FRAME-FREE VIDEO

Nuno Vasconcelos and Andrew Lippman

Massachusetts Institute of Technology
Media Laboratory
20 Ames St., Cambridge, MA 02139, USA
{nuno,lip}@media.mit.edu

ABSTRACT

Current digital video representations emphasize compression efficiency, lacking some of the flexibility required for interactive manipulation of digital bitstreams. In this work, we present a video representation which can encompass both space and time, providing a temporally coherent description of video sequences. The video sequence is segmented into its component objects, and the trajectory of each object throughout the sequence is described parametrically, according to a spatiotemporal motion model. Since the motion model is a continuous function of time, the video representation becomes frame-rate independent and temporal resolution a user-definable parameter. I.e. the traditional sequence of frames, with temporal structure hard-coded into the bitstream at the time of production, is replaced by a collection of scene snapshots assembled on the fly by the decoder. This enables random access and temporal scalability, the major building blocks for interactivity.

1. INTRODUCTION

If there is a defining characteristic of digital communications media that characteristic is the potential for *interactivity* - the ability of the user to actively search, browse through or even produce information, instead of passively "tuning in" to what is going on a broadcast channel [3]. Central to the idea of interactivity is the concept of *random access* - the ability to decode a portion of the bitstream (e.g. a frame in the case of a moving sequence) without having to decode the information which is immediately before or after. For example, browsing through digital video implies the ability to skip frames, typically at a variable rate (which decreases as the user approaches the point of interest), and new information is many times created by editing (cutting and pasting) video clips from different bit-streams into a new presentation.

Unfortunately, random access is not easy to implement with highly temporally localized digital video representations constructed upon motion estimation based on frame-pairs. Under the "frame-pair" paradigm, decoding a given frame in the bit-stream implies decoding information relative to (and usually even reconstructing) all the frames between that and a reference frame (*access point*) whose location was arbitrarily assigned at the time of encoding. This imposes a significant computational burden on tasks such as fast-forward, reverse play or cutting and pasting video clips.

This work was supported in part by the members of the Television of Tomorrow consortium.

In [6], we introduced a spatiotemporal motion representation that, by relaxing the temporal localization of motion estimation, allows a coherent description of motion throughout a sequence of frames, avoiding the limitations inherent to the "frame-pair" paradigm. In this framework, time is a variable of a parametric spatiotemporal motion representation, and decoding a single frame or a set of non-consecutive frames requires no more effort than when they are decoded in the ordinary playing order. In fact, such a representation can be considered as frame-free: it is approximately as hard to reconstruct frames at a pre-determined temporal rate as it is to reconstruct a synthetic version at a different frame-rate. Thus, the representation becomes frame-rate independent, and temporal resolution becomes a parameter defined by the user according to display device and processing capabilities.

The main limitation of our previous work was, however, an implicit assumption of a single moving object and no occlusions. Although the system was made robust by including a delayed decision estimation procedure, it did not rely on any segmentation procedures, and as such could not handle scenes with multiple objects or object occlusions. These issues are addressed in this paper, where we present a generic implementation of the spatiotemporal representation.

2. SPATIOTEMPORAL MODEL-BASED OPTIC FLOW

The core of our system is the multi-frame optic flow estimator that was presented in [6], and which we now briefly review. This optic flow estimator is based on the concept of *motion paths*, $\mathbf{p}_\kappa^{(t)} = (x_\kappa^{(t)}, y_\kappa^{(t)})^T$, the locus of coordinates in the image plane onto which each point κ in the 3D world is projected as time evolves. We approximate the projection of the true motion on the camera plane by a quadratic trajectory in time

$$\left[\begin{array}{c} x_\kappa^{(t+\delta t)} \\ y_\kappa^{(t+\delta t)} \end{array}\right] = \left[\begin{array}{c} x_\kappa^{(t)} + v_{\kappa_x}\delta t + a_{\kappa_x}\delta t^2 \\ y_\kappa^{(t)} + v_{\kappa_y}\delta t + a_{\kappa_y}\delta t^2 \end{array}\right], \quad (1)$$

where v_x, v_y, a_x, and a_y are the horizontal and vertical components of the velocity and acceleration associated with motion path κ, and a first-order affine transformation in space

$$\mathbf{p}_{\kappa}^{(\mathbf{t}+\delta\mathbf{t})} = \begin{bmatrix} p_{\kappa_{xx}}^{(\delta t)} & p_{\kappa_{xy}}^{(\delta t)} \\ p_{\kappa_{yx}}^{(\delta t)} & p_{\kappa_{yy}}^{(\delta t)} \end{bmatrix} \mathbf{p}_{\kappa}^{(\mathbf{t})} + \begin{bmatrix} p_{\kappa_{x0}}^{(\delta t)} \\ p_{\kappa_{y0}}^{(\delta t)} \end{bmatrix}. \qquad (2)$$

I.e. we assume the motion of each object to be characterized by an affine transformation between successive frames, but where each point follows a quadratic trajectory in time, so that the resulting motion estimates are continuous and coherent across several frames.

Two theoretical results were shown in [6]: 1) equations 1 and 2 can hold simultaneously if and only if the velocity and acceleration that characterize the motion paths are themselves affine transformations of the image coordinates (i.e. planes in velocity and acceleration space)

$$\begin{bmatrix} v_{\kappa_x} \\ v_{\kappa_y} \\ a_{\kappa_x} \\ a_{\kappa_y} \end{bmatrix} = \begin{bmatrix} v_{xx} & v_{xy} \\ v_{yx} & v_{yy} \\ a_{xx} & a_{xy} \\ a_{yx} & a_{yy} \end{bmatrix} \begin{bmatrix} x_{\kappa}^{(t)} \\ y_{\kappa}^{(t)} \end{bmatrix} + \begin{bmatrix} v_{x0} \\ v_{y0} \\ a_{x0} \\ a_{y0} \end{bmatrix}; \qquad (3)$$

2) given a set of motion vectors, the least squares fit to the spatiotemporal motion model is separable in time and space; and can be obtained by first finding the best fit to the velocities and accelerations of all the paths (according to equation 1), and then finding the least squares approximation to the spatiotemporal parameters according to equation 3.

3. MULTIPLE OBJECTS AND OCCLUSIONS

Whenever there is motion, there are regions which become visible and regions which are occluded. Occlusions make the task of building initial path estimates (on which the least squares fit of the previous section is performed) significantly difficult. If a pixel is occluded in a given frame, the associated motion vector will be unavailable, and an algorithm which tracks the motion of that pixel by following the cascade of motion vectors across the sequence will originate incorrect estimates.

The problem is avoided if one relies on a parametric approximation of the flow-field, such as the affine motion model. As long as the affine approximation is a reasonable one and occluded regions are small in comparison to the visible ones, a least squares affine fit is an equally good approximation to the motion of both occluded and visible areas[1]. Thus, given the knowledge that a point is associated with an object and the affine parameters that describe the motion of that object throughout the sequence, the motion of the point between any two frames is computable even if the point is occluded in frames between them.

An additional problem posed to a spatiotemporal video representation is the determination of the reference frame which will be used as a starting point for the spatiotemporal trajectories, and from which the remaining frames of the sequence will be reconstructed. Clearly, any frame will have occluded regions which will not be represented well if that frame is chosen as reference. Also, different portions of the scene may be captured with different spatial resolutions as the sequence progresses (e.g. if the camera "zooms in"), and one would like to chose the reference frame so that spatial resolution is fully exploited.

A layered representation [7] is an elegant solution to these problems. In this approach, each object is associated with a different image layer where the information about the object is accumulated as the scene progresses. If certain regions of the scene are captured in higher detail as the sequence progresses, the corresponding layer will have higher resolution in those regions. Since the layer accumulates information across the entire sequence, it will contain information that was not visible as a whole in any of the frames of the original sequence. Finally, the layered representation is a natural one when parametric motion representations, such as the discussed above, are required. For all these reasons, we use the layered representation to generate our reference images and bootstrap the spatiotemporal representation.

4. THE SPATIOTEMPORAL REPRESENTATION

In this section, we describe the algorithm that computes the reference images and motion parameters of the spatiotemporal representation.

4.1. Image layers

As mentioned above, we start by building a set of object layers using a procedure similar to that of [7]. For each framepair we compute the respective optic flow, using a standard "sum of squared differences" estimator [1]. We then use a procedure based on the expectation-maximization (EM) algorithm [2] to simultaneously compute the segmentation of each image-pair into the objects which compose it, and estimate the affine parameters for the motion of each object[2]. Next, we use a procedure similar to that proposed in [7] to integrate information across the image sequence and build a layer, or reference image, for each of the segmented objects. At this point, we have built the layered representation and are ready to estimate the spatiotemporal motion parameters.

4.2. Estimation of motion parameters

Since the least-squares fit to this model is separable in space and time, we start by finding the parameters of equation 1 for all the motion paths originated by a given object. For each point in the object's reference image, we first determine the corresponding trajectory across the sequence by following the cascade of motion vectors that starts at this point. Notice that, since we are relying on a parametric approximation of the optic flow, this process is reliable even

[1] Notice that this is true for any *parametric* representation, not a particular property of the affine approximation.

[2] The EM algorithm is a statistical tool for maximum-likelihood estimation from incomplete data that, when applied to motion estimation, results in an iterative procedure composed of two steps: the expectation step where, given a set of motion parameters, a segmentation mask is computed; and the maximization step where, given the segmentation mask, the motion parameters are updated in order to maximize the likelihood of the observed data [5].

when the point is occluded during a portion of the trajectory.

Once the trajectory is determined, we find its velocity and acceleration parameters (equation 1) using standard least-squares methods [4]. Defining Δt_i as the temporal distance between image i and the reference image, and $\Delta x_\kappa^{[i]}$ as the corresponding horizontal displacement of the point in the image plane, the least-squares estimates of the trajectory's horizontal[3] acceleration and velocity are given by

$$\hat{\mathbf{F}}_\kappa = \begin{bmatrix} \sum_i \Delta t_i^4 & \sum_i \Delta t_i^3 \\ \sum_i \Delta t_i^3 & \sum_i \Delta t_i^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_i \Delta t_i^2 \Delta x_\kappa^{[i]} \\ \sum_i \Delta t_i \Delta x_\kappa^{[i]} \end{bmatrix}, \quad (4)$$

where $\hat{\mathbf{F}}_\kappa = (\hat{a}_{\kappa_x}, \hat{v}_{\kappa_x})^T$.

Finally, given the value of the temporal velocity and acceleration estimates for all the points in the reference image, we find the set of spatiotemporal motion parameters that optimally satisfy equation 3 in the least-squares sense. Defining the vector of spatiotemporal motion parameters $\mathbf{A}_{\mathbf{xaff}} = (a_{xx}, a_{xy}, a_{x0}, v_{xx}, v_{xy}, v_{x0})^T$, the matrix of reference coordinates for each trajectory

$$\mathbf{P}_\kappa = \begin{bmatrix} x_\kappa^{[0]} & y_\kappa^{[0]} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_\kappa^{[0]} & y_\kappa^{[0]} & 1 \end{bmatrix},$$

and the regressor matrix $\mathbf{N} = (\mathbf{P}_1, \ldots, \mathbf{P}_p)^T$, the least squares estimate of the motion parameters is

$$\hat{\mathbf{A}}_{\mathbf{xaff}} = (\mathbf{N}^T\mathbf{N})^{-1}\mathbf{N}^T\hat{\mathbf{F}}. \quad (5)$$

The temporal and spatial least-squares fits are described in more detail in [6].

5. SIMULATION RESULTS

In order to demonstrate the frame-rate independence of the spatiotemporal representation, we used it to synthesize a video sequence at new frame rates. The left column of figure 1 presents two frames of the sequence, and the associated segmentation mask. The right column of the figure presents three frames of the sequence reconstructed from the spatiotemporal representation. The top and bottom images are the reconstruction of the images to their left, while the image in the center is temporally located between the other two and did not exist in the original sequence.

The reconstructed sequence is a good approximation of the original, and there is no significant difference in quality between the reconstructed frames which are temporally co-located with those in the original sequence and the new frame. There are some artifacts in the region of the tree branches, and on the border between the houses and the sky. These are mostly due to segmentation noise, in particular pixels which are assigned to different layers in successive frames. We are currently investigating more robust segmentation schemes where the spatiotemporal motion model is used to enforce temporal segmentation consistency.

When viewed in a display device, the synthetic sequence has a much finer motion rendition than the original. This is illustrated in figure 4, where spatiotemporal slices of both sequences are shown. These slices were created from the spatiotemporal volume associated with the sequence (obtained by stacking several consecutive frames) by cutting a 2D slice parallel to the horizontal axis. The higher resolution of the synthetic sequence is clear from the figure.

Notice that all the frames shown in the right column of figure 1 are obtained by warping the reference images. I.e. it is not necessary to fully reconstruct the two frames of the original sequence in order to synthesize a new frame between them.

6. CONCLUSIONS

In the digital communications world of ubiquitous networking and computational power, video production is moving from the studio to the home. This shift is, however, dificulted by current digital video representations (such as MPEG) which, being based on a highly temporally localized processing paradigm, are poorly suited for tasks involving the interactive manipulation of digital bitstreams. In this paper, we address this issue by proposing a new representation based on spatiotemporal objects, or image layers, which allows the decoder to define parameters such as display rate as variables of the decoding process, and moves away from the traditional rigid frame structure.

7. REFERENCES

[1] J. Barron, D. Fleet, and S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, vol. 12, 1994.

[2] A. Dempster, N. Laird, and D. Rubin. Maximum-likelihood from Incomplete Data via the EM Algorithm. *J. of the Royal Statistical Society*, B-39, 1977.

[3] N. Negroponte. *Being Digital*. Alfred A. Knopf, Inc, 1995.

[4] G. Strang. *Linear Algebra and its Applications*. Harcourt Brace Jovanovich, Inc., 1985.

[5] N. Vasconcelos and A. Lippman. EM-Based Motion Estimation and Segmentation. In preparation.

[6] N. Vasconcelos and A. Lippman. Spatiotemporal Model-Based Optic Flow Estimation. In *Proc. ICIP'95*, 1995.

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

[3] Here, we analyze only the least squares fit to the x component, the results are similar for the y component.

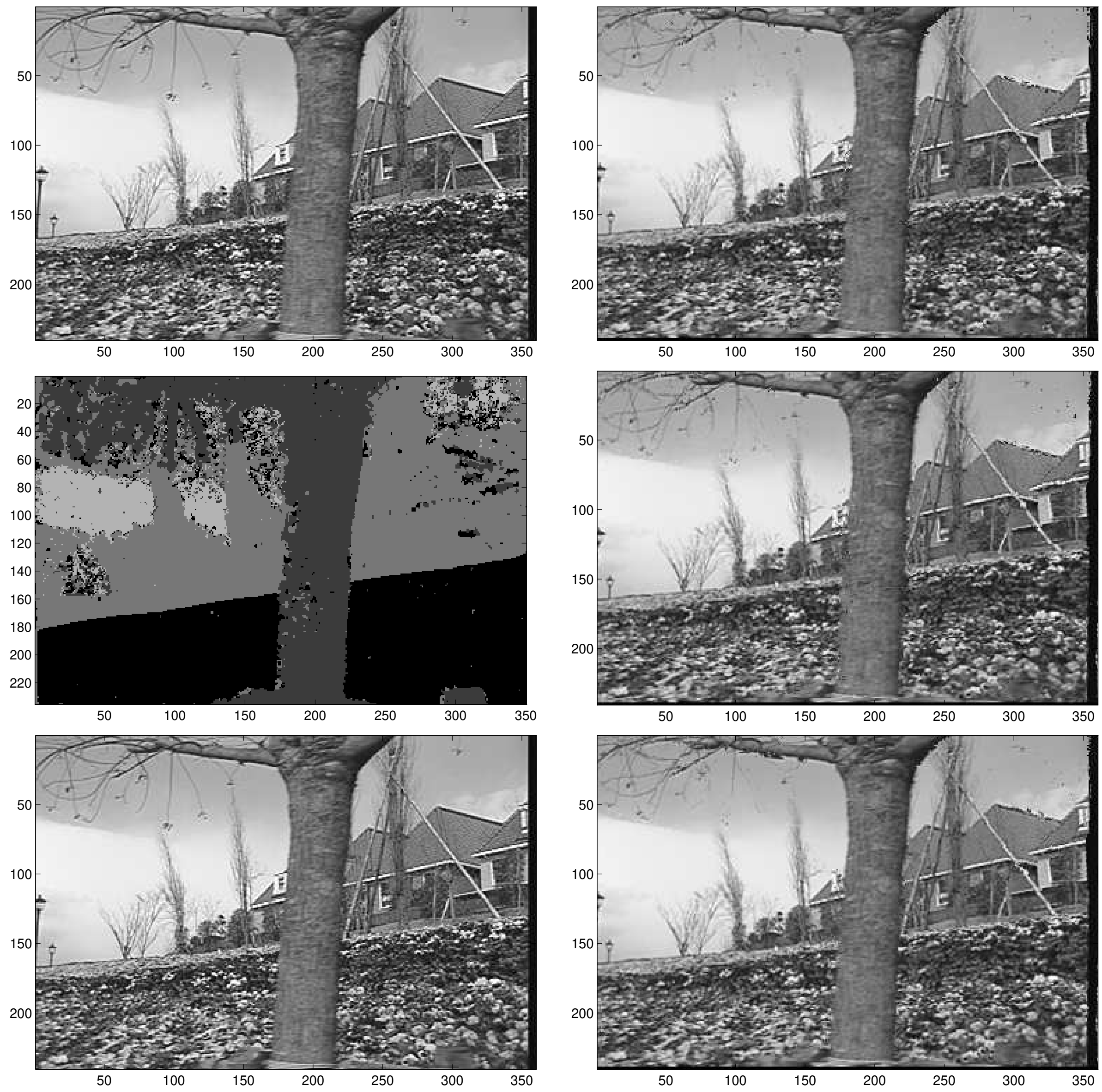

Figure 1: Left: Two consecutive frames of "flower garden" and corresponding segmentation mask. Right: Reconstructed frames. The middle image of the right column did not exist in the original sequence.

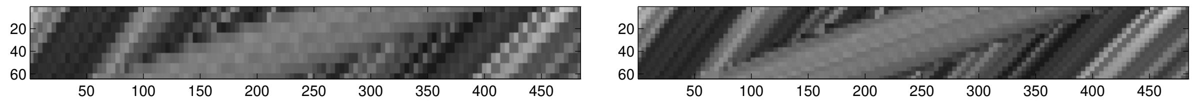

Figure 2: Spatiotemporal slices of the original (left) and synthetic (right) sequences. These slices are parallel to the horizontal axis, depicting the motion of the tree and part of the flower bed. For clarity, they were magnified by a factor of four in both dimensions.