TRAJECTORY BASED VIDEO OBJECT MANIPULATION

Rajvi Shah and P J Narayanan

CVIT, IIIT Hyderabad, India
rajvi.shah@research.iiit.ac.in, pjn@iiit.ac.in

ABSTRACT

We propose an object centric representation for easy and intuitive navigation and manipulation of videos. Object centric representation allows a user to directly access and process objects as basic video components. We demonstrate a trajectory based interface and example operations, which allow users to retime, reorder, remove or clone video objects in a 'click and drag' fashion. This interface is created by extracting object motion information from the video. We use object detection and tracking to obtain spatiotemporal video object tube. The corresponding object motion trajectories are represented in a 3D (x, y, t) grid. Users can navigate and manipulate video objects by scrubbing or manipulating corresponding trajectories. We show some example applications of proposed interface like object synchronization, saliency magnification, visual effects and composite video creation.

Index Terms— Motion based Video Representation, Interactive Video Composition, Object based Video Access

1. INTRODUCTION

The proliferation of digital cameras has caused a tremendous increase in user created images and videos. Manipulating captured images has become a home-user's task due to the availability of numerous easy to use photo editing tools. In comparison, video manipulation is still less common. Basic video editing platforms are easy to use, but these tools provide limited functionality such as split and merge videos, add captions or audio etc. Professional video editing platforms are rich in functionality, but these tools demand high technical expertise for use. Moreover, most of these tools model and represent videos as a collection of frames stacked against a timeline. Though this frame-time model is best suited for passive playback and media synchronization, it makes object centric manipulation of videos a laborious task. A naïve user usually gets discouraged by complex software controls and cumbersome processing.

The motivation of our work is to use computer vision techniques to improve usability of video manipulation interfaces. For a common user, it is more convenient to think of objects or activities as basic video entities and not the frames. We propose an object centric representation for easy and intuitive navigation and temporal manipulation of video objects. We model the video as a collection of spatiotemporal object volumes (object tubes) placed in a 3D grid as depicted in Fig. 1. With the advancement in computer vision techniques for object detection and tracking, creating such representation with little or no human intervention is now possible.

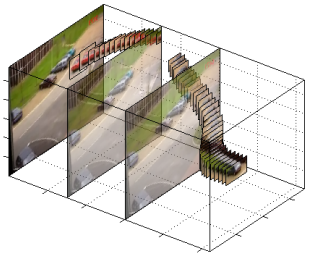


Fig. 1. Object tube model of a video

We extract motion information from the video as explained in Sec. 3 and represent object trajectories in a 3D interaction grid. Users can scrub, move or modify these trajectories to manipulate video objects interactively. Motion based video representations are used in other video navigation [4, 6, 7] and annotation [6] systems. The focus of these systems is on providing an in-scene Direct Manipulation interface and not on video content manipulation. Object motion information is also used in [11] to produce synopsis videos. This system combines motion information with spatiotemporal optimization constraints for automation. We, on the contrary, make object motion information available for user interaction. This allows the user to interactively produce multiple composite videos by modifying object trajectories in different ways.

Proposed representation allows interaction and manipulation at object-level. This representation typically works for long shot videos. Such video shots are captured from sufficient distance from the object so as to put the entire object and its activity in relation to the background. For example, surveillance videos, art performance videos, sports videos etc.

In later sections, we discuss a prototype interface and associated operations. We show that using these simple 'click and drag' operations a user can navigate, retime, reorder, remove or clone video objects. We demonstrate a few potential applications with example scenarios and conclude with a discussion of future scope.

2. RELATED WORK

Motion based interfaces for direct video navigation have been proposed before [4, 6, 7]. These interfaces allow users to navigate a video by scrubbing trajectories of active regions or objects on video frame itself. For motion representation, these systems use SIFT feature flow [4], dense optical flow [6] or object detection and tracking techniques [7]. The focus of these systems was to provide a better video playback interface and not to change or manipulate the actual content of the video. These systems stress on direct manipulation interface for more natural browsing experience.

Another motion based system [5] was proposed for video navigation, annotation and some manipulation tasks. This system also focuses on a direct manipulation interface and uses a dense motion representation [12] for pixel-level interaction. Though this interface allows object synchronization for desired still frame composition, it does not allow creation of a retimed video.

We focus on navigation as well as content manipulation. In our interface, scrubbing and other operations are performed in a separate 3D grid and not in video window itself. We do not stress on direct manipulation because we believe that a 3D representation is more natural for temporal manipulations. A 3D representation gives insight into object's occupancy in the pixel space while retaining the timeline for synchronization tasks. This enables the users not only to navigate and create desired stills but also to produce various temporal effects and create composite videos interactively. We compute and represent object-level motion which works well for object-level interaction in long shot videos.

3. PRE-PROCESSING

Proposed interface is built using object motion information. We pre-process the video to build this representation. Pre-processing includes detecting moving objects, tracking associated regions and constructing a constant background image.

3.1. Automatic Extraction

Object Detection: In a fixed camera environment, we use a background subtraction method proposed by Li et al. [8] which works well for both indoor and outdoor videos. Background subtraction algorithm labels every pixel in a frame either foreground or background. A connected-component test is run on these binary frames to group foreground pixels as plausible object bounding boxes. Semi-automatic matting techniques can be used for high-quality articulated segmentation with additional complexity[13].

If objects in the video are fixed to be of a single category, for example people, than the background subtraction based detection can be replaced by a feature based object detector like [3].

Object Tracking: After detecting the plausible moving objects, object position and size are tracked in every frame. A number of object tracking algorithms are proposed in computer vision literature [14]. We use a hybrid-tracker as described in [1]. This tracking method performs a simple connected-component tracking using Kalman filtering when objects in a video are well separated. When Kalman filter's prediction suggests a possible overlap of objects in next frame, a reliable Mean-shift tracker is used [2]. We chose region tracking over compute intensive dense motion tracking as our representation aims at object-level interaction which does not benefit much from dense motion information.

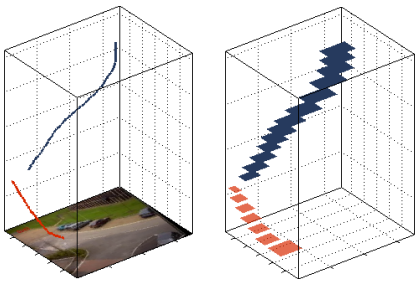
Background Image Construction: We model the video as a collection of spatiotemporal object tubes placed in a constant surrounding and allow temporal modifications of these tubes for new video creation. Moving these tubes to an earlier or later time will create holes in original spatiotemporal segments. The binary labeled video from object detection stage is used to fill a constant background image.

3.2. Manual Annotation

Algorithms for object detection and tracking mentioned earlier have been found to work reasonably well under most scenarios. In segments, where these algorithms do not produce perfect labeling, we allow the user to manually mark or correct the required bounding boxes in a few key frames and interpolate the results for intermediate frames.

4. INTERACTIVE OPERATIONS

A prototype interface is built using object motion trajectories. In this interface, object trajectories and their corresponding spatiotemporal occupancy are represented in 3D grids as shown in the Fig. 2. We call these grids interaction grid and visualization grid respectively. Users can perform navigation and manipulation operations in the interaction grid and simultaneously visualize spatiotemporal occupancy of objects in the free viewpoint visualization grid. We discuss the supported operations for both navigation and manipulation mode in detail.



(a) Interaction Grid (b) Visualization Grid

Fig. 2. Prototype Interface

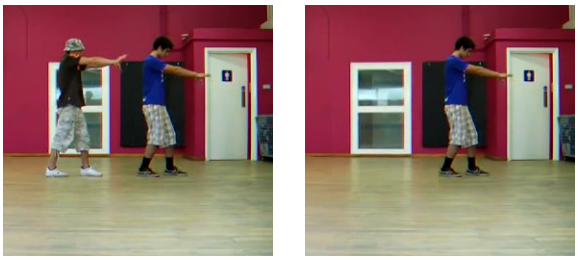


Fig. 3. Video mode and Object mode Navigation

4.1. Navigation

A user can control video navigation by scrubbing object trajectories with mouse. We provide two modes of navigation, Simple Video Navigation and Single Object Navigation. We also provide a *WYSIWYG* mode of creating videos in which users can create new videos the way they browse it.

Simple Video Navigation: In this mode, video content is not altered but the video playback is controlled by mouse position on the object trajectories. This mode is similar to video navigation interfaces discussed earlier in Sec. 2.

Single Object Navigation: In this mode, only the active trajectory object is displayed on the constant background still of the scene. Hence, user's scrubbing action results in motion of only a single, currently active object, replacing the other moving objects by constant background.

We provide a record option to record user's navigation actions and use it simultaneously to create a new video. This mode allows users to create various retiming and reordering effects in video just by scrubbing the object trajectories at desired speed and in desired order.

Fig. 3 shows video frames for Simple Video Navigation Mode and Single Object Navigation Mode for the same mouse position on trajectory of the dancer in right. Frame in left is actual video frame, whereas frame in right is generated by superimposing active object segment on pre-computed background for object navigation mode.

4.2. Temporal Manipulation

User can create various object centric temporal effects using simple and intuitive click and drag operations. Manipulation interface includes four basic operations - move, copy, erase and modify.

Move: A user can drag and move the object trajectories along time-axis to shift the object tubes to an earlier or later time in the video effectively shifting the object's lifetime.

Copy: A user can copy a trajectory and paste it with a shift in time to create multiple instances of the same object.

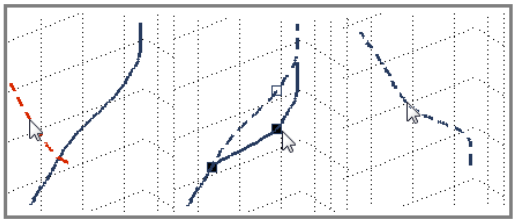


Fig. 4. Move, Shrink and Invert Operations

Erase: A user can erase a trajectory by moving the mouse on trajectory in erase mode, effectively erasing the object from specified time segment.

Modify: A user can select a segment of any trajectory and shrink or stretch it with mouse drag along time-axis to speed up or slow down the selected object's motion. Alternatively the user can select a single point on the trajectory and stretch it to stop the moving object for specified time. A third variation of this operation allows the user to select a trajectory and invert it with a left click to reverse the object's motion. Fig. 4 shows move, shrink and invert operations being performed on object trajectories.

At any point of time, user can use the visualization grid to observe spatial occupancy of objects. We highlight all probable regions of object overlap by changing the object-tube color.

Move, modify and erase operations might create seams at object boundaries due to illumination changes. Blending techniques like [10] can be used to avoid visible artifacts. Temporal speedup can be achieved by skipping frames but temporal magnification requires up-sampling for smooth playback. Nominal temporal up-sampling can be achieved by frame interpolation techniques. At higher rates more sophisticated techniques like [9] can be used to produce better results. We do not further elaborate on required post-processing for these operations as it is well covered in literature and not the focus of this work.

5. APPLICATIONS

Proposed interface can be useful in many scenarios. We discuss four potential applications and example scenarios. (Demo video can be found at `http://researchweb.iiit.ac.in/~rajvi.shah/vnm/`).

Object Centric Navigation: We introduced Single Object Navigation in Sec. 4. As this mode freezes or removes all moving objects, other than the one being currently browsed, it allows the viewer to concentrate on a single activity. For example, this mode can be used by a sports instructor to browse only a particular player's actions without being distracted by other activities.

Fig. 5. Stills from Synopsis(L) and Cloning Effect(R) Videos

Saliency Magnification: Using shrink and stretch operations introduced in Sec. 4, a user can magnify salient activities. For example, while editing a dance video, a user can shrink trajectory segments related to trivial dance movements or stretch segments of significant and expressive movements creating focus on important movements.

Visual Effects: Using a combination of copy and erase operations, a user can easily produce clone effect and stroboscopic effect. Move and Modify operations enable a user to retime and reorder objects. This can be used to synchronize objects or produce desired time lags between objects.

Interactive Video Synopsis: Rav-Acha et al. [11] proposed a spatiotemporal occupancy optimization based solution for creating video synopsis automatically. Our interface can be useful to produce such synopsis videos interactively.
Fig. 5 shows stills from a synopsis video and a clone effect video. In original video, the object tubes of cars are separated in time (See Fig. 1). We move the red car's trajectory to a later time to create a synopsis in which both the cars move together. We copy the blue car's trajectory with a shift in time to create a clone car.

6. CONCLUSION AND FUTURE WORK

We use object motion data to ease object centric video object navigation and manipulation tasks for a common user. A 3D grid based interaction and visualization interface is proposed. This representation enables a user to retime, reorder and navigate video objects in a more convenient and intuitive way. Though our representation is not generic enough to model any video, it is a very natural representation to manipulate long shot videos like surveillance, stage performance, sports etc. Proposed approach uses background subtraction for moving object detection and constant background construction. This limits the application of interface to fixed-camera videos. Currently, we are trying to extend our approach to support simple camera motion. Another useful extension of this work is to estimate the complexity of object motion and represent it visually to aid a user focus on probably more important video segments. We believe that augmenting video context and motion cues with user interface can significantly improve the usability of video manipulation tools. Proposed interface is one such step to achieve that overall goal. We believe that the fidelity and popularity of such interfaces will increase with the progress in computer vision and video processing techniques.

References

[1] T. P. Chen, H. Haussecker, A. Bovyrin, R. Belenov, K. Rodyushkin, A. Kuranov, and V. Eruhimov. Computer vision workload analysis: Case study of video surveillance systems. *Intel Technology Journal*, 9(2): 109–118, 2005.

[2] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proceedings IEEE CVPR 2000*, volume 2.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings IEEE CVPR 2005*, volume 1, 2005.

[4] P. Dragicevic, G. Ramos, J. Bibliowitcz, D. Nowrouzezahrai, R. Balakrishnan, and K. Singh. Video browsing by direct manipulation. In *Proceedings ACM CHI 2008*.

[5] D. B. Goldman, C. Gonterman, B. Curless, D. Salesin, and S. M. Seitz. Video object annotation, navigation, and composition. In *Proceedings UIST 2008*.

[6] T. Karrer, M. Weiss, E. Lee, and J. Borchers. Dragon: A direct manipulation interface for frame-accurate in-scene video navigation. In *Proceedings ACM CHI 2008*.

[7] D. Kimber, T. Dunnigan, A. Girgensohn, F. M. S. III, T. Turner, and T. Yang. Trailblazing: Video playback control by direct object manipulation. In *Proceedings IEEE ICME 2007*.

[8] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian. Foreground object detection from videos containing complex background. In *Proceedings ACM MULTIMEDIA 2003*.

[9] D. Mahajan, F.-C. Huang, W. Matusik, R. Ramamoorthi, and P. Belhumeur. Moving gradients: a path-based method for plausible image interpolation. *ACM Trans. Graph.*, 28, 2009.

[10] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*.

[11] A. Rav-Acha, Y. Pritch, and S. Peleg. Making a long video short: Dynamic video synopsis. In *Proceedings IEEE CVPR 2006*.

[12] P. Sand and S. J. Teller. Particle video: Long-range motion estimation using point trajectories. *International Journal of Computer Vision*, 80(1).

[13] J. Wang and M. F. Cohen. Image and video matting: a survey. *Found. Trends. Comput. Graph. Vis.*, 3, January 2007.

[14] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 38, 2006.