Appears in IEEE Conference Computer Vision and Pattern Recognition, Santa Barbara, California, 1998.

A Spatiotemporal Motion Model for Video Summarization

Nuno Vasconcelos Andrew Lippman
MIT Media Laboratory - {nuno,lip}@media.mit.edu

Abstract

The compact description of a video sequence through a single image map and a dominant motion has applications in several domains, including video browsing and retrieval, compression, mosaicing, and visual summarization. Building such a representation requires the capability to register all the frames with respect to the dominant object in the scene, a task which has been, in the past, addressed through temporally localized motion estimates. In this paper, we show how the lack of temporal consistency associated with such estimates can undermine the validity of the dominant motion assumption, leading to oscillation between different scene interpretations and poor registration. To avoid this oscillation, we augment the motion model with a generic temporal constraint which increases the robustness against competing interpretations, leading to more meaningful content summarization.

1 Introduction

Given the ubiquity of bandwidth, connectivity, storage, and computational resources associated with modern communications networks, massive repositories of pictorial information start to appear throughout them. The usefulness of such repositories will be, to a significant extent, determined by the availability of systems which can help users navigate through them, and interact with or manipulate their content.

In the case of video databases, the magnitude of stored information is by itself an overwhelming problem as on-line analysis of each frame in the video stream becomes impractical, even if this analysis consists only of very simple operations. There is, therefore, a need to develop procedures for the automatic summarization of video content which can then be used to speed up browsing and retrieval operations. Of particular interest are methods capable of providing *visual* summarization of the video streams, as these summaries can be directly inspected by human users of the video repository.

Due to this interest and the fact that visual summarization of video sequences has application in a wide range of other domains, a significant body of research has been devoted to this topic in the recent past. The fundamental idea is to compute a single image map which is representative of the pictorial content of the video sequence by warping all the frames contained in it into a reference coordinate frame and somehow combining their pixel intensities. Because different solutions to the problem have evolved in different research communities, with different applications in mind, the resulting representations have received diverse names. Among these are *salient stills* [6], *video mosaics* [8, 9], *video sprites* [7], and *video layers* [11][1].

In spite of this diversity, all these procedures are similar in the sense that they follow the following two fundamental steps.

1. Fitting a global motion model to the motion between each pair of successive frames.
2. Computing the summarizing image map by accumulating the information from all the frames after they have been aligned according to the motion estimates computed in the previous step.

In this paper, we show that relying on temporally localized motion estimates limits the ability of these representations to produce an image map that meaningfully summarizes the video content. This is a direct consequence of the fact that representations based on temporally localized motion models cannot capture the global characteristics of the video stream along the temporal dimension. While the intensity map on which they rely contains visual information summarizing the entire sequence and the parametric motion description is valid over the entire spatial extent of any given frame, the underlying motion models account only for a highly localized temporal neighborhood (usually a frame pair) of the spatiotemporal volume spanned by the sequence. Therefore, they provide no guarantee of coherence along the temporal dimension, allowing motion estimates to oscillate between competing scene interpretations and leading to poor image registration.

[1] While, strictly speaking, layering always includes the construction of multiple image maps, the construction of each of these can be implemented, once the scene is segmented into the objects of interest, by the procedures discussed in this paper.

In this work, we introduce a truly global motion representation, in both the spatial and temporal dimensions, by augmenting the motion model with a generic temporal constraint that avoids this oscillation. The resulting model is parametric in both space and time and can be fitted to the *entire* sequence at once, with marginal increase in computational complexity. Consequently, it locks to the motion which is dominant over the *entire* spatiotemporal volume, leading to temporal coherence and a significantly better content summarization.

We would also like to point out that, while the focus of this paper is on visual summarization, the advantages of a parametric spatiotemporal motion representation are not limited to this domain. For example, the fact that a compact description of the dominant motion throughout the entire sequence is available, also makes the representation attractive for content-based retrieval. Because our representation originates a single image map and a single spatiotemporal parameter vector for each sequence, it allows retrieval based on either the map, the motion, or both. Motion based retrieval is difficult when motion is characterized by a large set of temporally localized estimates.

2 Content summarization by image registration

The main assumption underlying procedures for video summarization through image registration is that there is a *dominant* motion among the motions of the various objects in the scene. If there is a single motion (e.g. a static scene and a moving camera) then (assuming the motion model matches the true scene motion) the summarization is perfect. If more than one motion is present, the object with the dominant motion is correctly aligned and the remaining objects are "blurred out". The result is a summarizing map where the dominant object appears crisp and the remaining objects are substituted by ghostly versions that provide a sense for the action in the scene.

One of the main limitations of the dominant motion assumption is that it is not always straightforward to determine what motion will be dominant. To illustrate this point, consider a sequence of a bird flying in a region of uniformly blue sky. Because the sky has no texture and, therefore, any motion will be a good fit for the sky region, the dominant motion will be that of the bird. If, however, the sky is textured (e.g. it contains clouds or stars) or there is also a tree in the background, the motion of the bird will no longer dominate. In practice, which motion is dominant depends on the relative sizes of the objects, how they are textured, the relative amplitudes of their velocities, and the occlusion relationships originated as they move.

The problem is that all these factors change as the sequence progresses and the dominant motion may not be dominant at all instants. This is illustrated by the simple example of Figure 1 which displays three snapshots of a sequence composed by two squares of similar texture but different sizes, translating at the same speed in opposite directions. When there is overlap, the smaller square (B) occludes the larger one (A).

Since all other factors are equal, the dominant motion is that of the square with the largest number of visible pixels, and A will dominate for most of the sequence. However, in the period where B occludes A (depicted by the center snapshot), there may be several frame-pairs for which B dominates. Hence, as shown in Figure 2, the estimate of the dominant motion will switch between the two possibilities as the sequence progresses. In result, neither of the two objects will be correctly aligned in the resulting summarizing map, i.e. both will be blurred-out to at least some extent, and it will be much harder to perceive the scene dynamics from this map than if the registration would have been performed with respect to one of the squares alone.

The importance of integrating motion estimates throughout the sequence has been realized by Irani and co-workers in [5]. They propose a recursive procedure for building the map on the fly where, for each frame, they compute the best affine motion estimate between the current map estimate and that frame. The map is then registered with the frame and updated by taking a weighted average of the two. The rationale is that, as the sequence progresses, the map locks onto the object of dominant motion and the other objects are blurred out. This, in turn, reinforces the lock[2].

In the case of the figure, such a procedure would start by following A, and B would initially be wiped out of the summarization map. However, as soon as there were overlap between the two squares, some of B's texture would start to be included as well. By the time of the center snapshot in the figure, depending on the rate at which old information is discarded from the map and the velocities of the two objects, the map's texture would either resemble that of A, that of B or something in between. While in the first case everything would go well; in the latter two, B would, with high likelihood, be tracked throughout the rest of the sequence, leading to a situation even worst than that

[2]We should note that their work was not aimed at recovering an image map for summarization, but instead to obtain improved motion estimates and segmentations.

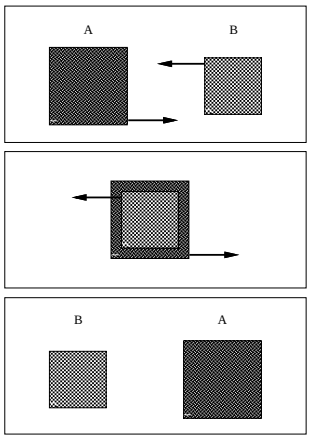

Figure 1: Three snapshots of a sequence where temporally localized motion estimates fail to identify the dominant motion.

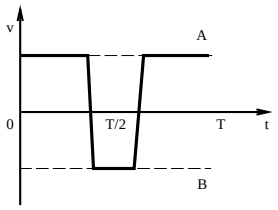

Figure 2: Velocities of each of the objects in the sequence of Figure 1 as a function of time. Dashed lines indicate the paths, in velocity space, of each of the squares. The solid line indicates the trajectory of the dominant motion, according to the temporally localized motion model. The occlusion near $t = T/2$ leads to a switch regarding which motion is dominant.

of Figure 2.

Even though temporal integration is a correct step towards eliminating the uncertainty originated by several competing scene interpretations, it does not completely address the difficulties created by the fact that different interpretations may become dominant at different time instants. This issue can only be addressed through representations capable of capturing dominance over the entire spatiotemporal volume spanned by the sequence. We next introduce a spatiotemporal motion model which leads to representations with such properties.

3 The spatiotemporal motion model

We start by assuming that the motion between consecutive frames in the video sequence can be characterized by an affine transformation, i.e.

$$d^x_{j,j+1} = c^1_j + c^2_j x_j + c^3_j y_j \quad (1)$$

$$d^y_{j,j+1} = c^4_j + c^5_k x_j + c^6_k y_j, \quad (2)$$

where j is the frame number, $\mathbf{x}_j = (x_j, y_j)^T$ are the image coordinates of pixel $\mathbf{x}$, and

$$\mathbf{d}_{j,j+1} = (d^x_{j,j+1}, d^y_{j,j+1})^T = (x_{j+1} - x_j, y_{j+1} - y_j)^T$$

is the displacement applied to the pixel from frame j to frame $j+1$. However, in order to guarantee consistency of motion estimates across time, we augment the motion model by imposing a generic temporal constraint: *each pixel follows a path along the sequence according to a smooth trajectory characterized by a (low-order) polynomial, i.e.*

$$\mathbf{x}_j = \mathbf{x}_0 + \sum_{i=0}^{M} \phi_i t^i_j, \quad (3)$$

where t_k is the time-stamp of frame k. The number $M+1$ of terms of this polynomial provides a trade-off between the degree of smoothness of the approximation, and the capability of following the pixel's trajectory. If $M+1 = N$, where N is the number of frames in the sequence, the model can follow exactly any possible trajectory, but provides no extra constraint other than those already imposed by the affine model. On the other hand, if $M = 0$ the model forces the pixel to land in the same location at every frame, i.e. allows no motion. In our experience, a low-order polynomial provides a good compromise between these factors - we have used $M = 2$ in the experiments reported in section 5. The framework is, however, generic and valid for any value of M.

In appendix A, we show that, given the temporal constraint of equation (3), the motion between successive frames will be affine if and only if the polynomial

coefficients ϕ_i are themselves the result of an affine transformation of $\mathbf{x}_0$, i.e.

$$\phi_i^x = \rho_i^1 + \rho_i^2 x_0 + \rho_i^3 y_0 \quad (4)$$
$$\phi_i^y = \rho_i^4 + \rho_i^5 x_0 + \rho_i^6 y_0. \quad (5)$$

Substituting these equations in equation (3) and grouping terms, we obtain

$$d_{0,j}^x = \sum_{i=0}^{M}(\rho_i^1 t_j^i) + \sum_{i=0}^{M}(\rho_i^2 t_j^i)x_0 + \sum_{i=0}^{M}(\rho_i^3 t_j^i)y_0, \quad (6)$$
$$d_{0,j}^y = \sum_{i=0}^{M}(\rho_i^4 t_j^i) + \sum_{i=0}^{M}(\rho_i^5 t_j^i)x_0 + \sum_{i=0}^{M}(\rho_i^6 t_j^i)y_0, \quad (7)$$

i.e. the displacement of the pixel between frames 0 and j is the sum of $M+1$ affine transformations with coefficients proportional to the $M+1$ powers of t_j. Defining

$$\Phi(\mathbf{x}_0) = \begin{bmatrix} 1 & x_0 & y_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_0 & y_0 \end{bmatrix}, \quad (8)$$

$$\mathcal{T}_j = \begin{bmatrix} t_j^M \mathbf{I}_6 & \dots & t_j \mathbf{I}_6 & \mathbf{I}_6 \end{bmatrix},$$

and

$$\mathbf{p} = (\mathbf{p}_M, \dots, \mathbf{p}_0)^T,$$

where $\mathbf{I}_6$ is the identity matrix of order six, and $\mathbf{p}_i = (\rho_i^1, \dots, \rho_i^6)^T, i = 0, \dots, M$, the spatiotemporal trajectory of the point can be written in a compact form as

$$\mathbf{x}_j = \mathbf{x}_0 + \Phi(\mathbf{x}_0)\mathcal{T}_j\mathbf{p} = \Psi_j(\mathbf{x}_0). \quad (9)$$

4 Estimation of the model components

Given a video sequence $\mathcal{F}_1, \dots, \mathcal{F}_N$, we model each of the frames, $\mathcal{F}_j$, as the outcome of a Gaussian process with mean described by the affine warping of the summarizing map $\mathcal{S}$, temporally co-located with $\mathcal{F}_1$. From equation (9), and dropping the subscript of $\mathbf{x}_0$,

$$P(\mathcal{F}_j(\Psi_j(\mathbf{x}))|\mathbf{p}, \mathcal{S}(\mathbf{x})) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(\mathcal{F}_j(\Psi_j(\mathbf{x})) - \mathcal{S}(\mathbf{x}))^2}{2\sigma^2}}.$$

Assuming that each of the Gaussian variables is independent, the joint density for all the pixels in the sequence is characterized by

$$P(\mathcal{F}_1, \dots, \mathcal{F}_N|\mathbf{p}, \mathcal{S}) \propto \exp\left\{\sum_{j,\mathbf{x}}(\mathcal{F}_j(\Psi_j(\mathbf{x})) - \mathcal{S}(\mathbf{x}))^2\right\}.$$

In order to determine the parameters of the spatiotemporal motion model and the summarizing map $\mathcal{S}$ which best explain the observed image data, we rely on a Maximum Likelihood (ML) framework, according to which the optimal motion parameters and summarizing map are those which minimize the cost function

$$\mathcal{J}(\mathbf{p}, \mathcal{S}(\mathbf{x})) = \sum_{j,\mathbf{x}}(\mathcal{F}_j(\mathbf{x} + \Phi(\mathbf{x})\mathcal{T}_j\mathbf{p}) - \mathcal{S}(\mathbf{x}))^2. \quad (10)$$

The minimization is performed by iterating between the estimation of the motion parameters given an estimate of the summarizing map, and the updating of the map given the new parameter values. Given an estimate for $\mathcal{S}$, the optimal new set of parameters $\mathbf{p}'$ is

$$\mathbf{p}' = \min_{\mathbf{p}} \mathcal{J}(\mathbf{p}, \mathcal{S}(\mathbf{x})) \quad (11)$$

and, given this new set of parameters, the updated estimate of $\mathcal{S}$ is, for each location of the map,

$$\mathcal{S}'(\mathbf{x}) = \min_{\mathcal{S}(\mathbf{x})} \mathcal{J}(\mathbf{p}', \mathcal{S}(\mathbf{x})). \quad (12)$$

4.1 Estimating the motion parameters

To minimize equation (11) we rely on the *Gauss-Newton* method [2] which, as shown in appendix B, leads to an iterative procedure of the form

$$\mathbf{p}^{k+1} = \mathbf{p}^k + \gamma^k \mathbf{d}^k, \quad (13)$$

where

$$\mathbf{d}^k = \left[\sum_j \mathcal{T}_j^T \alpha_j^k \mathcal{T}_j\right]^{-1} \sum_j \mathcal{T}_j \beta_j^k, \quad (14)$$

$$\alpha_j^k = \sum_{\mathbf{x}} \Phi(\mathbf{x})^T \nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j^k(\mathbf{x})) \nabla_{\mathbf{x}}^T \mathcal{F}_j(\Psi_j^k(\mathbf{x})) \Phi(\mathbf{x}), \quad (15)$$

$$\beta_j^k = \sum_{\mathbf{x}} [\mathcal{F}_j(\Psi_j^k(\mathbf{x})) - \mathcal{S}(\mathbf{x})] \Phi(\mathbf{x})^T \nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j^k(\mathbf{x})), \quad (16)$$

$\mathcal{F}_j(\Psi_j^k(\mathbf{x}))$ is the result of warping the j^{th} frame with the current estimate of the transformation associated with it (Ψ_j^k), $\nabla_{\mathbf{x}}$ the gradient with respect to the image coordinates, and γ_k a scalar determined by a line-search.

The procedure for the estimation of the spatiotemporal motion parameters can therefore be summarized as follows.

1. Set $k = 0$. Compute an initial parameter estimate $\mathbf{p}^0$. Our initialization strategy is to compute the affine transformations between successive frames in the sequence, through a variation of the method proposed in [1], and then find the $\mathbf{p}^0$ that provides the least squares fit to the temporally localized estimates.

2. For each frame in the sequence, $\mathcal{F}_j, j = 1 \ldots N$:

   - warp the frame according to the current estimate of the motion parameters $\mathbf{p}^k$ and equation (9);
   - compute the spatial gradient of the warped frame, $\nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j^k(\mathbf{x}))$;
   - compute α_j^k and β_j^k according to equations (15) and (16).

3. Compute $\mathbf{d}^k$.

4. Find γ^k by a line search. In our implementation, this is done by considering $\gamma_l^k = 2^{-l}, l = 0, \ldots, 4$, computing $\mathbf{p}_l^{k+1} = \mathbf{p}^k + \gamma_l^k \mathbf{d}^k$ for every l, and choosing the one which minimizes the cost function of equation (11).

5. If $||\mathbf{p}^{k+1} - \mathbf{p}^k|| < T$, where T is a pre-defined threshold, stop. Otherwise, set $k = k + 1$ and go to 2.

It can be shown [10] that the entire process requires only a marginal increase of computation in relation to that already required by the frame-based motion estimates.

4.2 Updating the summarizing map

Once the optimal motion parameters are determined, the estimate of the map $\mathcal{S}$ can be updated through the minimization of equation (12). It is straightforward to show that setting to zero the derivative, with respect to $\mathcal{S}(\mathbf{x})$, of this equation leads to

$$\mathcal{S}'(\mathbf{x}) = \frac{1}{N} \sum_j \mathcal{F}_j(\mathbf{x} + \Phi(\mathbf{x}) \mathcal{T}_j \mathbf{p}'). \qquad (17)$$

This has the intuitive appeal that once the optimal motion parameters are found, the optimal summarization map is simply the mean of all the images in the sequence after they are warped to the map's coordinate frame. Equation (17) has, in fact, been used in the majority of previous proposals for the the construction of image layers [11] and mosaics [8].

Given the new summarizing map, a new set of motion parameters can be computed, leading to the iterative minimization of equations (11) and (12). Notice that, since each step in the iteration is guaranteed to decrease the cost function or leave it unchanged and the cost function is bounded below by zero, the procedure is guaranteed to converge to a (possibly local) minimum.

5 Summarization results

In order to test the improvements obtained with spatiotemporal modelling, we applied both the temporally localized and the spatio-temporal model to the summarization of various sequences. The model relying on pairwise affine estimates works well when there is a single global motion (e.g. static scene and moving camera), but runs into problems whenever there is ambiguity with repect to motion dominance. This is illustrated by Figure 3 which depicts a scene consisting of a static weakly textured background, and a person with approximatly affine body motion, and non-rigid arm motion.

The summarizing maps on the bottom left and right of the figure were obtained, respectivelly, with the temporally localized motion model and the spatiotemporal model with a second-order temporal constraint ($M = 2$ in equation (3)). While the computational cost of the two methods is comparable, the temporally localized motion model leads to an erratic estimate and significant uncertainty in the recovered map. On the other hand, the spatiotemporal model locks onto the body motion, leading to a map that summarizes the scene content in a much more meaningful way.

Notice that when all the frames are aligned with respect to the same object (in this case the body), it is not only easier to recognize this object (the person), but also to understand the scene dynamics. In the case of the figure, the spatiotemporal map provides a significantly better description for the motion of the arm throughout the sequence (even though the arm serves as a reference for some of the frames when the temporally localized model is used).

A Constraints on the temporal coefficients

Assuming that the trajectories of points in the image plane satisfy the constraint of equation (3), we now determine how the coefficients of that equation must, themselves, be constrained in order to guarantee affine motion between consecutive frames (equations (1) and (2)). For this, we prove the following theorem.

Theorem 1 *Consider a motion trajectory satisfying the constraint of equation (3). Then $\mathbf{x}_j$ is an affine transformation of $\mathbf{x}_0$ if and only if each of the coefficients in the equation is itself an affine transformation of $\mathbf{x}_0$. I.e. for a motion trajectory satisfying equation (3), $\mathbf{x}_j$ is an affine transformation of $\mathbf{x}_0$ if and only if equations (4) and (5) are satisfied.*

Proof:

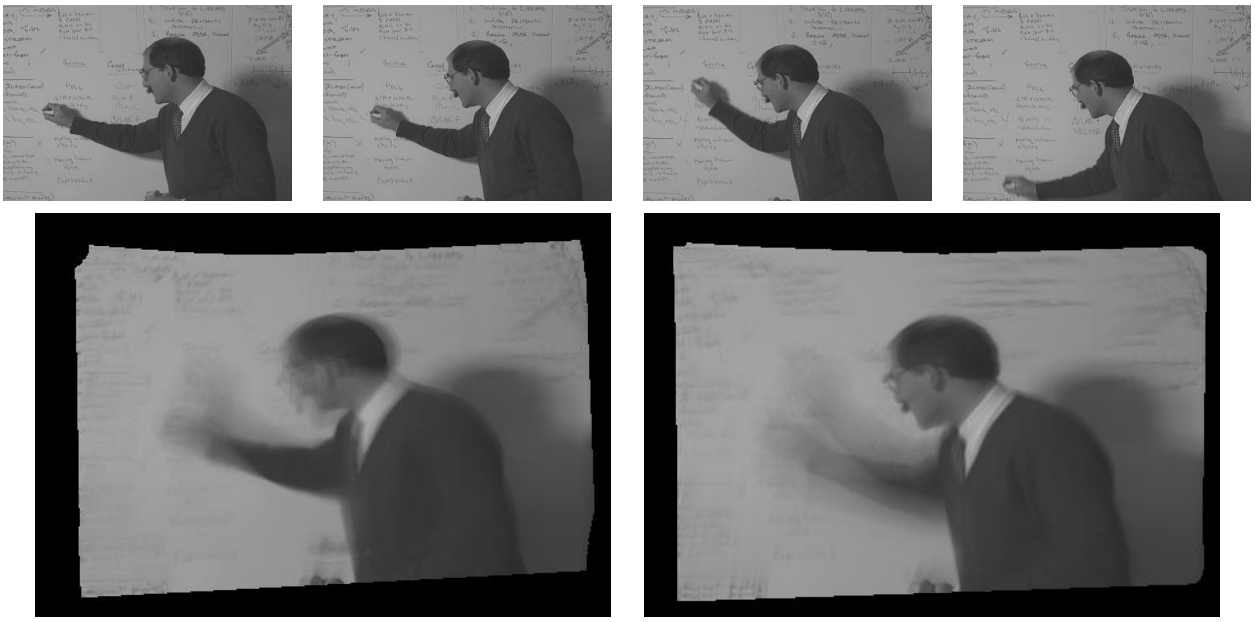

Figure 3: Four frames from a sequence containing three motions and high dominance ambiguity (top) and summarizing maps obtained with temporally localized (bottom-left) and with the spatiotemporal model (bottom-right). When localized estimates are used, registration is sometimes performed with respect to the body and other times with respect to the moving arm.

i) Assume equations (4) and (5) hold. Then by simple substitution in equation (3) we obtain equations (6) and (7). Comparing these equations with (1) and (2), it is clear that the former define an affine transformation between $\mathbf{x}_0$ and $\mathbf{x}_j$.

ii) In order to prove the reverse direction, we start by considering an *homogeneous* coordinate system [3], where $\mathbf{X}_j = (1, x_j, y_j)^T$ and noting that, in such a coordinate system, affine transformations are obtained by matrix multiplication. I.e. if $\mathbf{X}_j$ is an affine transformation of $\mathbf{X}_0$, then

$$\mathbf{X}_j = \mathbf{Q}_j \mathbf{X}_0, \tag{18}$$

where

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ - & - & - \\ - & - & - \end{bmatrix}, \tag{19}$$

and $-$ can be any real number. In the new coordinate system, equation (3) becomes

$$\mathbf{X}_j = \mathbf{X}_0 + \sum_{i=0}^{M} \mathbf{\Phi}_i t_j^i, \tag{20}$$

with $\mathbf{\Phi}_i = (0, \phi_i^T)^T$.

Assume that $\mathbf{x}_j$ is an affine transformation of $\mathbf{x}_0$. Then, combining equations (18) and (20)

$$\sum_{i=0}^{M} \mathbf{\Phi}_i t_j^i = (\mathbf{Q}_j - \mathbf{I})\mathbf{X}_0, j = 1, \ldots, N,$$

where N is the number of frames in the sequence. Next, pick any M distinct j (for example the first M) and construct the following system of equations

$$\begin{bmatrix} \vdots \\ \mathbf{X}_0^T(\mathbf{Q}_j^T - \mathbf{I}) \\ \vdots \end{bmatrix} = \begin{bmatrix} & \vdots & \\ 1 & t_j & \ldots & t_j^{M-1} \\ & \vdots & \end{bmatrix} \begin{bmatrix} \mathbf{\Phi}_0^T \\ \mathbf{\Phi}_1^T \\ \vdots \\ \mathbf{\Phi}_{M-1}^T \end{bmatrix}. \tag{21}$$

Calling $\mathbf{T}$ the matrix which is a function of the t_j^i, and noticing that it is a Vandermonde matrix, it is clear that (because all the j are different) it has full rank [4]. The system can thus be inverted, leading to

$$\mathbf{\Phi}_i^T = \left(\mathbf{T}^{-1}\right)_i \mathbf{V}, i = 1, \ldots, M,$$

where $\left(\mathbf{T}^{-1}\right)_i$ is the i^{th} row of $\mathbf{T}^{-1}$, and $\mathbf{V}$ the matrix on the left-hand side of equation (21). Hence, each $\mathbf{\Phi}_i$ is a linear combination of all the $(\mathbf{Q}_j - \mathbf{I})\mathbf{X}_0$ vectors,

i.e.

$$\mathbf{\Phi}_i = \left(\sum_{j=0}^{M-1} \mu_j (\mathbf{Q}_j - \mathbf{I}) \right) \mathbf{X}_0,\ i = 1, \ldots, M.$$

Because the $\mathbf{Q}_j$ matrices are of the form given in equation (19), the matrices $(\mathbf{Q}_j - \mathbf{I})$ have zeros in all the positions of their first rows and the equation becomes

$$\begin{bmatrix} 0 \\ \phi_i \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ - & - & - \\ - & - & - \end{bmatrix} \mathbf{X}_0, i = 1, \ldots, M$$

i.e. ϕ satisfies equations (4) and (5) for all i. □

It follows from the properties of affine transformations that if, for all j, $\mathbf{x}_j$ is an affine transformation of $\mathbf{x}_0$, then it is also an affine transformation of $\mathbf{x}_{j-1}$, i.e. the motion between consecutive frames is affine.

B Parameter estimation

The optimal set of spatiotemporal motion parameters is, for a given map, the one which minimizes equation (11). As pointed out in section 4.1, this minimization is carried out through the Gauss-Newton method. For a least squares cost function

$$\mathcal{J}(\mathbf{p}) = \sum_i J_i(\mathbf{p})^2,$$

this method consists of the iteration described by equation (13) with

$$\mathbf{d}^k = \left[\sum_i \nabla_{\mathbf{p}} J_i(\mathbf{p})^T \nabla_{\mathbf{p}} J_i(\mathbf{p}) \right] \sum_i J_i(\mathbf{p})\, \nabla_{\mathbf{p}} J_i(\mathbf{p}).$$

For the cost function of equation (10)

$$J_i(\mathbf{p}) = \mathcal{F}_j(\Phi(\mathbf{x})\mathcal{T}_j\mathbf{p}) - \mathcal{S}(\mathbf{x})$$

and

$$\nabla_{\mathbf{p}} J_i(\mathbf{p}) = \mathcal{T}_j^T \Phi(\mathbf{x})^T \nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j(\mathbf{x})),$$

where $\Psi_j(\mathbf{x})$ is defined by equation (9), leading to

$$\mathbf{d}^k = \left[\sum_{j,\mathbf{x}} \mathcal{T}_j^T \Phi(\mathbf{x})^T \nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j(\mathbf{x})) \nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j(\mathbf{x}))^T \Phi(\mathbf{x}) \mathcal{T}_j \right]^{-1} \times \sum_{j,\mathbf{x}} (\mathcal{F}_j(\Psi(\mathbf{x})) - \mathcal{S}(\mathbf{x}))\, \mathcal{T}_j^T \Phi(\mathbf{x})^T \nabla_{\mathbf{x}} \mathcal{F}_j(\Psi_j(\mathbf{x})). \quad (22)$$

Since the $\mathcal{T}_j$ do not depend on $\mathbf{x}$, they can be taken out of the summation with respect to the spatial coordinates, leading to equations (14) to (16).

References

[1] P. Anandan, J. Bergen, K. Hanna, and R. Hingorani. Hierarchical Model-Based Motion Estimation. In M. Sezan and R. Lagendijk, editors, *Motion Analysis and Image Sequence Processing*, chapter 1. Kluwer Academic Press, 1993.

[2] D. Bertsekas. *Nonlinear Programming.* Athena Scientific, 1995.

[3] J. Foley, A. van Dam, S. Feiner, and J. Hughes. *Computer Graphics: Principles and Practice.* Addison Wesley, 1990.

[4] R. Horn and C. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[5] M. Irani, B. Rousso, and S. Peleg. Computing Occluding and Transparent Motions. *International Journal of Computer Vision*, 12:1, 1994.

[6] M. Massey and W. Bender. Salient Stills: Process and Practice. *IBM Systems Journal*, Vol. 35(3 and 4), 1996.

[7] M.Lee, W. Chen, C. Lin, C. Gu, T. Markoc, S. Zabinsky, and R. Szeliski. A Layered Video Object Coding System Using Sprite and Affine Motion Model. *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 7, February 1997.

[8] H. Sawhney and S. Ayer. Compact Representations of Videos Through Dominant and Multiple Motion Estimation. *IEEE Trans. on Pattern. Analysis and Machine Intelligence*, Vol. 18, August 1996.

[9] R. Szeliski. Image Mosaicing for Tele-Reality Applications. In *Proc. IEEE Workshop Applications of Computer Vision*, 1994.

[10] N. Vasconcelos and A. Lippman. Spatiotemporal Video Modeling for Content Summarization. Technical report, MIT Media Laboratory, 1997. Available from http://www.media.mit.edu/~nuno.

[11] J. Wang and E. Adelson. Representing Moving Images with Layers. *IEEE Trans. on Image Processing*, Vol. 3, September 1994.