Fast Non-Linear Video Synopsis

Alparslan YILDIZ, Adem OZGUR and Yusuf Sinan AKGUL
{yildiz, akgul}@bilmuh.gyte.edu.tr, aozgur@gyte.edu.tr

GIT Vision Lab - http://vision.gyte.edu.tr
Gebze Institute of Technology
Kocaeli, Turkey

***Abstract*—This paper presents a very efficient method for extracting the non-linear summary of a video sequence by synthesizing new summary frames from a number of original frames. The non-linear summary of a video sequence is fundamentally different from the classical video summarizing techniques which discard full frames. The high efficiency of the method is due to the employment of dynamic programming to find the space-time surfaces in the video volume that have lesser motion information. We project the video volume onto a plane orthogonal to one of its axes and find a minimum cost path on the time derivative of the projected image. Back-projecting the minimum cost path down to the video volume gives the space time surface to be discarded. Applying this process several times results in the summarized video. One of the major contributions of our method is that it can work on real-time video sequences with a small memory requirement.**

We demonstrated our method on several examples which are available at `http://vision.gyte.edu.tr/projects.php?id=5`**.**

I. Introduction

With the decreasing prices of digital video equipment, video recording of activity is becoming very common in many areas such as surveillance, security, and activity analysis. The recorded video usually takes up large spaces and the retrieval and browsing of the video is usually a very time consuming task. An elegant solution to these problems would be storing and browsing the summaries of videos by discarding sections of videos with no activity, which would be very effective because most of the video sections contain no activity information at all.

The number of research proposals for summarizing videos is getting larger everyday. The most popular video summary method is simply discarding frames with least activity [1], [8], [9], [10], [12] but this simple method cannot compress a video shorter than number of possible key frames and may have to discard frames with much activity when forced. Another method [2], [6], [7], [11] is to keep shorter sub-sequences with much activity of the original video. While this method represents the original video better, resulting summary of the video is longer compared to the method that discards frames. Hierarchical video summary and browsing techniques [3] are more effective but they require parsing of the video which is expensive and not always applicable.

The common feature among the above systems is to use or discard complete frames from the original video. If the activity covers only a small section of the video frame, then the whole frame is considered involving the activity. Although this approach keeps the chronological ordering of the objects in the video sequence, it cannot synthesize new frames using the activity from different frames. Such non-linear frame synthesis technique would produce much more compact video synopsis.

A more sophisticated idea that produces much more compact synopsis would be considering the video as a volume build by consecutive frames and trying to discard volumetric regions with low activity. Fig. 1 shows the effect of the volumetric video synopsis. The green and red activity blocks of different time intervals and different image regions are combined into a singe volume resulting in the summarized video where both green and red activity blocks can be viewed simultaneously. Fig. 2 shows the synopsis effect on a video sequence from our system.

A non-linear video synopsis approach is given by Rav-Acha *et al.* [4] who represented the video synopsis as an energy minimization over the whole volume. This method lets objects move on the time axis independently to compress the activity from different time intervals into a very small time volume. Furthermore, chronology of a single pixel value is allowed to change, meaning that events of different time steps for the same region of the video image can be collated in any order. In the final summarized video, a single frame is most likely composed of activity from different frames of the original video. While this method may seem like a good solution for a compact video synopsis, the overall energy minimization is very complex and is not suitable for real-time applications.

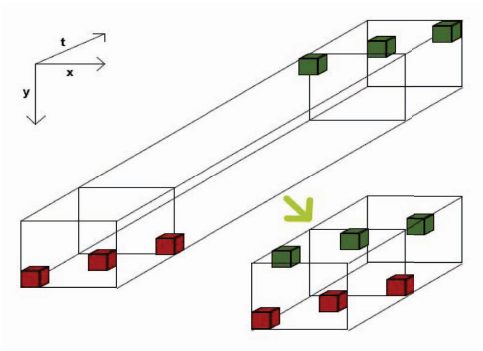


Fig. 1. The effect of volumetric video synopsis. Green and red activity blocks of different time intervals appear in the same time interval of the synopsis video.

(a) Sample frames from an input video sequence of 241 frames

(b) Sample frames from the corresponding synopsis video of 128 frames

Fig. 2. In video synopsis objects from different time intervals appear together to represent a compact summary of the original video. An input video of 241 frames given in (a) is summarized into 128 frames(50%) to get the synopsis video given in (b). Frame numbers are shown on upper left corner of the frames. The same video can be summarized into only about 230 frames by classical frame discarding methods.

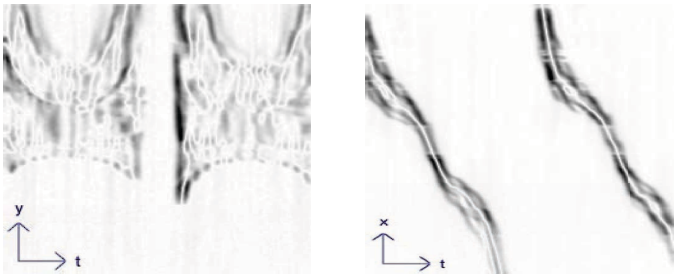
This paper presents a novel real-time system for the non-linear video synopsis generation. The system employs the non-linear image scaling method[5] which scales images while considering the image content such as object boundaries. This is achieved by computing an energy image from the original image such that pixels with more information have higher energy values. The method is very efficient due to the employment of dynamic programming to find a minimum energy path over the energy image. This path can be discarded without losing significant image information because it includes only pixels with minimal information. Applying this method several times results in shrinking the image while preserving the pixels with information.

Direct application of the non-linear image scaling to video synopsis raises the problem of finding minimum cost surface (instead of minimum cost path) in the space time volume of the video. Finding minimum cost surface with dynamic programming is not practical because the algorithm becomes exponentially complex for three dimensional volumes.

We introduce a new approach to solve the video synopsis problem using dynamic programming on the projections of video volumes rather than the volumes themselves. Since the projections of video volumes are 2D images, dynamic programming can run on them with minor modifications. Although the projected videos lose some activity information in the scenes, we observed that producing video synopsis on the projected videos are perfectly feasible for real life situations. The presented approach is simple but is very effective and implementation is not difficult. It does not require any complex function optimizations and is considerably faster than the alternative methods. Another major advantage of our approach is that video synopsis with synthesizing new frames in real-time is possible with a buffer of a few megabytes. This can be done in a pipe-line style arrangement of buffering and processing units. The processing unit can process buffered frames periodically and store only the synopsis video. Compared to other real-time methods that only discard full frames and offline methods that minimize energy functionals on the whole video volume, our method can be considered as standing in the middle: it can summarize partial volumes in real time rates without the need of storing the original video and it can synthesize new frames from the original video for a more compact video synopsis. Note that we do not assign binary labels to the motion pixels as done in [4], which prevents any

data loss due to binarization and eliminates any need for a binarization threshold.

The rest of the paper is organized as follows. In Section II-A we review the non linear image resizing method of [5] and in Section II-B and II-C we describe our method of extending non-linear image resizing to video synopsis problem. In Section II-D we propose the application of our method to a real-time video sequence. We present experiments and results in Section III and the conclusions are given in Section IV.



(a) Horizontal projection (b) Vertical projection

Fig. 3. Calculated projections for the input sequence given in Fig. 2(a). Vertical projection is calculated using Eq. 3 and horizontal projection is calculated in a similar way. It is clear that horizontal projection loses much of the motion information while vertical projection preserves the motion significantly.

II. Method

A. Dynamic Programming and Non-linear Image Resizing

In non-linear image resizing[5], shrinking is done by removing the pixels with the least information along a path. To do so, a new energy image is constructed from the original image such that pixels with high information have higher energy values. For horizontal shrinking, non-linear image resizing finds a vertical minimum energy path on the energy image and removes the pixels belonging to that path. Finding more minimum energy paths and removing corresponding pixels result in the horizontally smaller image of the original image.

The energy image E can be defined as the gradient magnitude of the original image. Pixels with high gradient are likely to be preserved after the shrinking.

$$E = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$$

For horizontal shrinking, a minimum energy path on E should be found. A vertical path on E should start from the first row and must have exactly one element on each row of the image so that removing the pixels corresponding to the path should shrink every row exactly one pixel. On a WxH image, a vertical path is defined as

$$S^v = \{(x(j), j)\}_{j=1}^{H}, s.t. \forall j, |x(j) - x(j-1)| \leq 1, \quad (1)$$

where x is a mapping from row numbers j to a $[1, ..., W]$. A vertical path S^v is composed of points in each row j and the neighboring points of a path can have maximum one displacement in the horizontal direction. Similarly a horizontal path is defined as

$$S^h = \{(i, y(i))\}_{i=1}^{W}, s.t. \forall i, |y(i) - y(i-1)| \leq 1. \quad (2)$$

Finding the vertical or horizontal minimum energy path on E and removing the pixels on the path will shrink the image in the desired dimension. The vertical minimum energy path is found using dynamic programming with the following recursion

$$M(i,j) = E(i,j) + min\{M(i-1,j-1), M(i,j-1), M(i+1,j-1)\},$$

where energy image E is the cost matrix and M is the table to be filled with the cumulative cost values for the paths. When M is filled, the last row of M has the minimum costs for the paths ending on that row. To find the path itself, backtracking is done starting from the minimum cost found on the last row of M. At the end of this process we have the minimum path across the energy image. The pixels belonging to this path are discarded to shrink the image by one pixel.

This method can be generalized from 2D images to 3D space-time video volume. Shrinking the time dimension of the space-time volume would produce non-linear video synopsis. Instead of finding an image path we should find a surface of pixels with least motion information. However, finding such a surface with dynamic programming would take exponential time. To reduce complexity we reduce one dimension of the video volume by projecting it onto a plane orthogonal to its x or y axes.

B. Volume Projections and Surface Discarding

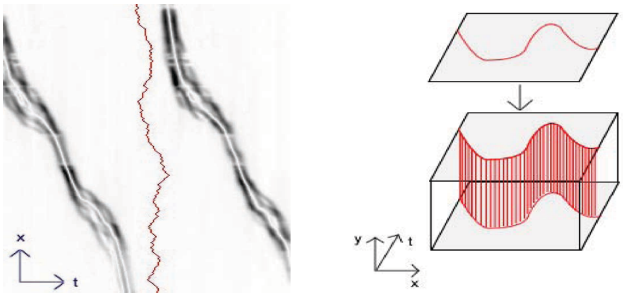
We observe that in most real life activity dynamic objects usually move horizontally because they stand on the ground and the camera is usually placed such that x axis of the camera reference frame is parallel to the ground. With this observation, we choose to project the video volume onto the plane orthogonal to its y axis. Energy image is built as the gradient magnitude in the t direction using the following formula

$$E(i,j) = \sum_{k=1}^{H} I(j,k,i) - I(j,k,i-1). \quad (3)$$

Fig. 3 shows the energy image obtained by Eq. 3 for the sequence shown in Fig. 2(a). Darker areas correspond to higher energy values (higher motion information). Projecting the space-time volume onto a plane orthogonal to its x axis will result losing information about horizontal motion (see Fig. 3).

Running dynamic programming on the energy image E gives us the path of pixels representing the surface to be discarded. An example of such a path can be seen in Fig. 4(a). The corresponding surface to be discarded is shown in 4(b)

Applying this procedure several times will let us remove several surfaces and shortening the video by several frames.



(a) A minimum path on the projection image of video shown in Fig. 2(a)

(b) Back-projection of the minimum path down to the 3D space-time video volume

Fig. 4. Once the minimum path on the projection image is found, it is back-projected down to the video volume to find the space-time surface to be discarded.

When we remove all the energy paths with smaller costs, we obtain the non-linearly summarized video. Doing this operation with high cost paths will result in discarding pixels with high motion information. An example of such a situation is given in Fig. 5(a). Mostly this is due to the connectedness property of the path found by the dynamic programming. An intuitive solution is to let the path break into several individual parts while keeping length of the path fixed, which is explained in the following section.

C. Bands

Eq. 1 and 2 force the paths to be always connected, which is problematic where there are no low energy region to form a path and in this situation some parts of the path are forced to cross the pixels with high energy values (see Fig. 5(a)). In most cases dividing the path into two or three independent parts and solving them individually helps avoid discarding pixels with motion information, which can be achieved by splitting the energy image itself into bands normal to the path direction, see Fig. 5(b) and 5(c).

First, let us consider splitting the energy image into two parts. We choose a splitting point j^* using the following minimization

$$j^* = \underset{1<j<W}{\operatorname{argmax}} \{ D(0,j) * D(j+1,W) \} \qquad (4)$$

where $D(j_1, j_2)$ is the number of passive columns between rows j_1 and j_2. A column i in region r defined by j, is decided to be passive if the average motion along the column is below a certain threshold τ.

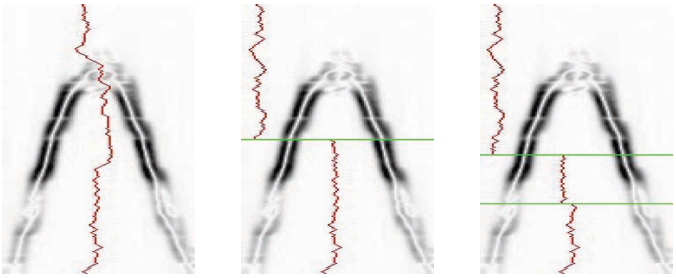
$$passive(i,r) = \begin{cases} 1 & \text{if } \sum_{j' \in r} E(i,j') < \tau \\ 0 & \text{otherwise} \end{cases} \qquad (5)$$

Individual minimum cost paths are constructed for each region split by j^*. Fig. 5(b) shows the regions defined by j^* minimizing Eq. 4. Paths drawn in each region on the figure are minimum cost paths for the region they belong, a single minimum cost path for the whole image would have to cross the motion pixels drawn in darker.

Splitting the energy image into three regions is similar. In this case we find two j values $(j_1, j_2)^*$ via the following minimization.

$$(j_1,j_2)^* = \underset{1<j_1<j_2<W}{\operatorname{argmax}} \{ D(0,j_1) * D(j_1+1,j_2) * D(j_2+1,W) \} \qquad (6)$$

Fig. 5(c) shows the separated bands defined by $(j_1, j_2)^*$ minimizing Eq. 6. Fig. 5(c) also shows the individual cost paths through these bands. If we did not used multiple bands, the minimum cost path would have passed through the motion pixels as shown in Fig. 5(a).



(a) Cost: 6640 (b) Cost: 914 (c) Cost: 1080

Fig. 5. Projected energy images of the video shown in Fig. 8. (a) When there is no low energy region to form a connected path, the path is forced to cross the motion pixels. Splitting the projection image into bands helps avoid discarding motion pixels. (b) Projection image is split into two bands. (c) Projection image is split into three bands. Total cost of the divided paths are much lower than the fully connected path.

D. Real-Time Video Synopsis

Classical video summarizing techniques that work in real-time can only discard full frames. These techniques fail to summarize any video sequence with constant continuous motion in a small part of the frame. Although more advanced techniques can handle this type of videos by minimizing energy functionals over the whole space-time volume, they cannot be used in real-time applications due to their algorithmic complexities. The method introduced in this paper can be effectively used for real-time synopsis of such videos. Our method still needs to work on a volume rather than individual frames, thus working on small portions of the video sequence is necessary. A simple solution is to run our method periodically on a buffer accumulated in time. Once the buffer is full, the projection of the volume can be directly used to select the pixels to discard from different locations of the video portion and only the resulting synopsis video needs to be stored explicitly. Running our method either when the buffer is full or when a scene change is detected [11], [12] can be seen as a more sophisticated idea.

In real-time video synopsis, we actually need only the projection of the video portion, which can be build cumulatively, and the actual frames can be stored in a compressed format. In addition, we can discard frames with no motion without considering even buffering such frames. Our current implementation allows successive 15 minute compression

periods with a about 10 megabytes of buffer space on an average hardware for a typical surveillance video of 320x240 resolution at 15 frames per second.

III. Experiments

We have tested our method with several experimental videos. In the resulting synopsis video, dynamic objects from different time intervals can be seen at the same time, performing the same motion as they did before. In Fig. 6, frames from the resulting synopsis video of the original video (Fig 2(a)) can be seen. The original video is a total of 241 frames and the synopsis video is computed by shortening 113 frames in length. The same video can be summarized into only about 230 frames by classical frame discarding methods.

Fig. 7 shows an example where we separated the energy image into two bands. Here, the original video was 96 frames in length and we shortened the video by 48 frames to generate the synopsis video. If we did not use the multiple bands there would have been some serious artifacts on the appearance of the dynamic object because the minimum path would cross the motion pixels, as seen in Fig. 5(a).

Fig. 8 shows a source video of a person waking across the scene back and forth. Summarizing this input video without using multiple bands would result in a heavily cluttered synopsis video. In Fig. 9 and 10 we give the results for this input video using two and three bands.

The experiments and results can be reached at `http://vision.gyte.edu.tr/projects.php?id=5`.

IV. Conclusion and Discussion

We presented a very efficient method for obtaining the non-linear video synopsis. Summarizing a video sequence lets the viewer observe much more compact information in a short amount of time. Although it is possible to quickly summarize a video using linear whole frame elimination methods, these methods cannot handle videos with very small moving image regions. More advanced non-linear methods, on the other hand, can summarize a video more efficiently by considering the volumetric motion densities in the space-time volume of the video. By synthesizing new frames from the original video frames, much more compact video summaries can be obtained. These non-linear methods, however, are very slow due to complex optimizations involved.

The method presented in this paper can be considered as standing between the fast classical linear and advanced non-linear video summarizing techniques. The method is very fast like the linear methods and it can easily run at real time rates on average hardware. The method, at the same time, can synthesize new frames by considering volumetric densities for a much compact video summary like the non-linear methods. In other words, our method carries best of both classical and advanced video summarizing methods.

The key for the high performance of the presented method is to employ fast dynamic programming methods on the projections of space-time video volumes. A method of dividing the projection image into bands is presented as a solution to difficult videos where a single minimum dynamic programming cost path on the projection image compulsorily crosses relatively important pixels. Dividing the projection image into bands lets the minimum cost path to be composed of individual paths in each band. As a result, energy of a composed path is likely to be lower than a single connected path for the same projection image.

Fast video synopsis can easily be applied to very long videos. An example of a typical long video is a movie, which are usually consist of scenes with dominant scene changes which will be reflected in the video projections we use. By detecting the scene changes, our method can be used on different space-time projections for each scene. Even without scene change detection, fast video synopsis will not discard pixels from scene borders meaning that a scene in the synopsis video will not contain pixels from other scenes. However, detection of scene changes, which can easily be detected on the volume projections, may speed up the process.

As the future work, we plan to work on improvements like automatic selection of number of bands necessary for each scene, preserving object integrities, and modifying the formulation so that the discarded surface is a minimum or nearly minimum surface in the video volume. The proposed method is very suitable for real-time scenarios like surveillance videos and as a future work, we plan to implement our algorithm on a stand alone compact vision sensor that can produce summarized videos in real time.

References

[1] C. Kim and J. Hwang. An integrated scheme for object-based video abstraction. In ACM Multimedia, pages 303311, New York, 2000.

[2] A. M. Smith and T. Kanade. Video skimming and characterization through the combination of image and language understanding. In CAIVD, pages 6170, 1998.

[3] X. Zhu, X. Wu, J. Fan, A. K. Elmagarmid, and W. G. Aref. Exploring video content structure for hierarchical summarization. Multimedia Syst., 10(2):98115, 2004.

[4] A. Rav-Acha, Y. Pritch, and S. Peleg. Making a long video short: Dynamic video synopsis. In CVPR06, pages 435441, New-York, June 2006.

[5] S. Avidan and A. Shamir. Seam Carving for content-aware image resizing. SIGGRAPH, 2007.

[6] B. T. Truong and S. Venkatesh. Video Abstraction: A Systematic Review and Classification. ACM Transactions on Multimedia Computing, Communications, and Applications, 3 (1), 1-37, 2006.

[7] J. Nam and A. T. Tewfik, Dynamic video summarization and visualization. In Proc. 7th ACM Int. Conf. Multimedia, 1999, pp. 5356.

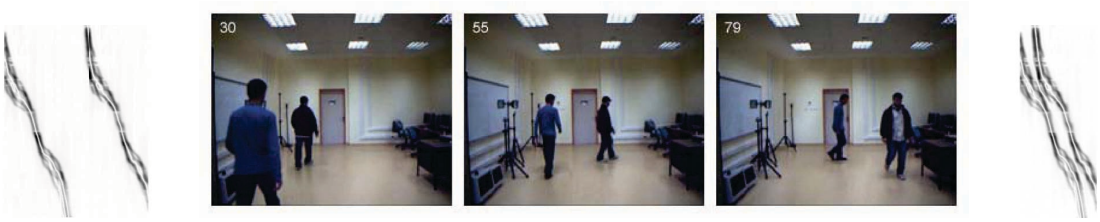
[8] A. Komlodi and G. Marchionini. Key frame preview techniques for video browsing. In Proc. 3rd ACM Converence on Digital Libraries, 1998, Pages 118-125.

[9] F. C. Li, A. Gupta, E. Sanocki, L. He and Y. Rui. Browsing Digital Video. In ACM CHI, 2000, Pages 169-176.

[10] S. Uchihashi, J. Foote, A. Girgensohn and J. Boreczky. Video manga: generating semantically meaningful video summaries. ACM Multimedia99, 1999.

[11] C.-W. Ngo, Y.-F. Ma and H.-J. Zhang. Video summarization and scene detection by graph modeling. IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 2, pp. 296-305, 2005.

[12] C. Cotsaces, N. Nikolaidis and I. Pitas. Video shot detection and condensed representation - a review. IEEE Signal Processing Magazine, vol. 23, no. 2, pp. 28-37, 2006.



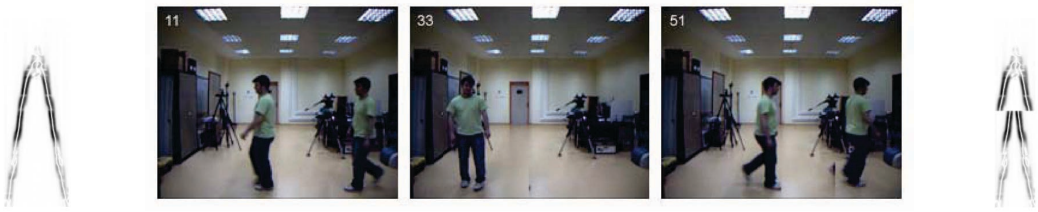
(a) (b) (c)

Fig. 6. Video synopsis of the input video shown in Fig. 2(a). The resulting synopsis video is generated by shortening the input video by 113 frames. (a) Projection image of the input video. (b) Sample frames from the synopsis video. (c) Projection image of the synopsis video.

(a) (b) (c)

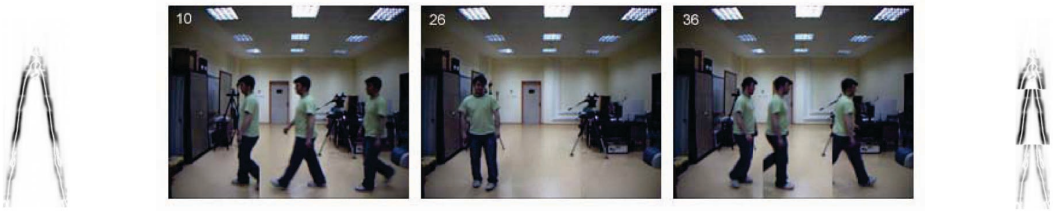
Fig. 7. Video synopsis generated for the first half of the input video shown in Fig. 2(a). The input video consists of 96 frames and the resulting synopsis video is generated shortening by 48 frames. (a) Projection image of the input video. (b) Sample frames from the synopsis video. (c) Projection image of the synopsis video.

Fig. 8. Sample frames from an experimental video sequence where a person walks across the scene, turns back and walks back to the starting position.



(a) (b) (c)

Fig. 9. Video synopsis generated for the input sequence in Fig. 8. The synopsis video is generated using two bands and shortening the input video of 107 frames by 40 frames. (a) Projection image of the input video. (b) Sample frames from the synopsis video. (c) Projection image of the synopsis video.



(a) (b) (c)

Fig. 10. Video synopsis generated for the input sequence in Fig. 8. The synopsis video is generated using three bands and shortening the input video of 107 frames by 50 frames. Using three bands allows summarizing the input video more aggressively. (a) Projection image of the input video. (b) Sample frames from the synopsis video. (c) Projection image of the synopsis video.