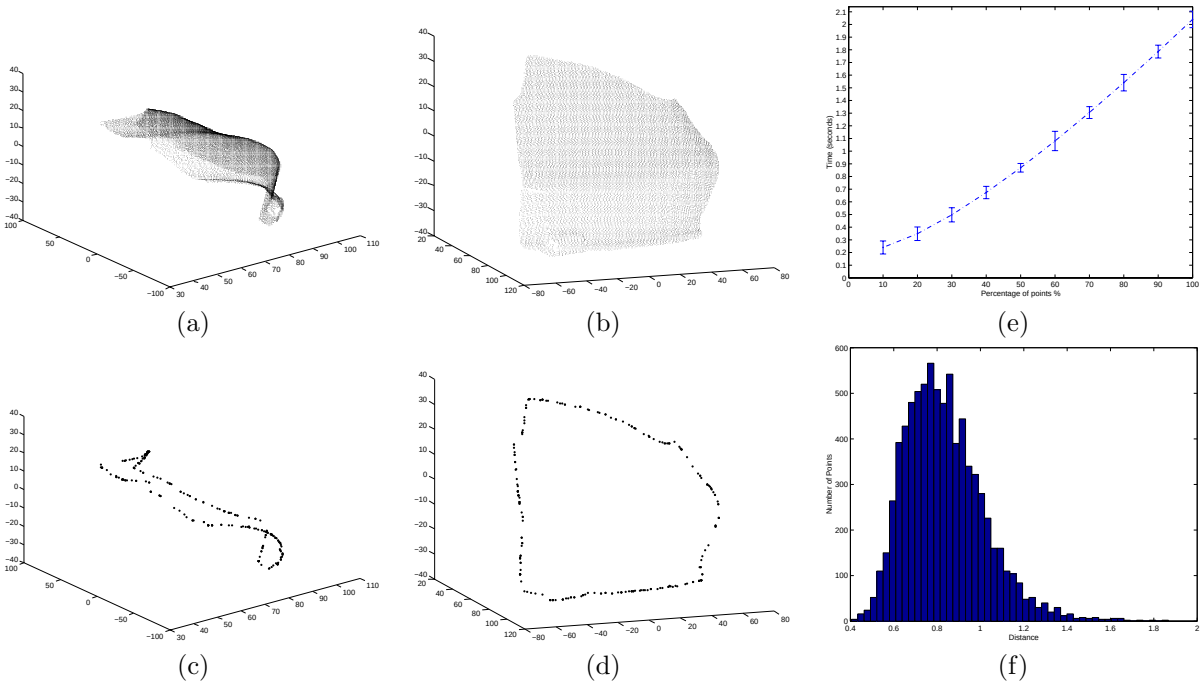


Figure 3.9: Two points of view of a range open surface in (a) and (b)where the number of points is 26436. Breaking curve points detected in (c) and (d) corresponding views. Computational cost analysis in (e). ϵ-Histogram in (f).

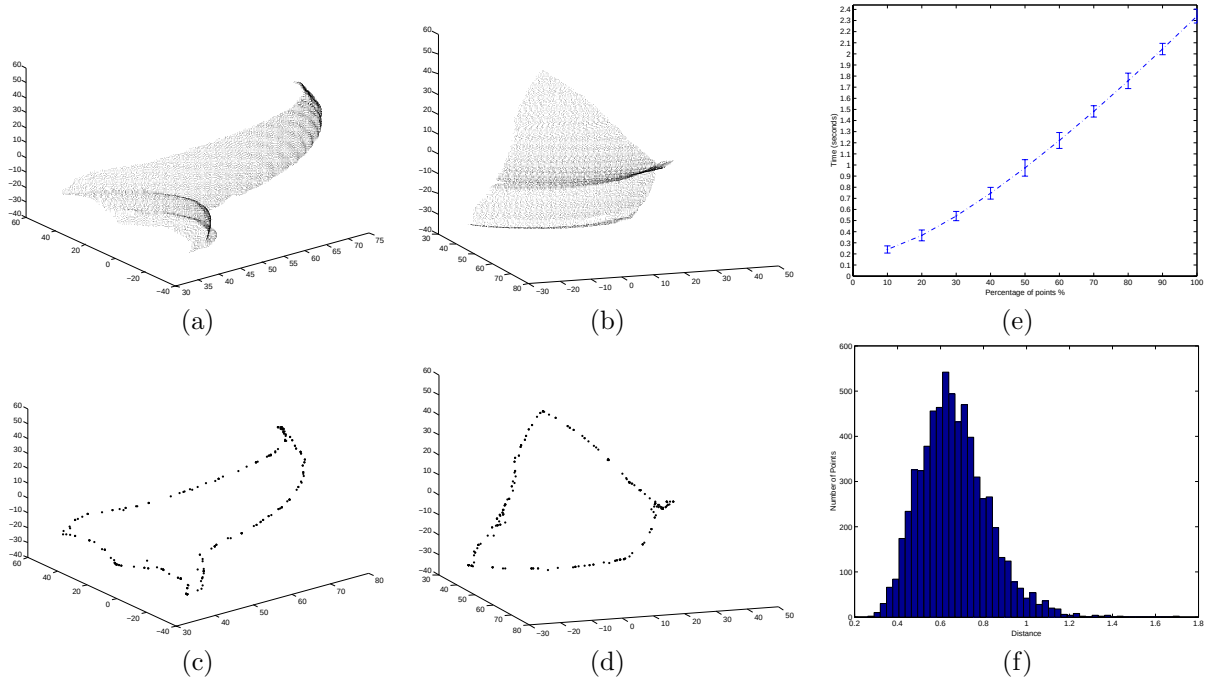


Figure 3.10: Two points of view of a range open surface in (a) and (b)where the number of points is 18495. Breaking curve points detected in (c) and (d) corresponding views. Computational cost analysis in (e). ϵ-Histogram in (f).

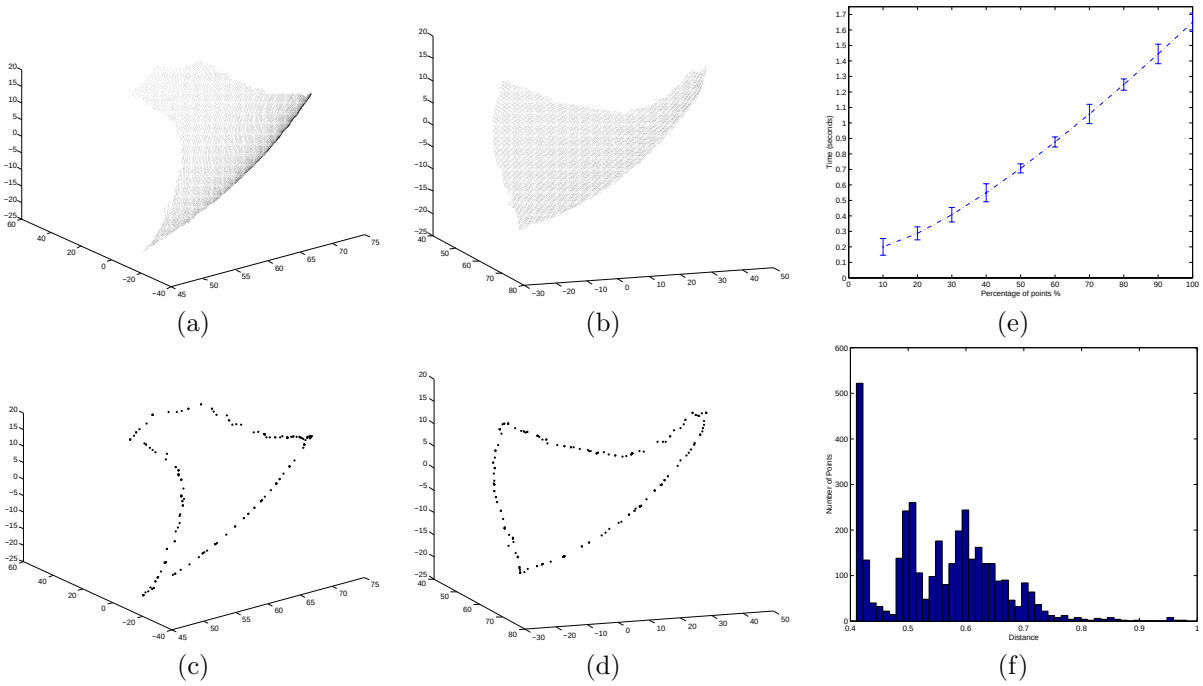


Figure 3.11: Two points of view of a range open surface in (a) and (b)where the number of points is 12072. Breaking curve points detected in (c) and (d) corresponding views. Computational cost analysis in (e). ϵ-Histogram in (f). 12072

Part II

Video Analysis and Summarization

Chapter 4

Appearance Constrained Brightness Constancy

Many sequences of images present two different sorts of motions; one is related to camera operations (panning, tilting, zooming, etc...), and the other type corresponds to movements of certain objects in the scene. In this chapter, we focus on motions that can be described as stochastic stationary processes. In nature, the latter type appear in a huge variety of scenes: the motion of the leaves on a tree when the wind blows, waterfalls, flames, etc... For these types of sequences, we will show the two main problems that arise when dealing with camera motion estimation: *i)* local spatio-temporal measurements are not sufficient, and *ii)* a certain degree of rank among the different types of pixel value variations across a sequence is necessary. We present an appearance based framework which involves both global and local information extracted from the images themselves. We show how a proper encoding for the images' appearance allows constraining the Brightness Constancy equation in order to minimize the stochastic motion contributions when estimating camera transformations.

4.1 Introduction

A sequence of images provides a mixture of different perceptual levels of visual motion information. Either the selection or rejection of a certain level of motion perception, as well as a particular combination of them, are usually guided by some specific purpose, such as camera motion estimation. These levels are not only related to spatial scales of observation but also to temporal scales. Two main levels of motion perception can be distinguished as local or global spatio-temporal information. For instance, consider a scene where a camera moving around the top of a tree on a windy day is capturing the motion of the leaves on the tree. One might

take a look at a small region of the camera's field of view (e.g. just around one leaf) without being able to explain the camera transformation across the sequence. On the other hand, one might just consider a pair of frames of the sequences, and there will also be no chance of estimating the camera motion, since there is an entanglement of different types and levels of motion; the wind making the leaves move randomly at a certain spatio-temporal scale while the camera is also moving at the same time. Both situations correspond to a local analysis of the motion perception: the first being in space and the second being in time. These can be interpreted as a spatio-temporal extension of the aperture problem.

The idea of exploring sequences with stationary stochastic motions has lead to a new fruitful and challenging area in Computer Vision: *video textures*. Modelling, recognition and synthesis for video textures have been implemented from different approaches [99, 32, 89, 35]. However, as Fitzgibbon [35] recently noticed, a significant problem arises in non-rigid stochastic scenes when it comes to dealing with camera motions -translations, rotations, etc. Image sequences subject to this condition are challenging, and moreover, complicated when approaching this problem by means of standard registration techniques. The key point is that stochastic variations can be present in a large number of pixels in each image. Techniques based on $2D$ parametric alignment between pairs of frames [107, 5, 87, 10, 53] do not capture temporal scale across the sequence evolution, therefore, non-rigidity will appear as a significant amount of noise causing disastrous effects on the final registration. Robust estimation techniques [17, 18, 5] are reliable when treating noise and moving objects as outlyers, however, the ratio of pixels affected by the stochastic process can be over the *breakdown point*. Note that some global information across time is necessary in order to be capable of decomposing the sequence into these two components (camera and stochastic) in a reliable way. This is not possible with just a pair of frames in order to determine the decomposition of their relative transformation into a camera operation and a stochastic motion. Two main problems arise when it comes to dealing with camera operation estimation and stationary stochastic processes at the same time:

- Local spatio-temporal measurements are not sufficient.
- We need to re-consider the idea of image *time derivative*, since image difference does not give a degree of significance among the different types of pixel value variations. In other words, what is the relevant information that contributes to estimate a camera transformation?

The purpose of this chapter is to point out the need for a combination of both local and global analyzes when dealing with camera motion estimation in sequences that combine stochastic stationary motions and camera operations. To this end, we present a computationally effective technique that uses the fact that images sharing the same plane geometry are also sharing appearance information under a certain camera transformation. A constraint in terms of the images' appearance is given to the optical flow constraint equation, -also known as the Brightness Constancy assumption. Previously, the authors of [19] introduced the notion of *subspace constancy assumption*, where visual prior information is exploited in order to build a

views+affine transformation model for object recognition. Their starting point was that the training set had to be carefully selected with the aim of capturing just appearance variabilities; that is to say, the training set was assumed to be absent of camera (or motion) transformations. Once the learning step was performed, the test process was based on the computation of the affine parameters and the subspace coefficients that map the region in the focus of attention onto the closest learned image. However, in this chapter, the topic that we deal with has as input data the images of a sequence that include camera (or motion) transformations. Our framework is based on the assumption that there is a time scale τ such that stochastic motions contribute less to the total appearance variation than camera transformations. In other words, when having a global *view* of the sequence, we are able to have a notion of the camera motion. Such a global *view* involves taking enough frames ($\tau <$ number of frames) in order to capture the time scale. Our results reveal that in this type of sequence, what is really important is the selection of the number of frames (time scale τ) to be analyzed rather than the amount of fluctuations in a single pair of images. From a computational point of view, it is worth noting that the optimization steps are reduced to linear least squares problems, and the solution turns out to be in a closed form for each iteration. Moreover, the framework we present allows the introduction of a Bayesian formulation for model selection.

First, the chapter presents the motivation for a proper encoding of image sequences. Subsequently, the formulation of the Brightness Constancy (BC) assumption in terms of the appearance representation is developed yielding an appearance constrained BC equation for motion estimation. Later, the formulation of a parametric model for camera transformation is presented in order to show the connection between the appearance constrained BC and parametric models. Finally, the experiments have the purpose of exposing the contributions of our approach such as: capturing a proper scale from images themselves, camera motion estimation and reliability analysis for the estimates.

4.2 Appearance Framework

4.2.1 Image Representation

A technique is necessary that distinguishes both types of variations and which, therefore, takes into account their significance in terms of amplitudes and temporal scales of observation. We can see that the problem is focused mainly on finding a proper encoding for the underlying appearance of images. To address the problem of appearance representation, the authors in [106, 77, 68] proposed Principal Component Analysis as a redundancy reduction technique during the codification process of the principal features. The idea is to express the images of a sequence in terms of an energy ordered basis, where the first components contain the information relative to camera transformation (as in fig. 4.1). Thus, the contribution of fluctuations -due to stochastic processes- is minimum during the registration process. This idea is based on the following condition:

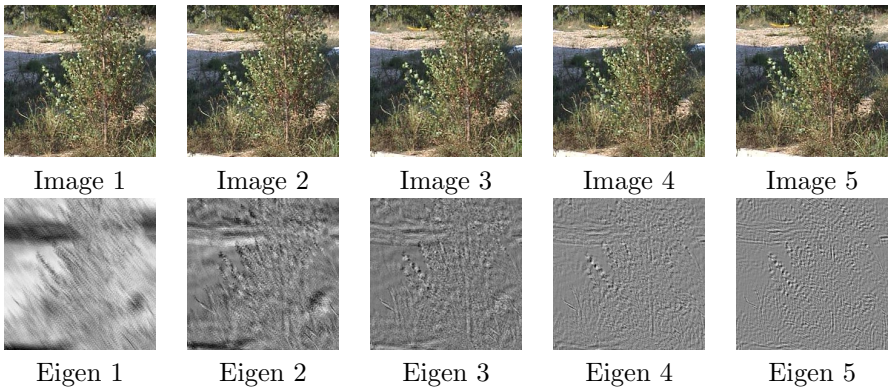


Figure 4.1: Decomposition of 5 images (top row) into an orthogonal eigen basis (bottom row) which has been ordered in terms of decreasing variance (left to right).

We deal with sequences of images that have a certain time scale τ such that their maximum appearance variation among them is due to camera transformations.

Figure 4.1 shows five images of a sequence. Even though most of the pixels in each image correspond to regions where the motion is according to a stationary stochastic process (due to the wind during the capture process), the eigen-decomposition shows that the first components tell us about the camera motion. These components are obtained by taking the Singular Value Decomposition for this set of images. Representing these images in terms of a few number of these *Principal Components* (PC), we drastically reduce the complexity added by the stochastic motion of the leaves. This new representation is performed as a linear projection $\mathcal{P}$ onto the subspace of appearance that the selected set of images *share* [106, 77, 68]. Considering an image $I(\vec{x})$ as an intensity function for a certain domain $\mathcal{D}$ of pixel locations $\vec{x} \in \mathcal{D}$, this linear projection $\mathcal{P}$ generates a response at each pixel location $\vec{x}$ as a linear combination of a reduced number q of principal components $\{W_1(\vec{x}), \ldots, W_q(\vec{x})\}$:

$$\mathcal{P} : I(\vec{x}) \rightarrow I_{\mathcal{P}}(\vec{x}) = \sum_{k=1}^{q} W_k(\vec{x}) a_k \tag{4.1}$$

where a_k are the coefficients that combine the components $W_k(\vec{x})$ in order to obtain the reconstructed intensity $I_{\mathcal{P}}(\vec{x})$ at the pixel location $\vec{x}$. By virtue of orthogonality for the PC, each coefficient a_k is computed as follows:

$$a_k = \sum_{\vec{x} \in \mathcal{D}} W_k(\vec{x}) I(\vec{x})$$

We use this topographic $\vec{x}$-dependent notation in order to be consistent with the formulation for the camera transformation estimation. Choosing the number of PC q is an issue considered in the experiments that we present; it has to be small enough to avoid the stochastic fluctuations, and, big enough to capture accurately the camera motion estimation.

4.2.2 *Projected* Brightness Constancy Assumption

Having a suitable image representation that allows ranking the different types of variations produced in a sequence, we are able to constrain the *Brightness Constancy Assumption* in terms of appearance.

Consider two images $I'(\vec{x})$ and $I(\vec{x})$ which are related by some geometrical *displacement*[1] $\mathcal{F}$ (differentiable map) between pixel positions:

$$\mathcal{F} : \vec{x} \rightarrow \vec{x}' = \vec{x} + \vec{u}(\vec{x})$$

Therefore, the intensity values of these two images are related through:

$$I'(\vec{x}) = I(\vec{x}')$$

Applying the displacement $\mathcal{F}$ between pixel locations, the latter equation is written as:

$$I'(\vec{x}) = I(\vec{x} + \vec{u}(\vec{x})) \tag{4.2}$$

When the transformation $\mathcal{F}$ can be assumed infinitesimal, equation (4.2) can be written in a first-order approximation form, which corresponds applying a brightness constancy assumption to the images transformation:

$$I(\vec{x} + \vec{u}(\vec{x})) \approx I(\vec{x}) + \vec{u}(\vec{x}).\nabla I(\vec{x}) \tag{4.3}$$

Here, the aim is to give a constraint to the equation (4.3) for a given representation of the images' appearance. To this end, we need to project the images to the subspace where stochastic fluctuations can be neglected, i.e., $\mathcal{P}$:*Image* $\rightarrow$ *Projection*,

$$\begin{aligned} I'(\vec{x}) \quad &\rightarrow \quad I'_{\mathcal{P}}(\vec{x}) = \sum_{k=1}^{q} W_k(\vec{x}) b_k \\ I(\vec{x}) \quad &\rightarrow \quad I_{\mathcal{P}}(\vec{x}) = \sum_{k=1}^{q} W_k(\vec{x}) a_k \end{aligned}$$

Combining equations (4.2) and (4.3):

$$\sum_{k=1}^{q} W_k(\vec{x}) b_k = \sum_{k=1}^{q} W_k(\vec{x}) a_k + \vec{u}(\vec{x}).\nabla \sum_{k=1}^{q} W_k(\vec{x}) a_k$$

and defining the *projected* temporal derivative and *projected* gradient as follows:

$$\begin{aligned} \frac{\partial I(\vec{x})}{\partial t} \Rightarrow \frac{\delta I(\vec{x})}{\delta t} \quad &\equiv \quad \sum_{k=1}^{q} W_k(\vec{x})(b_k - a_k) \\ \nabla I(\vec{x}) \Rightarrow \widetilde{\nabla} I(\vec{x}) \quad &\equiv \quad \sum_{k=1}^{q} a_k \nabla W_k(\vec{x}) \end{aligned}$$

[1] Also known as diffeomorphism.

therefore, we obtain the *appearance subspace* constrained brightness constancy equation:

$$\frac{\delta I(\vec{x})}{\delta t} - \vec{u}(\vec{x}).\widetilde{\nabla} I(\vec{x}) = 0 \tag{4.4}$$

We can see that these new re-definitions of temporal derivative and gradient operator act according to the information encoded in each principal component $W_k(\vec{x})$. One would prefer to capture just one specific type of variation that has occurred across the sequence evolution. Our results show that camera motion information can be obtained with a good degree of reliability when a suitable number of frames (time scale) and an appropriate number of principal components have been selected.

4.2.3 Parametric Model for Affine Camera Motion Estimation

Direct computations from image spatio-temporal derivatives have been used extensively for parametric optical flow estimation. The aim of this section is to show briefly the connection between these new suitably re-defined spatio-temporal derivatives and a parametric model for the infinitesimal displacements. This allows the chapter to be self-contained. We use the affine model as an example to estimate the motion parameters in terms of appearance subspace constraints. However, more complex models[2] can easily be used following the same method; this depends on the requirements of the scenes that are analyzed.

For affine transformations, the instantaneous motion $\vec{u}(\vec{x})$ as a function of pixel location $\vec{x} = (x, y)$ is written as:

$$\vec{u}(\vec{x}) = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{pmatrix} \tag{4.5}$$

To simplify, let us name the location matrix as $M(\vec{x})$ and the parameter vector as $\vec{\theta}$, so that the previous equation can be written in a compact form: $\vec{u}(\vec{x}) = M(\vec{x})\vec{\theta}$. Now, we apply this model to equation (4.4), and the resulting least-squares minimization functional is:

$$\mathcal{E}(\vec{\theta}) = \sum_{\vec{x}} \mid \frac{\delta I(\vec{x})}{\delta t} - \widetilde{\nabla} I(\vec{x})^T M(\vec{x})\vec{\theta} \mid^2 \tag{4.6}$$

The minimum value of $\mathcal{E}(\vec{\theta})$ w.r.t. $\vec{\theta}$ is found setting derivatives to zero $\frac{\partial \mathcal{E}(\vec{\theta})}{\partial \vec{\theta}} = 0$ and solving:

$$A\vec{\theta} = \vec{b}$$

where,

$$A \equiv \left[\sum_{\vec{x}} M(\vec{x})^T \widetilde{\nabla} I(\vec{x}) \widetilde{\nabla} I(\vec{x})^T M(\vec{x}) \right] \mid_{6\times 6} \tag{4.7}$$

[2] But not more difficult.

and,

$$\vec{b} \equiv \left[\sum_{\vec{x}} M(\vec{x})^T \widetilde{\nabla} I(\vec{x}) \frac{\delta I(\vec{x})}{\delta t} \right] |_{6\times 1}$$

The solution for $\vec{\theta}$ is found by inverting A: $\vec{\theta} = A^{-1}\vec{b}$. The matrix A and the vector $\vec{b}$ can be written in terms of these components as follows:

$$\begin{aligned} A &= \sum_{i=1}^{q}\sum_{j=1}^{q} a_i a_j \left[\sum_{\vec{x}} M(\vec{x})^T \nabla W_i(\vec{x}) \nabla W_j(\vec{x})^T M(\vec{x}) \right] \\ \vec{b} &= \sum_{i=1}^{q}\sum_{j=1}^{q} a_i (b_j - a_j) \left[\sum_{\vec{x}} M(\vec{x})^T W_j(\vec{x}) \nabla W_i(\vec{x}) \right] \end{aligned}$$

From these two equations, we realize that there are two weighting issues contributing to the parameter estimation:

- The contribution of each pixel to the error measurement is not only local, since local spatial and temporal derivatives are computed from the principal components $W_k(\vec{x})$, which encode global information. Thus, a suitable selection of the number of principal components will minimize the contribution of stationary stochastic motions in the sequence.
- The coefficients (a_i, a_j, b_j) also determine the weight for each projected image in each principal component. Each coefficient a_i tells us about a distance measurement -in its associated principal component direction- between a projected image and the subspace origin. Therefore, the larger the distance, the more it contributes to the information encoded in that specific component to the parameter estimation. On the other hand, note that the projected temporal derivative is computed by the difference between the components of each projected image, i.e., $(b_j - a_j)$. This means that even though images might differ due to stochastic motions, we are able to minimize those kind of contributions.

4.3 Experimental Results

In this section, we describe our approach to estimating affine camera motions in sequences with stationary stochastic motions. Our aim is to show the contribution of a compact representation of appearance to image registration. We first analyze the behavior of the *projected* spatio-temporal derivatives as well as their contribution to the estimation of camera transformations. The selection of a time scale (number of frames) and a specific number of principal components are also issues to be considered here. In addition, computational aspects are also commented: (i) the fact that a coarse-to-fine framework is not necessary, since the proper scale is captured by the representation in terms of Principal Components; (ii) the decomposition of this non-linear registration problem into two linear sub-problems (PCA + parameter estimation).

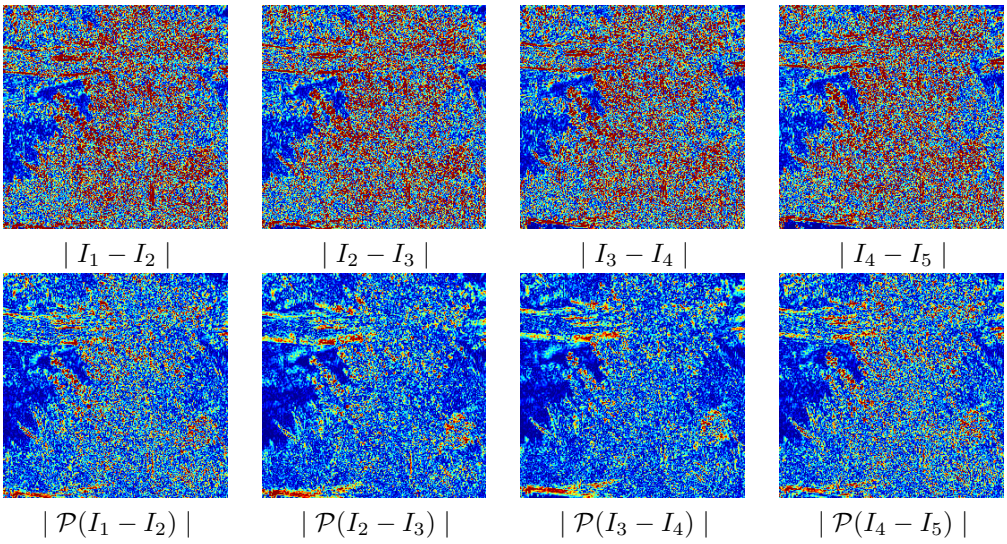
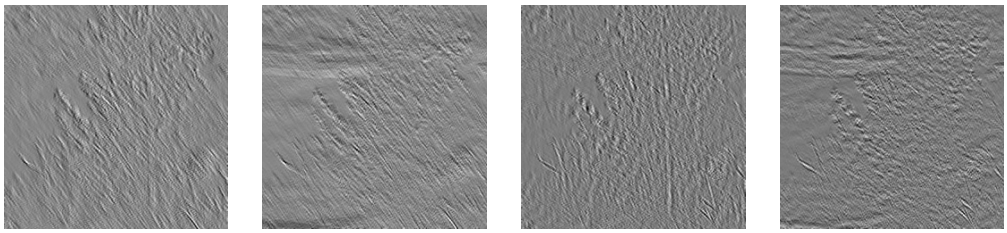


Figure 4.2: Image absolute difference between consecutive images: top row between pixel values, bottom row between coefficients in PC subspace.

4.3.1 *Projected* Spatio-Temporal Derivatives

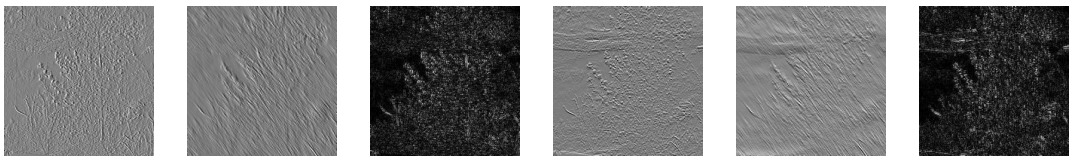
Our first experiment has the purpose of showing the contribution of a compact representation to camera transformation estimation. We consider here a scene where stationary stochastic motions are significantly present in each image transformation. With regard to figure 4.1, we can see the decomposition of five consecutive frames into an orthogonal basis that have been obtained by the PCA of the original images. Each component has an associated scalar value (variance) that gives us a notion of the significance w.r.t. the other components when reconstructing each image. The first component indicates the direction of maximum variance for the images distribution in the subspace representation, -w.r.t. the origin of the images representation space- the second component to the second maximum variance direction and so forth. We can see that the last components concentrate locally these variations. Consider the following; we select the two first PC for building the appearance subspace. Figure 4.2 shows the consequences of this selection; whereas a pixel-based image difference (top row) takes into account any pixel value variation, there is no distinction among the different levels of information that are encoded in the evolution of the sequence. However, a suitable representation of this sequence in terms of PC minimizes the contribution of higher frequency spatio-temporal terms (bottom row), since these local details were not taken into account in the two first Principal Components.

In this sense, the projected spatial derivatives are also an interesting issue. In this framework, they are built by linearly combining the appearance gradient basis $\nabla W_i(\vec{x})$ (figure 4.3). We can see the comparison between the standard gradient -in this case we have taken a $[1, -1]$ like operator- and the one performed by means of



$\partial_x W_1(\vec{x})$ $\partial_y W_1(\vec{x})$ $\partial_x W_2(\vec{x})$ $\partial_y W_2(\vec{x})$

Figure 4.3: Gradient basis for projected spatial derivatives: (two first columns) first component, (two second columns) second component.



$\nabla_x I_1(\vec{x})$ $\mathcal{P}(\nabla_x I_1(\vec{x}))$ Residue x $\nabla_y I_1(\vec{x})$ $\mathcal{P}(\nabla_y I_1(\vec{x}))$ Residue y

Figure 4.4: Gradient Vs projected gradient.

the PC basis in figure 4.4. Note that the "standard" gradient information relies on the operator that has been selected (gaussian scale-dependent filter, etc). However, in our framework, the point is that global information (PC encoding) is capturing the appropriate scale from the images themselves. This fact has a significant implication when implementing a registration algorithm, since no coarse-to-fine techniques are necessary.

4.3.2 Registration of scenes with camera + stochastic motions

We now combine a description of the algorithm and the results for a sequence of 50 frames and another of 30. We have selected an interval of 5 frames as a time scale, and 2 PC for the appearance representation for both sequences. Given that the translation dominates the camera transformation, a significant amount of occlusions in the border would appear if we took a larger interval; we need an interval that is big enough to capture the time scale for a camera motion, and small enough to avoid occlusion effects. Of course, this selection is sequence dependent, and, a more accurate approach should take into account that the partition of the sequence into intervals should be non-uniform. This is considered in the next section where we perform an analysis in terms of the reliability of the parameter estimates.

For each interval, the process is based on the following steps:

1. *Find the Principal Components $W_i(\vec{x})$ of the images through SVD of the images as in [106, 77, 68].*
2. *Estimate the transformation parameters $\vec{\theta}$ solving the least squares minimization of equation (4.6) taking into account that the spatio-temporal derivatives that are involved are now the* projected *spatio-temporal derivatives.*
3. *Update $\vec{\theta}$ and register the images using the new estimates $\vec{\theta}$ and repeat step 1 using this new set of images as input.*

Repeat these three step until a certain degree of tolerance of the error function (4.6) (usually 5-10 iterations). Note that this error measurement can not be zero for the type of sequences that we are considering in this chapter (due to stochastic motions). It is worth emphasizing that steps 1 and 2 of this process are performed by solving systems of linear equations with closed form solutions.

This process has to be performed for each interval. The resulting estimates are parameters of the *relative* transformations between consecutive images, however it is easy to build the global transformation parameters for a specific absolute coordinate system by means of the composition rule for affine matrices. The results[3] for this sequence are shown in figures 4.5 and 5.4.

Figure 4.5: Two registered frames of the sequence. (See video 1 :*"video1original.mpeg"* and *"registvideo1.avi"*).

4.3.3 Reliability Analysis for the Estimates

As previously mentioned, the selection of the number of frames, the number of PC for the appearance subspace and the number of iterations all play an important role in the final result. An unsuitable selection of them may contribute to an error propagation when composing the transformation parameters in order to frame the registered

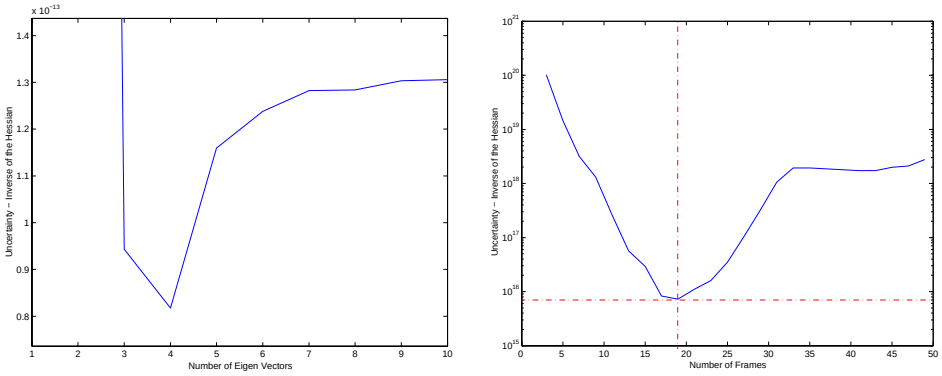
[3]See attached videos.

Figure 4.6: Two registered frames of the sequence. (See video 2 :*"video2original.mpeg"* and *"registvideo2.avi"*).

images in an absolute coordinate system. To overcome this problem, we propose two methods. The first involves performing the computation using a sliding interval, i.e. $(1,2,3,4,5) \rightarrow (2,3,4,5,6) \rightarrow (3,4,5,6,7)\ldots$, and take only the transformation parameters from the first to the second in each window , i.e. $\vec{\theta}_{12}, \vec{\theta}_{23}, \vec{\theta}_{34}, \ldots$, instead of partitioning the sequence into intervals. This technique obviously has a higher computational cost and in most cases is not necessary. The second method is developed in this section in order to demonstrate the contributions of a proper representation for the images of a sequence.

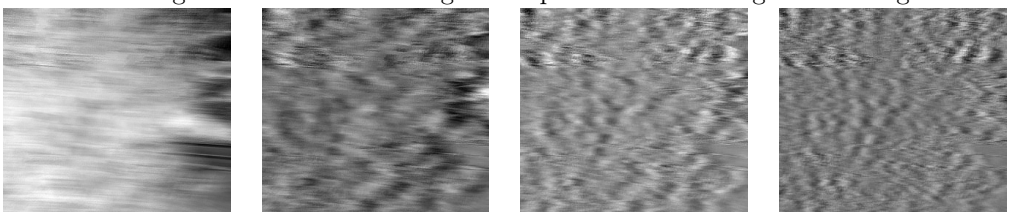
Influence of the Control Parameters

The approach of this analysis is based on the application of a Bayesian model comparison technique. The goal is to compute the uncertainty of the parameters under a specific model. The models that we are comparing differ in the selection of control parameters: number of frames and number of PC. It's easy to show [34] that this uncertainty is obtained from the Hessian of a specific error function. In our case, the error function is obtained from equation (4.6), and its Hessian corresponds to equation (4.7). The inverse of the determinant of this Hessian matrix measures how uncertain $\delta\vec{\theta}$ the maximum likelihood ML estimate $\hat{\vec{\theta}}$ is. First, we have taken 10 frames from the sequence that is related to the images in figure 4.5. The aim here is to compute what the most reliable selection of the number of PC for a fixed number of frames is. To this end, we have computed the ML estimates $\hat{\vec{\theta}}$ at different numbers of PC: from 1 to 10 PC. In figure 4.7(a) we show the uncertainty $\delta\vec{\theta}$ for each ML estimate $\vec{\theta}$ as a function of the number of PC that have been taken into account in the registration algorithm. For this sequence, a small number of PC (less than 4) does not capture all the relevant information from the images, therefore, the uncertainty is significantly high. On the other hand, when the number of PC tends towards the number of images ($q \rightarrow 10$) the uncertainty of the ML estimates increases, since within this limit there is no difference between the brightness constancy assumption and the *projected* BC. In this sequence, the optimal number corresponds to 4 PC, which evolve from a



(a) (b)

First 4 Eigen Vectors for the Original Sequence - 10 Non-registered Images -



First 4 Eigen Vectors for the Registered Sequence - 10 Images -

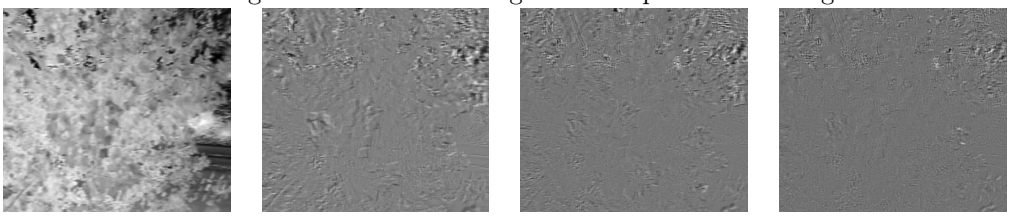


Figure 4.7: (a) Uncertainty of the ML estimates as a function of the selected number of PC basis. (b)Uncertainty of the ML estimates as a function of the selected number of frames (time scale). We have taken the first 10 frames of the sequence of 50 images (see fig. 4.5) and computed their first 4 eigen vectors before registration (first row) and after registration (last row). The first ones capture the variance due to camera transformation, and, the last ones due to the motion of the leaves.

first state shown in fig.4.7(first row of images) to a final state in fig. 4.7(bottom row) after convergence (10 iterations). The first state shows that the variations are due to camera transformations (in this specific case mainly translation) and the final state correspond to different scales of the motion of the leaves.

In figure 4.7(b) we show the uncertainty as a function of the number of frames that were taken into account for the registration algorithm. Note that when the number of frames is small the uncertainty is high since no time scale for the camera transformations is captured. When the number of frames increases, the occlusion effects appear. The optimal selection for the number of images is 19.

Bootstrap for Local Standard Deviation Analysis

The second issue we have studied is a local analysis of the uncertainty for a given fixed set of control parameters (time scale and PC). Our starting point is the functional in equation (4.6): $\mathcal{E}(\vec{\theta}) = \sum_{\vec{x}} \mid \delta_t I(\vec{x}) - \widetilde{\nabla} I(\vec{x})^T M(\vec{x}) \vec{\theta} \mid^2$, which uniformly sums the errors over all the pixel locations. In this case, the issue is focussed on *the contribution of each pixel to the uncertainty of the estimates*. To this end, the idea is to generate a large number of data sets, such that each set contains a random distribution of the original pixel locations. All the data sets contain the same number of pixels as the original one but with different distributions. This means that, in each set, some pixels may contribute to the error function more than once, and some others may not. Therefore, we estimate the parameters for each specific data set by means of minimizing eq. (4.6). Given that we are working with a large number of data sets, we are able to obtain a sample mean for the parameter estimates $< \vec{\theta} >$ and a measurement of their uncertainty. This is actually a numerical way of computing the Hessian that was mentioned previously. However, we are here interested on making computations using all the estimates obtained from the new *virtual* data sets.

We have taken the sequence of 10 images corresponding to the top row in figure 4.8, and we are now interested on the camera transformation estimation between the first and the second frame $\vec{\theta}_{12}$. Each image has dimensions 360×288. To build each *virtual* data set of pixel locations we select a random set of pixel locations that belong to the domain $[1, 360] \times [1, 288]$; and the total number of random pixels is the same as in the original image, i.e. 360×288. Note that some pixels may appear more than once and some others may not, this means that the contribution to the error functional of each pixel location $\vec{x}$ is now weighted by the number of times $\omega(\vec{x})$ that has been taken into account:

$$\mathcal{E}(\vec{\theta}_{12}) = \sum_{\vec{x}} \omega(\vec{x}) \mid \delta_t I(\vec{x}) - \widetilde{\nabla} I(\vec{x})^T M(\vec{x}) \vec{\theta}_{12} \mid^2$$

Now the estimation of $\vec{\theta}_{12}$ proceeds as in the algorithm described before. Therefore we obtain for each virtual set a parameter vector $\vec{\theta}^i_{12}$. We have generated 500 virtual data sets and run the estimation process with this particular weighing issue for all of them obtaining a set of 500 parameter vectors: $\{\vec{\theta}^1_{12}, \ldots, \vec{\theta}^i_{12}, \ldots, \vec{\theta}^{500}_{12}\}$. Using each parameter vector, we register image 2. Now we are dealing with a set of 500 registered images. Next, we have computed the standard deviation of the correspond-

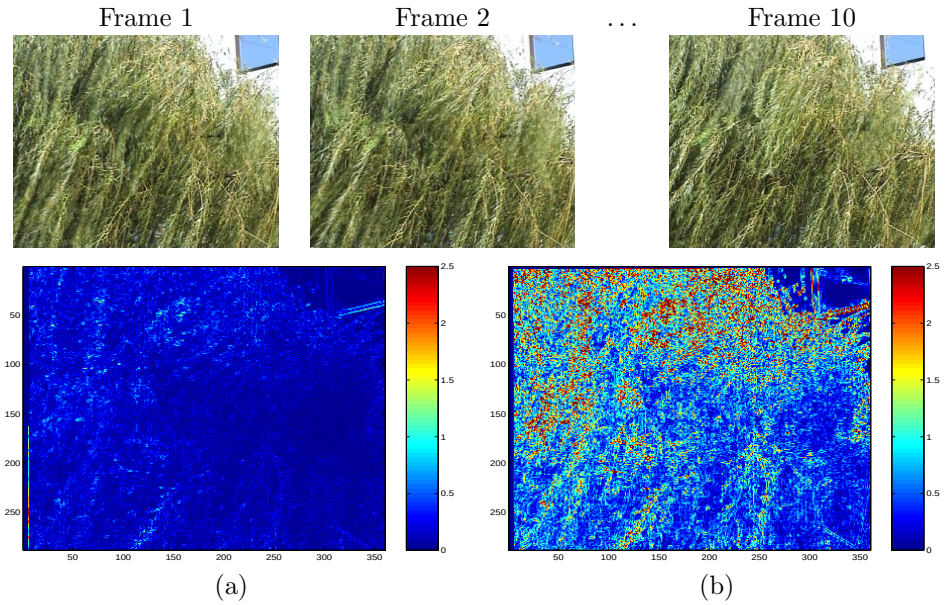


Figure 4.8: First, second and last frames of the sequence (top row). (See video 3 :*"video3original.mpeg"* and *"registvideo3.avi"*). (a) Standard deviation map corresponding to a local bootstrap analysis for our algorithm. (b) Standard deviation map obtained from a bootstrap analysis of a standard 2-frame parametric alignment technique.

ing 500 pixel values at each pixel location. This yields a map showing a reliability distribution as a function of each position in the image. Pixel locations with bigger standard deviation correspond to positions with a less reliable contribution to the camera motion estimation. Therefore, we have repeated a similar experiment using a standard 2-frame parametric alignment technique. Note that in our algorithm we have been using the information encoded in the sequence of ten images to perform the registration between the first and the second frame. On the other hand, for the standard 2-frame parametric technique, we used just two images. This should be taken into account when interpreting the results of this experiment. We notice too that there is a significant difference in the distribution of the standard deviation maps (figs. 4.8(a) and 4.8(b)). In case (b) variations concentrate on regions where the motion of leaves was entangled with the motion of the camera. Higher amounts of variation tell us about the areas in the image where the camera motion estimation is less reliable. This fact is due to the lack of global temporal information ($\sim$ time scale) in the standard parametric technique.

4.4 Conclusions

As an alternative to standard $2D$ registration techniques, in this chapter we have proposed an appearance based framework for video stabilization. We have addressed 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.