Automatic Heterogeneous Video Summarization in Temporal Profile

Hongyuan Cai, Jiang Yu Zheng
Indiana University Purdue University Indianapolis
hocai@cs.iupui.edu, jzheng@cs.iupui.edu

Abstract

Numerous videos are uploaded on video websites; most of them employ several kinds of camera operations for expanding FOV, emphasizing events, and expressing cinematic effect. To generate a profile of heterogeneous types of videos, an automatic video profiling method has been proposed to include both spatial and temporal information in a 2D image scroll. In this paper, we propose a uniformed scheme to segment video clips and sections, compute major optical flow and convergence factor, and then sample video volume across the major flow for the profiles. A video profile shows an intrinsic scene space less influenced by the camera ego-motion. It is also a fine-grained temporal representation that can be displayed in a video track to guide the access to frames, help video editing and visual archiving of environment, video retrieval, and browsing.

1. Introduction

Current video indexing uses key frames [1,11] from video clips/shots as well as their collections as video storyboard or tapestry [2,3]. However, the key frame is a sparse representation that picks only a discrete moment in a clip. Video mosaicing extends the *field of view* (FOV) to *multi-perspective view* [4]. Panoramic images are generated in a larger spatial domain by stitching panning video frames [4-10]. These methods stitch regions from different frames into a summary image, but have drawbacks as follows: (1) Camera motions are limited to static and pan. If the motion parallax appears as in a translation video, the result is not guaranteed (possible only for scenes with single depth layer, homogeneous texture or color [5]). (2) Some summary aims at visualizing actions by duplicating a target in an image [6,8,10]. It becomes cluttered if the video clip lasts for long or has multiple targets. (3) Lack of temporal order; it can index to a clip rather than to a frame for video editing. (4) The background matching and foreground segmentation are not robust in general for complex and dynamic scenes. Instantaneous events and non-rigid shapes such as fire, smoke, water, and so on may cause more problems. On the contrary, the temporal video index by collecting pixel data from a line in each frame during the camera rotation [14] and translation [12] strictly reflects temporal information. The line can be fixed [13,24] or dynamic [15] in the frames. The shortcomings are: (1) the generated views have different projections from the perspective projection, and (2) it requires deshaking in the original video or in the generated image [16,17] for a non-smooth camera motion.

Recently, Cai and Zheng [18,19] created a new speatio-temporal 2D profile from raw video for preview, given the fact that most video clips in databases have stable camera motions rather than random shaking. It contains one axis as the timeline lasting with the video, and the other indicates a spatial dimension in the video frame. Different from [3] that is a collection of icons from frames, the scenes in the profile are continuous and temporally scalable in the display, despite of the employed projection different from perspective projection. More general than [15], the profile can be extracted from all types of camera motions including rotating, translating, zooming, static cameras and their combinations. It contains most of the scenes in the video such that viewers know what is in the video by glancing at it. This paper addresses the automatic segmentation of sections that correspond to smooth camera operations, uniformed video sampling for profile, and shape normalization in the profile, as shown in the flow chart in Fig. 1.

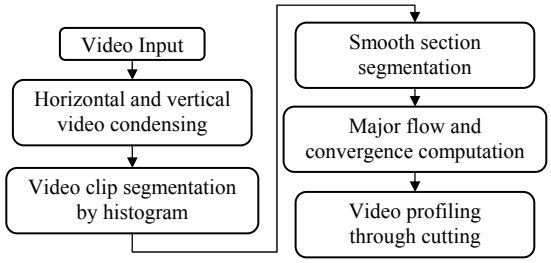


Figure 1.The brief flow chart of the system.

In the following, we propose an efficient method to automatically segment video clips by extracting smooth camera operations in Sections 2 and 3. Section 4 introduces a general framework of video sampling/ cutting for profiles. Section 5 discusses a profile normalization method for a better shape mode, followed with experimental results and conclusion.

2. Video Volume Condensing

Video clips with basic camera motions [18] and their composite motions [19] have been profiled. Here, we use condensed images to perform the task of automatic profiling for video database. We do not use the approach of computing optical flow $u(x,y,t)=(u_x,u_y,u_t)$ [22] and then summarizing the flow for a global direction using PCA, because of its high cost for processing large video databases and unstable results on deformable scenes, water, fire, and scenes without many features. We produce a flow graph that is one of the condensed images [16,17] to show reliable scene motions as traces across frames, and this achieves efficiency in obtaining the global motion. For simplicity, we collect the condensed images (Fig. 2) along the x and y directions in the frame as

$$C_y(t,x)=\frac{1}{h}\sum_{y\in C} I(x,y,t) \qquad C_x(t,y)=\frac{1}{w}\sum_{x\in C} I(x,y,t) \qquad (1)$$

For a video volume (clip) obtained from a smooth camera ego-motion (a single operation) or a directional motion of target crowds, we can specify a *major flow* component $v \in R^3$ in it. Denoting such a video segment as *section* by C, the major flow in C is defined as

$$v=(v_x,v_y,v_t)=\frac{1}{N}\sum_{x,y,t\in C} u(x,y,t) \qquad (2)$$

where $\|u(x,y,t)\|=1$ and N is the number of points in the clip. Its projections in y, x, and t directions are $V_x=(v_t,v_x)$, $V_y=(v_t,v_y)$, and $M=(v_x,v_y)$, respectively. M shows the direction in the frame, if the scenes have some common motion under a smooth camera motion. The V_x and V_y accumulated in C_y and C_x can sufficiently approximate the true major motion from the optical flow, and is straightforward and reliable.

3. Segmenting Camera Operations

The video clip is first segmented using traditional histogram differentiation [23]. We further separate a clip to sections, each with a monotonic camera operation/motion. In the condensed images, we explore the flow characteristics by examining traces in the condensed images; one with more motion traces than the other is called *flow graph* (Fig. 2(a)). The trace direction contains the camera motion information.

We take the partial derivatives ∇C_y (and ∇C_x) in the

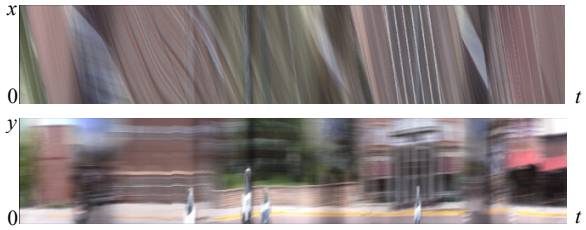


Figure 2. Flow graph and shape oriented condensed image

condensed images and compute the gradient directions at high contrast points. The vectors $e_x(t)$ (and $e_y(t)$) orthogonal to the gradients provide trace directions, and we normalize them to unit vectors. The major flow directions $V_x(t)$ (and $V_y(t)$) at each t are averaged from trace vectors in the condensed images.

At the same time, the variance $\sigma_x(t)$ of $e_x(t)$ are also computed over time in the condensed images. Figure 3 shows the results in the vertically condensed C_y. The value of $\sigma_x(t)$ suggests a camera zooming that allows us to separate diversified motion from directional motion. A distinct $v_x(t)$ further indicates a directional motion. The variance is computed for C_x as well.

With the sequences of $v_x(t)$ and $\sigma_x(t)$ as well as $v_y(t)$ and $\sigma_y(t)$, a smooth motion can be segmented in either condensed image as follows.

	small $\sigma_x(t)$	*large* $\sigma_x(t)$
$\|v_x(t)\|$ *small*	Static camera and scene	Zooming
$\|v_x(t)\|$ *large*	Camera pan	Pan+zoom, translation

We finally use a median filter to remove short segments (as noise) and group them into larger sections. Small variations in a large section are from the camera shaking and foreground target motion.

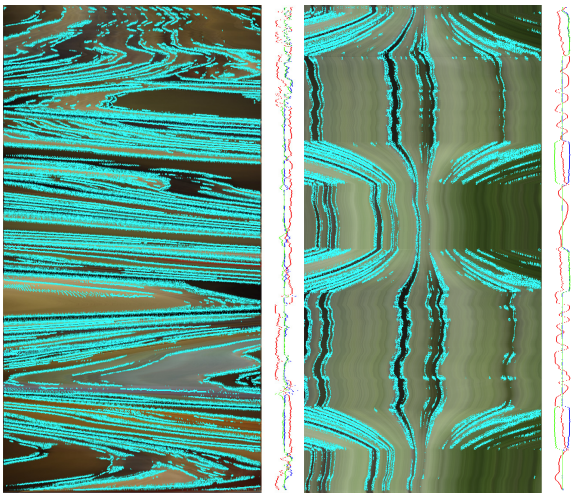


Figure 3. Computing major flow and convergence factor. High gradient positions on traces are marked in cyan in flow graphs for motion estimation. The value of $v_x(t)$, $\sigma_x(t)$ and $\kappa(t)$ are displayed in red, blue, and green curves along the time axis vertically (axis at the center).

In an entire section, we calculate convergence factor $\kappa(t)$ of the flow, which is similar as $\sigma_x(t)$ but indicates convergence or divergence effect [20]. In C_y, we compute $V_x=[\Sigma_{i,t}\, e(t)]/n$, where n is the number of high-contrast trace points. We obtain v_x by normalizing V_x so that $v_t=1$. The computation is also applied to C_x and the major flow components are obtained in condensed images (Fig. 4). Moreover, a median point $p(t_m,x_m)$ is located using all the trace points in the section. The convergence factor is defined as

$$\kappa(t) = \sum_i (e_i(t,x_i) - v_x) \times sign\left[x_i - (x_m + v_x(t-t_m))\right] \quad (3)$$

$$\begin{cases} < 0 & converge : zoom-out \\ > 0 & diverge : zoom-in \end{cases}$$

where e_i, i =1,2,… is the direction of trace point (t,x_i). The result of $\kappa(t)$ is shown as green curves in Fig. 3.

4. Profiling Video Volume to Digest

With successfully segmented sections, we perform global sampling to obtain their profiles. Different camera motions have different profile cutting strategies. The profile should not only have the advantage to include more scenes as a panorama does for browsing, but also have a dimension of time for indexing to a particular frame from any clicked/ selected position. To facilitate fast video browsing and transmission, we may sacrifice some image properties of perspective projection. In details, from the 3D volume of a video section $I(x,y,t)$, we map scenes to a 2D profile, $P(t,y)$ (or $P(t,x)$), to guarantee a single occurrence of scenes except occlusion. The profile reveals most scenes in the video for retrieval and display. That is, from $P(t,y)$, a video frame can be indexed through time t. Instead of composing segmented background and foreground in $I(x,y,t)$ for a mosaic, we use a moving pixel line L_y (or L_x) to sample the volume for a compact image belt. The sampled slice in the volume should cut against $v(t)$, rather than aligning with it to yield motion traces. A global cutting method is designed as follows:

- The sampling line is set parallel to an axis of image frame, thus parallel to many structure lines in the scene, in order to keep the shape integrity, since $P(t,x)$ (or $P(t,y)$) is displayed in a regular window. The sampling line more orthogonal to the major flow is selected, i.e., we select L_y if $v_x \geq v_y$, or L_x, otherwise. The flow graph is then C_y or C_x.
- After aligning the sampling line, say L_y, it is moved in the volume along a diagonal trajectory $x(t)$ in the flow graph to intersect the major flow v (i.e., V_x), i.e.,

$$p(t,y) = sampling[I(x,y,t) \mid x(t)] \quad (4)$$

$P(t,y)$ yields sharper scenes with diagonal cutting of flow, and maps all the scenes stably visible in the video into $P(t,y)$ with a coherent temporal scale. It does not cut the video clip back and forth temporally

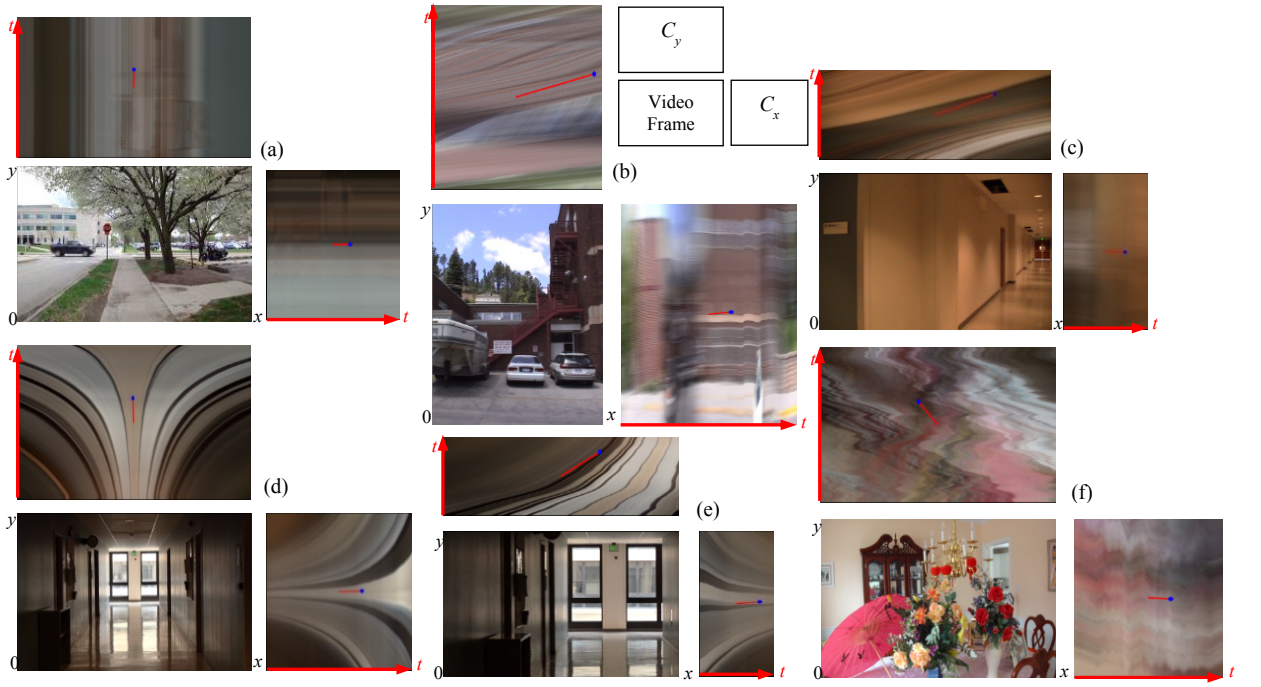


Figure 4. The major flow directions of video sections detected in condensed images C_y and C_x. The computed components V_x and V_y of the major flow vectors are plotted in red arrows with blue tips. (a) Static camera, (b) camera translation (vertically), (c) pan, (d) zoom, (e) pan plus zoom, and (f) orbiting motion around target [21] are examined.

with size-varied patches [9].

– If the major flow is accompanied with convergence or divergence effect due to a zooming operation, we bend the sampling curve $x(t)$ towards the enlarged frame in the clip so as to prevent scene blurring and recurrence in the profile. The convergence factor determines a curved or straight trajectory, where a Bezier curve is used as the controlled curve.

5. Temporal Mode and Shape Mode

The diagonal profile cutting is determined by the length of the section, which yields shape distortions in aspect ratio. For a more pleasant experience to browse profile, we further introduce a *shape mode* of the profile in addition to the *temporal mode* that strictly follows time code.

To preserve the shape information, we need to resize the resulting profile according to the angle between the cutting path and the major flow direction. As shown in Fig. 5, if the local cutting length l can be scaled to the same length as in the image denoted by L, the shape is preserved better in the profile. In the triangle of Fig. 5 formed by cutting segment l, its corresponding length L in the image, and the major flow direction V, we have

$$L(t) = l(t)\frac{\sin\alpha(t)}{\sin(\alpha(t)+\beta)} \tag{5}$$

where $\alpha(t)$ is the angle between l and $V_x(t)$, and β is the angle between l and the image plane. Both are known angles computed for cutting. One result is displayed in Fig. 6.

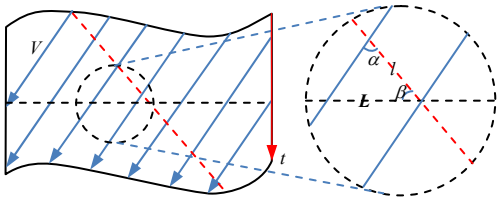


Figure 5. Obtaining shape mode of profile. Red line is the diagonal cutting direction in the flow graph.

6. Experiments

As a video is read in, it is scanned for color condensing. With the two condensed images, we segment video to clips and then to sections with smooth camera motions as in Fig. 3. A relatively larger variance suggests a zoom operation. The major flow from accumulated trace angles is detected and the flow graph is identified for cutting. A median filter of size $3\sigma_x$ is applied to major flow to remove the noises caused by foreground moving objects. We have examined our method on hours of videos in their profiling. The speed for profiling is significantly faster than mosaicing. Some results are shown in Figs. 7-10.

7. Conclusion

This work proposed the automatic profiling of video volumes for video digests. Based on the analysis of camera motion and global flow, a uniformed algorithm is developed to profile videos for all types of camera actions. The global motion of camera is estimated efficiently with two condensed images, and the generated 2D profiles containing both temporal and spatial information. The profiling method is global and is more flexible and faster than mosaicing methods. It can automatically map a video database to facilitate video browsing and editing. One can preview a profile before checking video itself in video browsing.

References

[1] B. Janvier, et al: Information-theoretic temporal segmentation of video and applications: multiscale keyframes selection and shot boundaries detection. *Multimedia Tools and Applications* 30, 273-288, 2006.

[2] D. Goldman, et al, Schematic storyboarding for video visualization and editing. *SIGGRAPH* 25, 862-871, 2006.

[3] C. Barnes, et al.: Video tapestries with continuous temporal zoom. *ACM SIGGRAPH* 29(89:1-89:9), 2010.

[4] A. Aner, J. Kender, Video summaries through mosaic-based shot and scene clustering, *ECCV* 45-49, 2002.

[5] A. Agarwala, et al: Photographing long scenes with multi-viewpoint panoramas. ACM *SIGGRAPH*, 853,2006.

[6] F. Liu, Y. Hu, M. Gleicher, Discovering panoramas in web videos. *ACM Multimedia,* 329-338, 2008.

[7] Rav-Acha, et al., Dynamosaicing: Mosaicing of dynamic scenes. *IEEE PAMI* 29, 1789-1801, 2007.

[8] C. Correa, K. Ma, Dynamic video narratives. *ACM SIGGRAPH*, 29(88), 2010.

[9] Y. Wexler, D. Simakov, Space-time scene manifolds. *ICCV*, 858-863, 2005.

[10] Y. Pritch, et al, Nonchronological video synopsis and indexing. *IEEE PAMI*, 30, 1971-1984, 2008.

[11] M. S. Kolekar, S. Sengupta, Semantic Concept mining in cricket videos for automated highlight generation, *Multimedia Tools and Applications*, 2010.

[12] J. Y. Zheng, Digital route panoramas. *IEEE Multimedia* 10(57-67), 2003.

[13] J. Y. Zheng, S. Sinha, Line cameras for monitoring and surveillance sensor networks. *ACM MM*, 433-442, 2007.

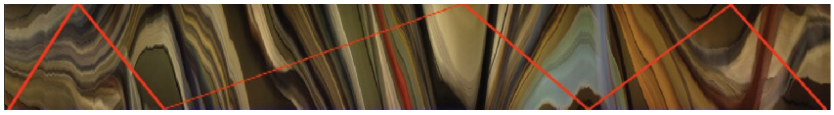
[14] S. Peleg, et al, Mosaicing on adaptive manifolds. *IEEE PAMI*, 22, 1144-1154, 2000.

[15] A. Zomet, et al: Mosaicing new views: The crossed-slits projection. *IEEE PAMI*, 25, 741-754, 2003.

[16] J. Y. Zheng, Y. Bhupalam, H. Tanaka, Understanding vehicle motion via spatial integration of intensities. 19th *ICPR*, 1-5, 2008.

[17] H. Cai, J.Y. Zheng, H. Tanaka, Acquiring shaking-free route panorama by stationary blurring. *IEEE ICIP*, 921-924, 2010.

[18] J. Y. Zheng, H. Cai, K. Prabhakar, Profiling video to visual track for preview. *IEEE ICME*, 1-6, 2011.
[19] H. Cai, J. Y. Zheng, Video anatomy: cutting video volume for profile. *ACM Multimedia*, 1065-1068, 2011.
[20] R. Nelson, J. Aloimonos, Obstacle avoidance using flow field divergence. *IEEE PAMI*, 11,1102-1106, 1989.
[21] J. Y. Zheng, Y. Fukagawa, T. Ohtsuka, N. Abe, Acquiring 3D models from rotation and highlight,12th *ICPR* , 331-336, 1994.
[22] S. Ali, M. Shah, Floor fields for tracking in high density crowd scenes. *ECCV* 2, 1-14, 2008.
[23] Y. Murai, H. Fujiyoshi, Shot boundary detection using co-occurrence of global motion in video stream, *ICPR* 1-4, 2008.


t (a) Flow graph and sampling trajectories


t (b) Temporal mode profile


(c) Shape mode of profile is longer than temporal one. The video contains forward walking accompanied with left and right pans.

Figure 6. Local scaling from temporal mode to shape mode. The video is from a wearable camera at a conference site.


t (a) Skateboarding, tilting visible.


t (b) Car race scaled temporally.


t (c) Basketball game

Figure 7. Profiles for back-and-forth panning (different from spatial mosaic with fixed length). Minor flow is visible [19].


0 *t* a key frame

Figure 8. Large camera motion following crowded players. The profile from consecutive pans shows game progress in the temporal domain. A key frame given contains true shape but does not reveal the context of entire clip.

Figure 9. Consecutive profiles of a concert video with simultaneous zoom and pan. (Top) Vertically condensed flow graphs. (Lower) Profile cut from the trajectory in the flow graph above. The profile is temporally scaled up.


0 *t*

Figure 10. A translation video taken in a moving bus. Minor flow reduction is applied after profiling acquisition [16,17].