VIDEO SUMMARIZATION USING FEATURE POINT TRACKING

Yousri Abdeljaoued and Touradj Ebrahimi

Ecole Polytechnique Fédérale de Lausanne (EPFL), Signal Processing Institute (ITS)
CH-1015 Lausanne, Switzerland
e-mail: Yousri.Abdeljaoued@epfl.ch and Touradj.Ebrahimi@epfl.ch

ABSTRACT

Based on the tracking of feature points, a key frame based representation of the video is generated. This representation is well suited for video summarization. An activity measure is defined, which depends on the number of tracks that are initiated or terminated. A segmentation of this activity measure into stationary segments is used to detect the shot boundaries as well as to extract the key frames. The evaluation of the algorithm on typical sequences shows that the extracted key frames capture the salient content of the video.

1. INTRODUCTION

With the increasing amount of audio-visual data that are broadcast or available on prerecorded format, there is an emerging need for efficient media management including browsing, filtering, indexing, and retrieval. This paper deals with video summarization which consists mainly in segmenting the video into elementary units and extracting representative frames from these units. Many visual features such as color and motion have been used for video summarization. Color histogram-based techniques in particular have been shown to be robust and effective [1]. However, such techniques have drawbacks in scenes with camera and object motion. We propose to use a feature point tracking system to compute an activity measure based on the number of tracks which are initiated or terminated to segment the video as well as to extract representative frames from the video.

This paper is organized as follows. Section 2 gives an overview on existing techniques for video summarization. Section 3 introduces our approach to summarize a video. Experimental results obtained with the proposed algorithm and a state-of-the-art algorithm are discussed in Section 4.

2. STATE-OF-THE-ART

It is useful to introduce the fundamental elements of a simple system for video summarization. This allows us to show the relationships between the different elements. Figure 1 shows a block-diagram of such a system. First, the original video is processed in order to extract low-level visual primitives, such as color, motion, and texture. Based on the low-level visual primitives, the video is first segmented into basic units called shots. Temporal segmentation corresponds to the detection of the boundaries between these shots. Once the shot boundaries are detected, the salient content of each shot is represented in terms of a small number of frames, called *key frames*. Temporal segmentation and key frame extraction make up the *video parsing* process. Finally, different summary representations are created from the detected shots and the extracted key frames. The main objective of these representations is to allow a content-based browsing of the video.

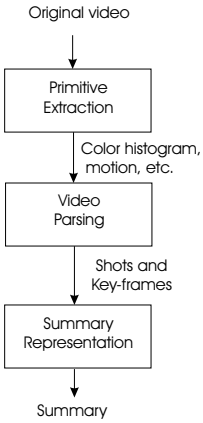

Fig. 1. Block-diagram of a simple system for video summarization.

Most techniques in the literature propose a solution only for one of these fundamental elements. In the following, we give an overview on algorithms for video parsing and video representation. We do not include methods for visual primitive extraction, because it is a different research area.

2.1. Video parsing

2.1.1. Temporal segmentation

Based on the visual primitives used by different temporal segmentation algorithms, it is possible to classify them into color-based, edge-based, and motion-based algorithms. The main issues addressed by these algorithms are the detection of gradual transitions (i.e. fade-in, fade-out, and dissolve), which is more difficult than the detection of cuts, and the identification of camera operations and object motion to reduce false positives.

Zhang *et al.* [2] used histograms as descriptors for color. If the computed histogram distance between two consecutive frames is higher than a threshold T_b, a cut is declared. Another threshold T_s, which is lower than T_b, is used to detect gradual transitions. If the consecutive histogram distance is higher than T_s but lower than T_b, then the corresponding frame is marked as a potential start of a gradual change, and the running histogram distance between this frame and the subsequent frames is computed. The end frame

of the gradual transition is detected when this running histogram distance is higher than T_b.

Zabih *et al.* [3] proposed to use edges as visual primitives for temporal video segmentation. First, edges are extracted using the Canny detector [4] from two consecutive frames. By computing a dissimilarity measure based on the fraction of edge-pixels which enter and exit between two consecutive frames, it is possible to detect cuts and gradual transitions as local maxima.

Bouthemy *et al.* [5] proposed to use the Iteratively Reweighted Least Squares (IRLS) technique to estimate efficiently the dominant motion. This robust estimation technique allows to detect the points which belong to the part of the image undergoing the dominant motion (inliers). If a cut occurs between two consecutive frames, the number of inliers n_d is close to zero.

2.1.2. Key frame extraction

Many criteria are used to extract key frames. These criteria can be divided into the following categories:

- **Shot-based criteria —** Nagasaka and Wang [6] proposed to select the first frame in a shot as a key frame. Although simple, this technique is not able to extract the salient content of a shot, especially for shots with motion and high activity.
- **Color-based criteria —** The current frame of a shot is compared to the last selected key frame by using the color histogram-based distance. If this distance is higher than a threshold T_k, the current frame is selected as a new key frame. This process is iterated until the end of the shot [2].
- **Motion-based criteria —** Wolf [7] proposed a motion-based approach. First the optical flow for each frame is determined, and a motion metric based on optical flow is computed. Then, by analyzing the motion metric as a function of time, key frames are selected at the minima of motion. This is explained by the fact that the camera stops on a new position, or the characters maintain gestures to emphasize their importance.

2.2. Summary representation

The following summary representations are possible:

- **Hierarchical summary —** This representation allows random access. The key frames are grouped and organized in order to obtain a coarse-to-fine hierarchy of summaries. Zhong *et al.* [8] used a fuzzy K-means clustering algorithm to group similar key frames in terms of a low-level visual primitive (e.g. color) into classes at each level of the hierarchy. Due to the sequential nature of video, the visualization of such hierarchical summaries is difficult, especially when the number of key frames is large.
- **Sequential summary —** This representation is simply a concatenation of the key frames which can be shown sequentially in time, for example as an animated slide show. The temporal order the of the key frames in the original video is preserved. This allows to understand the relationships between the different events in the video.
- **Mosaic-based summary —** Each shot is decomposed into static and dynamic components [9]. The static appearance is represented by a mosaic, which is constructed by aligning and integrating frames. The dynamic behaviour of the moving objects is represented by their trajectories and characteristic appearances. One of the drawbacks of this representation is that its use is limited to shots containing camera operations, such as panning and zooming. In fact, when the camera is still, key frames are better suited for summarization than a mosaic.

3. PROPOSED ALGORITHM

Given a video sequence, each frame is first processed by the feature point extraction algorithm presented in [10]. Thanks to a scale-space representation, this algorithm yields stable and well-localized feature points estimates. The extracted feature points are used as input for a feature point tracking system. The Interacting Multiple Model filter combined with the assignment algorithm proposed by Jonker and Volgenant (IMM-JV) [11] is used for the tracking of feature points. It also includes a track management step to initiate, update, and terminate tracks. In contrast to the Kalman filter together with the nearest neighbor filter used in [12], the IMM-JV tracking algorithm is suited for many video sequences with different levels of activity. Then, an activity measure is computed, which depends on the number of tracks that are terminated or initiated. A temporal segmentation algorithm is used to segment the activity signal into stationary segments. Such segments are equivalent to actions, where the key frames are extracted. This algorithm is also able to detect abrupt changes (cuts) as well as slow changes (gradual transitions). These changes correspond to shot boundaries.

The sequential representation is selected to summarize a video because it is simple and maintains the temporal order of the key frames in the original video.

3.1. Activity measure

Based on the survey of temporal video segmentation algorithms, the measure used for the comparison of two frames should be large at shot boundaries and small in between. We thus propose an activity measure that is defined as follows [12]:

$$a(k) = \max\Big(\frac{\text{Number of terminated tracks at k}}{\text{Total number of tracks at k-1}}, \frac{\text{Number of initiated tracks at k}}{\text{Total number of tracks at k}}\Big), \quad (1)$$

where k denotes the frame number. Note the key role of the track management step in the computation of this activity measure.

When the camera is still and there are no objects leaving or entering the scene, the activity measure is close to zero. On the other hand, transitions between camera operations or objects which enter or exit the scene are likely to give rise to a relatively small change in the activity measure. This can be explained by the small number of tracks which may be initiated or terminated.

When a shot boundary occurs, many tracks are terminated or initiated. This causes a large change in the activity measure over a short time interval. Therefore, this activity measure reflects well changes due to shot boundaries. More specifically, cuts correspond to an abrupt change during one frame, whereas gradual transitions are equivalent to slow changes over a limited number of frames (gradual transitions generally have a short duration). In addition to shot boundary detection, the proposed activity measure is also used for the extraction of key frames by exploiting the local changes within a shot.

3.2. Temporal segmentation algorithm

The temporal segmentation algorithm should decompose the activity measure into stationary segments. For example, these stationary segments could correspond to camera operations, or to a change in the object composition of the scene. The temporal segmentation algorithm should also be able to detect nonstationarities, which are equivalent to abrupt changes over one frame or slow changes over a small number of frames. The idea behind these requirements on the temporal segmentation algorithm is the way we model the activity measure of a video as a sequence of stationary segments, from which the key frames are extracted, and nonstationarities, which are the shot boundaries. The combination of the requirements on the activity measure and the temporal segmentation algorithm results in low numbers of false positives and false negatives, and thus leads to an efficient content-based key frame extraction scheme.

The *exponential weighted moving average* (EWMA) algorithm [13] has been selected for the temporal segmentation task because it fulfills all of the above requirements. Furthermore, the EWMA algorithm uses only the previous and the current activity measure as inputs. Therefore this algorithm is easy to implement on-line (and on-line processing is a desired property because of the sequential nature of the video). The use of the EWMA algorithm for temporal segmentation is the main novelty as compared to [12].

The EWMA algorithm starts by filtering the activity measure $a(k)$ as follows:

$$w(k) = \lambda a(k) + (1 - \lambda)w(k - 1), \tag{2}$$

where $w(k)$ is the filtered version of $a(k)$. This expression can be interpreted as a weighted average, in which $a(k)$ is given a weight λ $(0 < \lambda \leq 1)$, and $w(k - 1)$ is given a weight $1 - \lambda$. In this way the weight of historically "old" activity measures decreases geometrically as a function of time. This weighting allows us to detect abrupt changes as well as slow changes.

A change is detected at k if:

$$|w(k) - \mu| \geq 3\sigma_w, \tag{3}$$

where μ is the mean of $a(k)$, and σ_w is the standard deviation of $w(k)$, which is computed as follows:

$$\sigma_w = \sqrt{\sigma\Big(\frac{\lambda}{2 - \lambda}\Big)[1 - (1 - \lambda)^{2k}]}, \tag{4}$$

where σ is the standard deviation of $a(k)$.

When a change is detected at k, then $w(k)$ and μ are initialized with the current $a(k)$, and σ with a large value. We note that the mean μ and the standard deviation σ of $a(k)$ are not constant during the segmentation process. The estimates of both parameters, which characterize the local properties of the segment, are improved as more data of the corresponding segment is collected.

3.3. Key frame extraction

Based on the temporal segmentation of the activity measure, we could use stationarity as the criterion for the selection of key frames. Stationary segments correspond to the important actions within a shot. By selecting one frame in the middle of each stationary segment as a key frame, the salient content of a shot is captured. We believe that such a choice allows us to represent the segment in a compact way. For instance, if we consider a stationary segment which consists of a camera zoom, the frame in the middle of this state is a good compromise between the global and the focused view. If more details about the video sequence are required, we propose to select one frame at the beginning, one in the middle, and one at the end of each stationary segment.

Algorithm	Total	Detected	False positive
IMM-JV	23	23	5
Histogram	23	16	8

Table 1. Cut detection results for IMM-JV and color-histogram based algorithm with the three sequences.

Algorithm	Total	Detected	False positive
IMM-JV	7	4	2
Histogram	7	1	7

Table 2. Gradual transition detection results for IMM-JV and color-histogram based algorithm with the three sequences.

4. EXPERIMENTAL RESULTS

We have evaluated the different fundamental elements of our video summarization system. These elements are temporal video segmentation and key frame extraction. To provide a comparison to a state-of-the-art algorithm, we implemented a color histogram-based algorithm similar to the one described in [14] [2]. The reason behind the selection of the color histogram-based algorithm is its good performance [1]. In addition to that, this algorithm provides a solution for temporal video segmentation as well as key frame extraction.

We have selected three video sequences, each with a length of 3000 frames, for the evaluation of the different algorithms. The *news* sequence contains many cuts as well as gradual transitions. It includes some camera operations. Such sequences usually have a distinct temporal structure, namely the regularly spaced appearances of the anchor-person as an indication for the start of a new subject. The *nhk* sequence is a documentary which contains many captions. It also includes some camera operations. The *basketball* sequence involves many moving objects (players) as well as camera operations. We believe that the selected sequences are quite representative of typical video content.

4.1. Temporal video segmentation

Table 1 and Table 2 show the results obtained by the two temporal segmentation algorithms with the three sequences for cuts and gradual transitions. We note that the color histogram-based algorithm misses some cuts. This is due to the similar color distribution in some consecutive shots. It also has difficulties with the detection of gradual transitions. The use of a fixed threshold for the detection of a candidate gradual transition causes this failure. If we select a transition threshold so low that the gradual transition is detected, then too many false positives will be detected. Because of the adaptive thresholding used within the proposed segmentation algorithm (see Eq. (3), where σ_w and μ are adapted to the local characteristic of the activity measure signal), the gradual transition is detected correctly.

Algorithm	Number of key frames
IMM-JV	129
KF-NN	99
Histogram	47

Table 3. Number of key frames extracted from the three sequences by the different algorithms.

The proposed algorithm has a relatively high number of false positives. This is due to the lack of texture in some scenes, and thus only a small number of feature points is extracted. This results in an unreliable activity measure.

4.2. Key frame extraction

It is hard to evaluate the key frame extraction step, because it is quite a subjective matter. The evaluation consists of checking whether the extracted key frames capture the salient content of the sequence or not. We will also give the number of the key frames extracted by the different algorithms.

Table 3 gives the number of key frames extracted by the different algorithms. The color histogram-based algorithm fails to represent the content of the sequence, whereas the proposed algorithm achieves adequate abstraction and generate good key frames. In a shot which contains a transition through a camera operation (i.e. a fast pan after which the camera remains still), the extracted key frames obtained by the proposed algorithm are the best representative of the content (see Figure 2). The high accuracy of the key frame extraction step of the algorithm can be explained by the ability of the IMM-JV tracking algorithm to adapt itself to different camera operations.

(a) Color histogram

(b) IMM-JV

Fig. 2. Key frames extracted from the shot between frame 2465 and 2584 in the *news* by the two algorithms.

5. CONCLUSIONS

We have presented a content-based algorithm for video summarization. It relies on the tracking of feature points to compute an activity measure. The segmentation of the activity measure into stationary segments while detecting abrupt and gradual changes allows us to extract key frames and to detect shot boundaries.

The evaluation of the different fundamental elements of the summarization algorithm have shown that the temporal segmentation algorithm yields good results. Both cuts and gradual transitions are detected. The key frame extraction step achieves adequate abstraction and produces good key frames. However, the key frame extraction step still requires more objective evaluation and user studies.

6. REFERENCES

[1] J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Proc. SPIE Int. Symp. Elec. Imaging: Storage and Retrieval for Image and Video Databases*, San Jose, USA, 1996, pp. 170–179.

[2] H. J. Zhang, J. Wu, 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] R. Zabih, J. Miller, and K. Mai, "Feature-based algorithms for detecting and classifying scene breaks," in *Proc. ACM Multimedia Conf.*, San Fransisco, USA, November 1993, pp. 189–200.

[4] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[5] P. Bouthemy, M. Gelgon, and F. Ganansia, "A unified approach to shot change detection and camera motion characterization," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 9, no. 7, pp. 1030–1044, October 1999.

[6] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," in *Proc. Visual Database Systems*, 1992, Ed., 1992.

[7] W. Wolf, "Key frame selection by motion analysis," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, 1228–1231 1996, vol. 2.

[8] D. Zhong, H. J. Zhang, and S. Chang, "Clustering methods for video browsing and annotation," in *Proc. SPIE Int. Symp. Elec. Imaging: Storage and Retrieval for Image and Video Databases*, San Jose, USA, 1996, pp. 239–246.

[9] M. Irani and P. Anandan, "Video indexing based on mosaic representations," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 905–921, 1998.

[10] Y. Abdeljaoued and T. Ebrahimi, "Feature point extraction using scale-space representation," in *Proc. IEEE Int. Conf. Image Processing*, Singapore, October 2004.

[11] Y. Abdeljaoued, *Feature Point extraction and Tracking for Video Summarization and Manipulation*, Ph.D. thesis, EPFL, Swiss Federal Institute of Technology, LTS-DE, 1015 Lausanne, 2001.

[12] Y. Abdeljaoued, T. Ebrahimi, and C. Christopoulos, "A new algorithm for shot boundary detection," in *Proc. 10th European Signal Processing Conf.*, Tampere, Finland, September 2000, pp. 151–154.

[13] S. W. Roberts, "Control chart tests based on geometric moving average," *Techno-metrics*, vol. 1, no. 3, pp. 239–250, 1959.

[14] H. J. Zhang, A. Kankanhalli, and S. W. Smoliar, "Automatic partitioning of full-motion video," *Multimedia Systems*, vol. 1, no. 1, pp. 10–28, 1993.