watanabe. Karhunen-loeve expansion and factor analisys, theoretical remarks and applications. In *4th Prague Conference on Information Theory, statistical decission functions and random processes*, pages 645–660, 1965.

[109] Edward J. Wegman. Hyperdimensional data analysis using parallel coordinates. *J. Amer. Stat. Assoc.*, 85(411):664–675, September 1990.

[110] J. Weickert. Coherence-enhancing diffusion of colour images. In *Image and Vision Computing*, volume 17, pages 201–212, 1999.

[111] P. Whittle. On principal components and least squares method of factor analysis. *Skandinavisk Aktuarietidskrift*, 36:223–239, 1952.

[112] A. Willis, X. Orriols, S. Velipasalar, D. Cooper, and X. Binefa. Extracting axially symmetric 3d geometry from limited 3d range data. SHAPE-TR-2001-01, SHAPE Lab, Brown University, Providence, RI, 2001. http://www.lems.brown.edu/vision/publications/index.html.

[113] Orriols X., A. Willis, S. Velipasalar, D. Cooper, and X. Binefa. Bayesian estimation of axial symmetries from partial data, a generative model approach. CVC tech. rep. 49, Computer Vision Center, Universitat Autonoma de Barcelona, 2000.

[114] G. Young. Maximum likelihood estimation and factor analysis. *Psychometrika*, 6:49–53, 1940.

[115] H. Yu and W. Wolf. A visual search system for video and image databases. *Int. Conf. on Multimedia Computing and Systems*, pages 517–524, 1997.

[116] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying scene breaks, 1995.

[117] H.J. Zhang, A. Kankanhalli, and S. Smoliar. Automatic partitioning of video. *Multimedia Systems*, 1:10–28, 1993.

[118] Y.J. Zhang. Quantitative study of 3d gradient operators. *Image and Vision Computing*, 11:611–622, 1993.

[119] Wenyi Zhao, J. Wang, D. Bhat, K. Sakiewicz, N. Nandhakumar, and W. Chang. Improving color based video shot detection. In *ICMCS, Vol. 2*, pages 752–756, 1999.