VIDEO SUMMARIZATION BY SPATIAL-TEMPORAL GRAPH OPTIMIZATION

Shi Lu, Michael R. Lyu, Irwin King

Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong SAR
{slu, lyu, king}@cse.cuhk.edu.hk

ABSTRACT

In this paper we present a novel approach for video summarization based on graph optimization. Our approach emphasizes both a comprehensive visual-temporal content coverage and visual coherence of the video summary. The approach has three stages. First, the source video is segmented into video shots, and a candidate shot set is selected from the video shots according to some video features. Second, a dissimilarity function is defined between the video shots to describe their spatial-temporal relation, and the candidate video shot set is modelled into a directional graph. Third, we outline a dynamic programming algorithm and use it to search the longest path in the graph as the final video skimming. A static video summary is generated at the same time. Experimental results show encouraging promises of our approach for video summarization.

1. INTRODUCTION

Video is pervasive nowadays. However, it is difficult to quickly find out a video file that we want from the vast video repositories. Moreover, in the cases when the bandwidth is limited, e.g, the wireless hand-hold devices using MMS (Multimedia message services), a digest version of the video is needed to meet the limited bandwidth. To solve these problems, *video summarization* has received more and more attention.

Video summarization is a short summary of a longer video document. There are two different kinds of video summarizations: *static video summary*, which is composed of a collection of salient images extracted or synthesized from the original video, and *dynamic video skimming*, which is a shorter version of the original video made up of a set of continuous video clips.

From the user's point of view, a video summary with good quality should maintain the following three attributes: *conciseness, comprehensive coverage, and visual coherence*. Conciseness requires that the dynamic video skimming should not exceed the given length limit, and the static summary should not have too many images. An informative video summary, furthermore, should comprehensively cover the visual diversity and temporal distribution of the original video. Finally, for dynamic video skimming, too frequent scene change will cause jumpy feeling to the user. A coherent video skimming is consequently more preferable.

In recent years much work has been conducted on video summarization. For static summary, most early work selects key frame images by random or uniform sampling, like the MiniVideo systems [1]. Later work tends to adapt to the dynamic video content like [2]. A mosaic-based approach is suggested in [3]. In [4], the authors analyzed the video structure after video segmentation, and then get a tree-structured Video-Table-Of-Contents(V-TOC).

For video skimming generation, in the VAbstract system [5], key movie segments are selected to form a movie trailer. The Informedia system [6] selects the video segments according to the occurrence of important keywords in the corresponding caption text. Later work employ perceptional important features to summarize video. In [7], a user attention curve is constructed to simulate the user's attention toward different video contents. In [8] an utility function is defined for each video shot, and video skimmings are generated by the utility maximization. [9] assigns different weight scores on several important features then selects the video skimming that maximizes the feature score summation.

Although many of video summarization techniques have been proposed, few of them has clearly emphasize simultaneously achieving the three video summarization criteria for good quality. The current approaches are either domain-specific or focusing on only features, while neglecting the content coverage of the source video. In this paper we describe a novel three-stage graph-optimization based video summarization approach, engaging video segmentation, video feature detection and the spatial-temporal properties of the video, aiming at concurrently meeting the criteria listed above. Both the dynamic and static video summaries are generated accordingly.

The paper is organized as follows: In Section 2 we describe our three-stage video summarization procedure. In

Section 3 we show some experimental results. In Section 4 we make conclusion and discuss our future work.

2. VIDEO SUMMARIZATION PROCEDURE

2.1. Video shot detection and candidate shot selection

Video shot is the composing unit of edited videos. It is an image sequence recorded continuously by a single camera. Our shot detection method is similar to the method in [10], while we improve the filtering step for more accuracy. We use the first frame $kf_{i_{begin}}$ and the last frame $kf_{i_{end}}$ of the shot sh_i as the key frames to represent the visual content of the video shot. Since video shot itself is already a continuous coherent image sequence, based on video shots, we can ensure the coherence of the summary.

After the video shots are detected, we can select the candidate video shot set from the segmented shots according to some interesting features detected from the video. For example, we select piercing noise (to denote gunshot and explosion), human face occurrence, loud voice, and the color of fire as our interesting features for movie clips; for sitcom clips we select human face detection and background laughter as interesting features.

2.2. Graph modelling and optimization based skimming generation

After the candidate shot are selected, we can define a dissimilarity function between candidate video shots to measure their spatial-temporal relation. First, we convert the key frame images into HSV color space, then measure the visual similarity between two shots by the maximal H-S histogram correlation between their key frames, that is,

$$VisualSim(sh_i, sh_j) = \max_{x,y} HistCorr(kf_{i_x}, kf_{j_y}),$$

where $x, y \in \{begin, end\}$.

The temporal distance $TempDis(sh_i, sh_j)$ between two video shots sh_i and sh_j is defined as the temporal distance between their center points, in terms of their frame numbers.

Then we define our spatial-temporal dissimilarity function between two video shot sh_i and sh_j as:

$$Dis(sh_i, sh_j) = 1 - VisualSim(sh_i, sh_j) \times e^{-\frac{TempDis(sh_i, sh_j)}{k}},$$

where k is the parameter to control the slope of the exponential function.

From the definition we can see that both the visual (spatial) similarity and temporal distribution are included in the dissimilarity function. To allow for a good coverage of both the visual and temporal coverage of the video contents, we define the dissimilarity function such that it changes linearly with the visual similarity, but exponentially with the temporal distance. Thus we can search for the optimal video skimming that captures both the visual diversity and temporal coverage of the original video by optimizing this function.

After defining the shot-pairwise spatial-temporal dissimilarity function, we can model the selected candidate shots by a directional acyclic complete graph $G(V, E)$. A vertex v_i in the vertex set V corresponds to a video shot sh_i. Consequently, on each vertex there is a weight which is equal to the length of the corresponding video shot. An edge e_{ij} in the edge set E connects the two vertexes v_i and v_j, and the weight on e_{ij} is the spatial-temporal dissimilarity function between the video shots sh_i and sh_j. The direction of e_{ij} is from the earlier shot to the later shot. A simple example of the spatial-temporal relation graph on five candidate video shots is shown in Fig. 1.

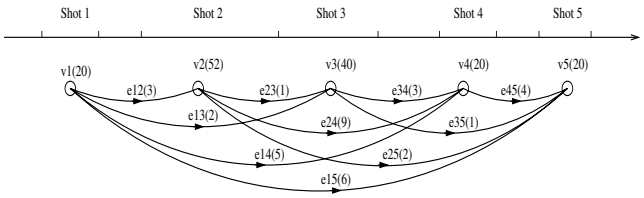

Fig. 1. Spatial temporal distance graph for candidate shots

A path $p_i = \{v_{i_1}, ...v_{i_n}\}$ in the spatial-temporal relation graph represents a set of video shots $\{sh_{i_1}, ...sh_{i_n}\}$, which is a video skimming whose total length is the summation of the weights on the vertexes $v_{i_1}, ...v_{i_n}$ in the path. The length of the path is the summation of the spatial-temporal dissimilarity function between consecutive video shot pairs. Since we want the video skimming to cover both the visual contents and the temporal distribution of the original video, given the summary length L_{vs}, we can search for a longest path in the spatial-temporal relation graph and use the video shots corresponding to its vertexes as the video skimming, under the constraint that the summation of the vertex weights is L_{vs}. However, given L_{vs}, maybe no path with the vertex summation exactly equal to L_{vs} exists. To solve this problem, we employ a tolerance threshold TH then search for the longest path with the vertex weight summation within the interval $[L_{vs} - TH, L_{vs}]$ and use the shots corresponding to the vertexes in the path as the optimal skimming.

2.3. Solution and algorithm

The problem of finding the longest path in the graph with its vertex weight summation to lie within an interval is a constrained optimization problem. Brute force searching is feasible but inefficient; however, the problem has an optimal

substructure and can be solved with dynamic programming as follows.

Suppose that the number of video shots is S_n. A path $p_{i_x} = \{v_{i_x}, v_{i_x+1}, ...v_{i_n}\}$ is a path in the spatial-temporal relation graph beginning with vertex v_{i_x}. Let L_r be the left vertex weight summation that the path should cover at most, $L_{opt}(p_{i_x}, L_r)$ be the optimal length of such a path, and l_i be the length of the video shot corresponding to vertex v_i. Then we have the following optimal substructure:
$L_{opt}(p_{i_x}, L_r) = \max_{t=i_x+1}^{S_n}(Dis(v_{i_x}, v_t) + L_{opt}(p_t,$
$L_r - l_t))$.

Based on the optimal substructure we can derive the following dynamic programming algorithm shown as Algorithm 1 to find the longest path in the spatial-temporal relation graph with the vertex weight summation limit constraint.

Algorithm 1 Dynamic programming algorithm

Input: The candidate shot set $Sh_{in} = \{sh_1....sh_{S_n}\}$, and the shot pairwise dissimilarity function $Dis(sh_i, sh_j)$;
Output: The maximum of the dissimilarity summation function value $LongestLength$ and all optimal sub-solutions $L_{opt}[currentshot][L_r]$.
BEGIN
Set $L_{opt}[i][j] = 0$ for all i,j;
for $L_r = TH$ to L_{vs} **do**
  $L_{opt}[LastShot][L_r] = -penalty$;
end for
for $i_x = S_n$ to 0 **do**
  for $L_r = 0$ to L_{vs} **do**
    $opt = -infinity$;
    for $t = i_x + 1$ to S_n **do**
      if $l_t < L_r$ **then**
        if $opt < L_{opt}[t][L_r - l_t] + Dis(sh_t, sh_{i_x})$ **then**
          $opt = L_{opt}[t][L_r - l_t] + Dis(sh_t, sh_{i_x})$;
        end if
      end if
    end for
    $L_{opt}[i_x][L_r] = opt$;
  end for
end for
$LongestLength = L_{opt}[0][L_{vs}]$;
END

To ensure that the vertex summation of the path is within the specified interval, we add a negative penalty value to $L_{opt}(LastShot, L_r)$ for $L_r > TH$, so that the length of the found path will lie in $[L_{vs} - TH, L_{vs}]$.

Algorithm 1 calculates the length of the optimal path and all the optimal sub-solutions. With the optimal sub-solutions we can easily trace back and find the global optimal path. The trace back algorithm is omitted here. If the algorithm failed to find any solutions in the current interval $[L_{vs} - TH, L_{vs}]$, we will increase the tolerance threshold value, and continue searching until we finally get a solution.

The time complexity of the algorithm to find the optimal path is $O(n^2 \times L_{vs})$, and its spatial complexity is $O(n \times L_{vs})$.

3. EXPERIMENTS AND DISCUSSIONS

To test the performance of our video summarization method, we implemented the dynamic programming algorithm and applied them to several video clips. We employed a PC platform with 2.0Ghz P4 CPU on the Win2000 OS. In our experiments, we choose all the video shots with one or more features in its duration as candidate video shots.

The example movie clip is 477 seconds long, with 11,409 video frames. After candidate shot selection, the candidate video shots contain 4,775 frames. We select the following video features for candidate shot selection: human face occurrence, loud voice, loud noise like gunshot and explosion, and the color of fire. Any video shot that have one or more features listed above is selected as a candidate shot. The feature distribution and the candidate shots are shown in Fig. 2. The exponent control parameter k in the spatial-temporal dissimilarity function is set to 400, and the tolerance threshold is set to 20 frames. The distribution of the candidate

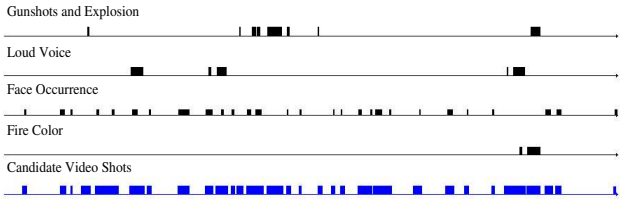

Fig. 2. Candidate shots selected according to 4 features

video shot set and the video skimming are shown in Fig. 3, and the static video summary is shown in Fig. 4.

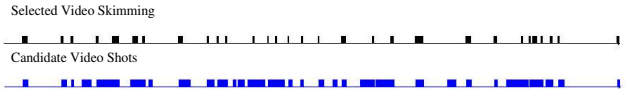

Fig. 3. Temporal distribution of the selected video shots

From Fig. 3 we can see that the selected video shots have covered the temporal axis quite well. From Fig. 4 we further observe that, the visual contents covered by the video skimming are quite diverse. Consequently, the generated video skimming does cover both the visual and the temporal contents of the original video.

Ten people were invited to watch the video skimming generated with various compression rates then answer some questions about the video contents. For example, suppose

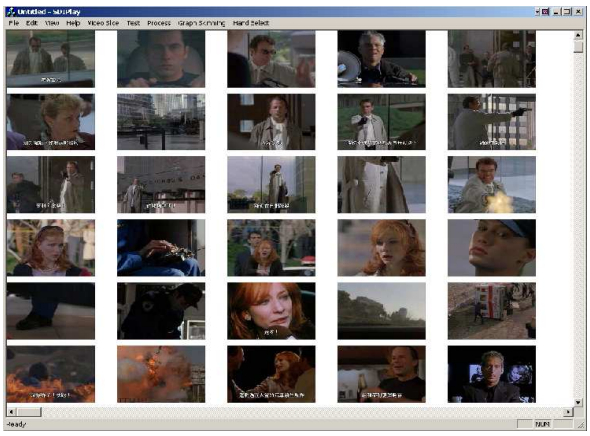

Fig. 4. Static video summary

there are N key actors in the video, we ask the test users to tell how many key actors they can perceive by watching the video skimming. Thus the score for the question "Who?" is defined as the actor number that the users can find divided by N. Question "What?" deals with the key events in the video. Question "Coherent?" asks the users to give their evaluation scores according to their feelings to the coherence of the video skimming. All scores are scaled to 10.

Table 1 shows the user test results. From the table we conclude that the video skimmings still make good sense to people with a compression rate at 0.15. A higher compression rate at 0.30 yields better result. Also, from the table we can see that the users agrees that the coherence of the video skimmings is acceptable.

Clip	Length	Rate	Who?	What?	Coherent?
Movie1	477 sec.	0.15	7.78	8.05	6.94
		0.30	9.07	9.50	8.15
Movie2	1230 sec.	0.15	8.33	7.50	7.35
		0.30	9.63	9.82	8.56
Sitcom1	1200 sec.	0.15	7.67	8.23	7.40
		0.30	8.46	8.68	8.62
Cartoon1	930 sec.	0.15	7.14	8.09	7.03
		0.30	9.52	9.38	8.21

Table 1. User test results

4. CONCLUSION AND FUTURE WORK

Video summarization is a very valuable tool for video browsing and management. In this paper, we formulate the techniques to locate the candidate video shot set according to several video feature distributions, define the spatial-temporal dissimilarity function between video shots, model the candidate shot set into a spatial-temporal relation graph, and use dynamic programming to generate both the dynamic video skimming and the static video summary by searching a constrained longest path in the spatial-temporal relation graph. The obtained experimental results are encouraging.

In the future, we will employ higher video structures, like video style analysis and video editing syntax, into our framework. Appropriate intra-shot compression will be studied to shorten the selected video shots' lengths in order to further magnify the content coverage.

5. ACKNOWLEDGEMENT

The work described in this paper was fully supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK 4351/02E and Project No. CUHK4182/03E).

6. REFERENCES

[1] Y. Taniguchi, A. Akutsu, Y. Tonomura, and H. Hamada, "An intuitive and efficient access interface to real-time incoming video based on automatic indexing," in *Proceedings of the third ACM international conference on Multimedia*, 1995, pp. 25–33.

[2] H. J. Zhang, D. Zhong, and S. W. Smoliar, "An integrated system for content-based video retrieval and browsing," *Pattern Recognition*, vol. 30, no. 4, pp. 643–658, 1997.

[3] M. Lee, W. Chen, C. Lin, C. Gu, and T. Markoc, "A layered video object coding system using sprite and affine motion model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 1, pp. 130–145, 1997.

[4] Y. Rui, T.S. Huang, and S. Mehrotra, "Constructing table-of-content for videos," *ACM Multimedia Systems Journal, Special Issue Multimedia Systems on Video Libraries*, vol. 7, no. 5, pp. 359–368, Sept 1999.

[5] R. Leinhart, S. Pfeiffer, and W. Effelsberg, "Video abstracting," *Communication of the ACM*, pp. 55–62, December 1997.

[6] M. A. Smith and T. Kanade, "Video skimming and characterization through the combination of image and language understanding techniques," in *Proceedings of the IEEE Computer Vision and Pattern Recognition*, 1997, pp. 775–781.

[7] Y. F. Ma, L. Lu, H. J. Zhang, and M. J. Li, "A user attention model for video summarization," in *Proceedings of ACM Multimedia*, 2002, pp. 533–542.

[8] H. Sundaram, L. Xie, and S. F. Chang, "A utility framework for the automatic generation of audio-visual skims," in *Proceedings of the ACM Multimedia*, 2002, pp. 189–198.

[9] S. Lu, I. King, and M. R. Lyu, "Video summarization using greedy method in a constraint satisfaction framework," in *Proceedings of 9th International Conference on Distributed Multimedia Systems*, 2003, pp. 456–461.

[10] A. M. Ferman and A. M. Tekalp, "Efficient filtering and clustering methods for temporal video segmentation and visual summarization," *Jounal of Visual Communication and Image Representation*, vol. 9, no. 4, pp. 336–51L, 1998.