Stevica CVETKOVIC, Marko JELENKOVIC, Sasa V. NIKOLIC

University of Nis, Faculty of Electronic Engineering, Nis, Serbia

Video summarization using color features and efficient adaptive threshold technique

Abstract. *Most of the methods for video summarization relay on complicated clustering algorithms that makes them too computationaly complex for real time applications. In this paper we propose an efficient approach for video summary generation that does not relay on complex clustering algorithms and does not require frame length as a parameter. Our method combines MPEG-7 Color Layout Descriptors with adaptive threshold technique to detect shot boundaries. For each shot a keyframe is extracted and similar keyframes are eliminated in a simple manner. A MOS measure evaluation on a standard dataset show that the method produces video summaries of highest visual quality.*

Streszczenie. *W artykule zaproponowano nieskomplikowany algorytm do tworzenia skrótów materiałów wideo. Metoda łączy w sobie deskryptor warstwy koloru MPEG-7 z techniką progu adaptacyjnego, co pozwala na wykrywanie granic stopklatki. Dla wielu takich samych lub podobnych klatek, pozostawiana jest tylko jedna z nich. (**Tworzenie skrótu wideo z wykorzystaniem właściwości koloru oraz technik progu adaptacyjnego**).*

Keywords: Video summary, MPEG-7, adaptive threshold.
Słowa kluczowe: skrót wideo, MPEG-7, próg adaptacyjny.

Introduction

Enormous popularity of the Internet video repository sites like YouTube or YahooVideo caused increasing amount of the video content avilable over the Internet. In such a scenario, it is neccesary to have efficient tools that allow fast video browsing. This tools should provide concise representation of the video content as a sequence of still or moving pictures - i.e. video summary. Generally there are two types of video summaries [1]: static video summary and dynamic video skimming. The aim of the static video summarization is to select those video frames that would be the most informative and concise representation of the original video. Dynamic video summaries represent collection of short video clips that contains only important video shots. Since static video summaries are the most common technique used in practical video browsing applications, we focused our research on static video summarization.

Most of the existing work on static video summarization is performed by clustering similar frames and selecting representatives per clusters [2, 3, 4, 5, 6]. A variety of clustering algortihms were applied such as: Delaunay Triangulation [2], k-medoids [3], k-means [4], Furthest Point First [5, 6]. etc. Although they produce acceptable visual quality, the most of these methods relay on complicated clustering algorithms, applied directly on features extracted from sampled frames. It makes them too computationaly complex for real-time applications. Another restriction of these approaches is that they require the number of clusters i.e. representative frames to be set a priori.

The contribution of this paper is to propose a fast and effective approach for video summary generation that does not relay on complicated clustering algorithms and does not requre length (number of summary frames) as a parameter. Our approach is based on low-level frame color features combined with adaptive threshold technique. The evaluation of the method is done on a standard video dataset by comparing the visual quality of results with state of the art methods. Our approach achieved high visual quality of produced video summaries avoiding computationally intesive operations.

In the rest of the paper we will first give overview of the proposed method followed by detailed description of every step. Then, the experimental evaluation using our implementation will be presented. Conclusions are drawn in the last section.

Overview of the proposed method

We proposed an approach which is based on several efficient video processing procedures (Fig. 1). At first, video frames are sampled in order to reduce further computational burden. Then, efficient MPEG-7 Color Layout Descriptor (CLD) is extracted on pre-sampled video frames. These features are deployed for shot boundary detection using adaptive threshold approach. The method sequentially computes local threshold of frame differences inside the sliding window. Shot changes are detected at places where the frame difference is maximal within the window and larger than the local threshold. Then, a representative keyframe is extracted for each shot and similar keyframes are eliminated in a simple manner. As a final result the most informative keyframes are selected as a video summary. In the rest, detailed description of every step of the method is presented.

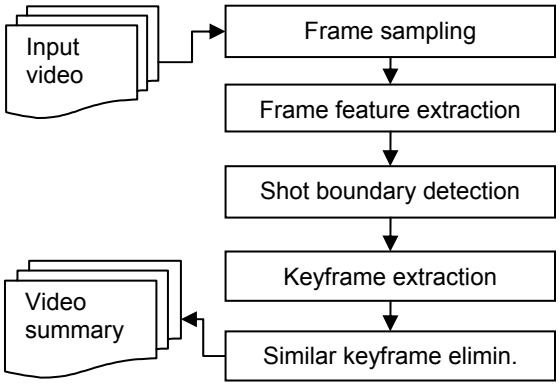

Fig. 1. Our approach for video summarization

Frame sampling

To avoid redundant frame comparisons, frames should be sampled using appropriate strategy. The sampling is based on the observation that there is a visual redundancy among certain number of frames per second. We applied the most common strategy by uniform sampling with a fixed sampling rate. Although coarse sampling rate could significantly reduce processing time of forthcoming steps, too high values could produce unsatisfied final results. Therefore, we made a compromise and sampled every 10th frame of input video, for further processing steps.

Frame feature extraction for shot boundary detection

Frame feature extraction is a crucial part of a keyframe extraction algorithm which directly affects performances of the algorithm. The visual feature which is optimal for video processing applications, should satisfy several main requirements [6]:

- Robustness: A frame feature should stay largely invariant for the same frame content under various types of transformations, such as format conversion or content editing.
- Discriminability: Features extracted for different video frames should be distinctly different.
- Compactness: A frame feature should be of insignificant size, comparing to the data size of the original video frame.
- Low complexity: The algorithm for the feature extraction should have computational complexity as low as possible.

Although many visual features are proposed in the literature [7, 8], comparative evaluations [9, 10] led us to conclusion that the Color Layout Descriptor (CLD) [9, 10] is an optimal choice in terms of previous requirements. CLD has been designed to efficiently and compactly represent spatial layout of colors inside image. It is obtained by extracting local representative colors of 64 image blocks in YCbCr color space, and compressing them using Discrete Cosine Transformation (DCT). The descriptor is characterized by low computational complexity, very compact representation and invariance to resolution changes. The extraction process starts with image partitioning step, where each RGB channel is divided into 8x8=64 non-overlapping blocks, to guarantee resolution invariance. Then, a single representative color is computed for each block by simple pixel averaging that provides sufficient accuracy with minimal computation costs. In the next step, color space is converted to YCbCr and each color channel matrix is transformed by DCT to obtain three sets of 64 DCT coefficients. Finally, a zig-zag scanned DCT coefficients are concatenated into a feature vector consisting of certain number of the most representative Y, Cb and Cr coefficients. Although a sufficient number of DCT coefficients to form a feature vector could vary in range from 12 to 64, our rough experiments have shown that 12 dimensional CLD features are sufficient for video shot boundary detection.

Let $\mathbf{f}_i^{CLD}$ denotes CLD feature extracted from a video frame at position i in a sampled video sequence: $\mathbf{f}_i^{CLD} = (f_{Y1}^{CLD},...,f_{Y6}^{CLD},f_{Cb7}^{CLD},f_{Cb8}^{CLD},f_{Cb9}^{CLD},f_{Cr10}^{CLD},f_{Cr11}^{CLD},f_{Cr12}^{CLD})$

A 12-dimensional CLD feature is extracted for each sampled frame of a video and used for further processing steps.

Shot boundary detection

An important preprocessing step for many video analysis tasks is shot boundaries detection (extraction). The goal of this step is to divide the input video into shots, where a shot is defined as a sequence of video frames taken continuously by one camera. Shot boundaries are detected at positions where shot transition occurs. Our shot boundary detection method uses adaptive threshold value calculated within sliding window of a sampled video. Instead of comparing extracted CLD features directly to local thresholds, the difference of successive frame features is first calculated and normalized by the maximum value in the video. Difference values that are maximal within the window and larger than local thresholds are considered as shot changes.

Let a normalized Euclidean distance of successive CLD frame features is denoted as d_i^{CLD}:

$$d_i^{CLD} = \left\| \frac{\mathbf{f}_i^{CLD} - \mathbf{f}_{i-1}^{CLD}}{\max(\mathbf{f}_i^{CLD})} \right\| \tag{1}$$

The value d_i^{CLD}, in the middle position within the sliding window of size $2m+1$, will be recognized as a shot change if the following two conditions are fulfilled:

- The d_i^{CLD} is the maximum within the sliding window, i.e. $d_i^{CLD} = \max_{i-m \le j \le i+m} (d_j^{CLD})$
- The d_i^{CLD} is larger than the local threshold T_i.

For adaptive threshold computation, we propose to use following approach:

$$T_i = \alpha \cdot \mu_i + T_{const} \tag{2}$$

$$\mu_i = \frac{d_{i-m}^{CLD} + d_{i-(m-1)}^{CLD} + ... + d_{i+m}^{CLD}}{2m+1} \tag{3}$$

where α and T_{const} are empirically determined parameters, while μ_i represents mean d_i^{CLD} value within sliding window. After a shot boundary is detected, m samples are elapsed before the next examination.

Fig. 2 shows an example for distribution of d_i^{CLD} distances along time. Filled squares are used to mark detected shot transitions. Note that introduction of T_{const} component is crucial in case of arrays of small distance values (frames between 150 and 200). In this case local peaks do not necessary represent shot boundaries, although they overcome first component value $\alpha \cdot \mu_i$ of the threshold.

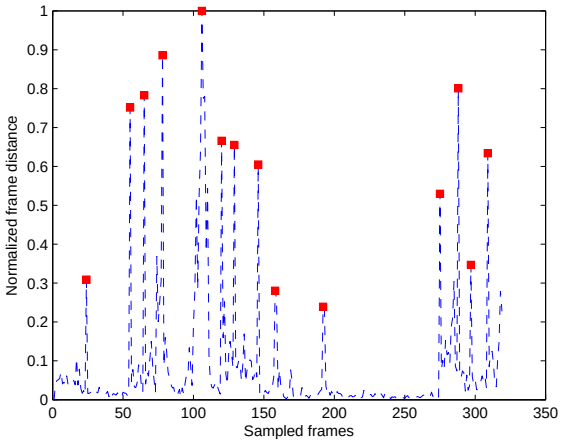

Fig. 2. Normalized frame distances between succesive sampled frames.

The proposed method was largely inspired by works of [13] and [14]. It sublimates fundamental priciples of these two methods into a novel technique for efficient shot boundary detection.

Keyframe selection and video summary generation

A detected video shot can be compactly represented by a single frame that is called the keyframe. Typically a

keyframe is choosen to be the first or the last frame of a shot, although some methods use any single frame inside a shot. In our approach, we selected a middle frame of a shot as the keyframe.

After keyframes are extracted, there could be similar keyframes that appear at different temporal positions in the video. In order to eliminate similar keyframes, and leave only the most informative one, we applied a simple technique. First, a distance matrix is calculated between all keyframes. Then, frames with distance less than predifined threshold T_{dist} are removed from the summary. Finally, the most representative keyframes are remained in the form of video summary. Fig. 3 presents results of our method before and after similar keyframe elimination.

Fig. 3. Extracted keyframes before (1st and 2nd row) and after elemination of similar keyframes (3rd row).

Experimantal evaluation

The evaluation is carried out using a subset of videos downloaded from the Open Video Project [15], the same one as used in [5]. It contains 7 videos of a different type with respect to color and motion, encoded using MPEG-1 with resolution 352x240 pixels. Following video sequences were used: *(1) A new Horizon 1, (2) Ocean floor Legacy 8, (3) Drift ice 8, (4) The voyage of the Lee 15, (5) Exotic Terrane 1, (6) Hurricane Force 3 and (7) Digital Jewelery*. For experimental evaluation we implemented the complete algorithm using Matlab. The following parameters were empirically determined and used for all tests: $m = 8$, $\alpha = 1.5$ and $T_{const} = 0.15$.

The quality evaluation is performed by measuring a Mean Opinion Score (MOS), using the same procedure as [5]. Our results are compared with reported results of Open Video Project [15], VISTO [5], Delaunay Triangulation (DT) [2] and k-means.

Fig. 4 presents comparative results of MOS measure obtained on test videos. The quality of the video summary is scored on a scale 1 to 5 (1 = bad, 2=poor, 3=fair, 4=good, 5=excellent). It can be concluded that our method achived one of the best scores for each of test videos.

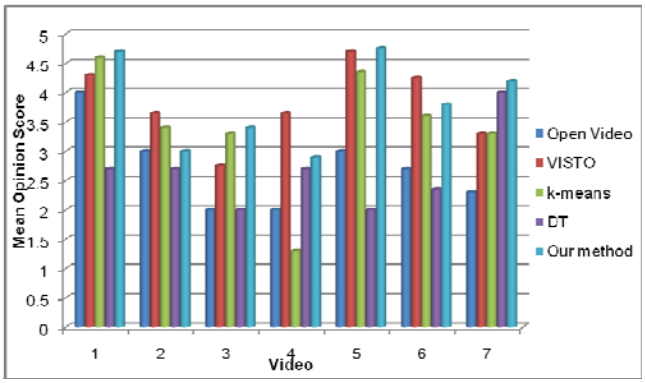

Fig. 4. MOS evaluation of test videos

In order to demonstrate visual quality of our results compared to the other methods, we presented video summaries of one video clip used for testing in Fig. 5. As the MOS results reported, it can be observed that the output of our method, VISTO and k-means achieved the best visual quality of produced video summaries.

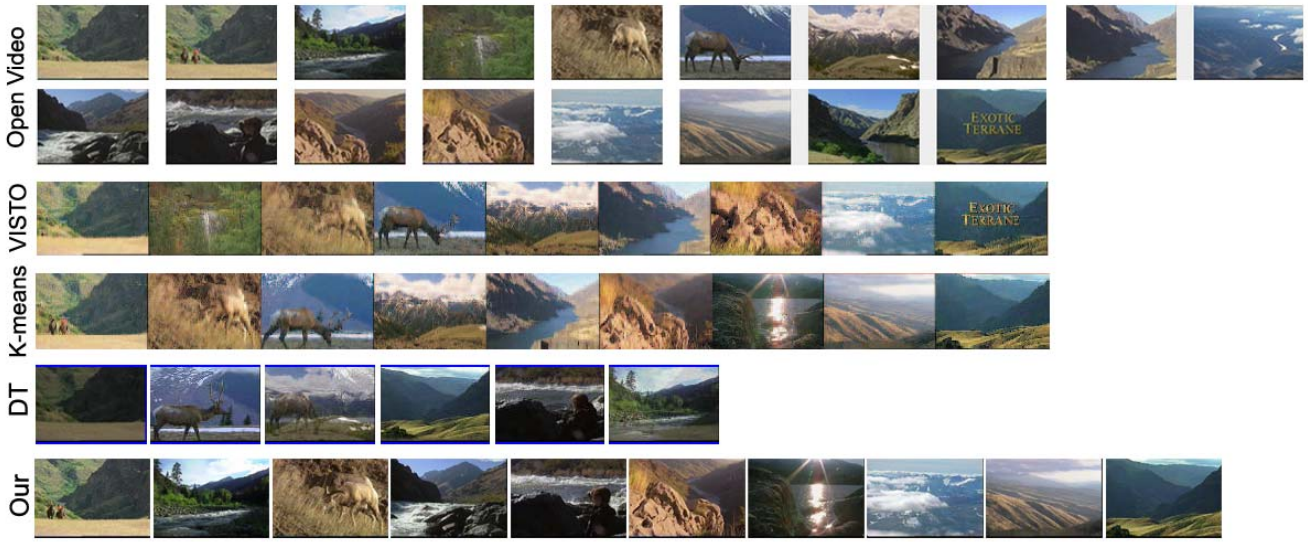

Fig. 5. Visual comparasion of video summaries generated using several methods on video clip 5 ("Exotic Terrane 1").

Conclusion

In this paper we proposed an efficient method for video summary generation. It combines MPEG-7 CLD with adaptive threshold technique to detect shot boundaries. For each shot a keyframe is extracted and similar keyframes are eliminated in a simple manner to produce concise and informative video summary. The method was evaluated on a standard video sequences and compared with state of the art methods. A MOS measure comparison showed that the method produces video summaries of high visual quality without using computationally expensive clustering techniques. In addition, it does not require length of the summary as a parameter. It makes our method suitable for real time video processing of deverse types of compresed videos, including modern H.264/SVC videos [16].

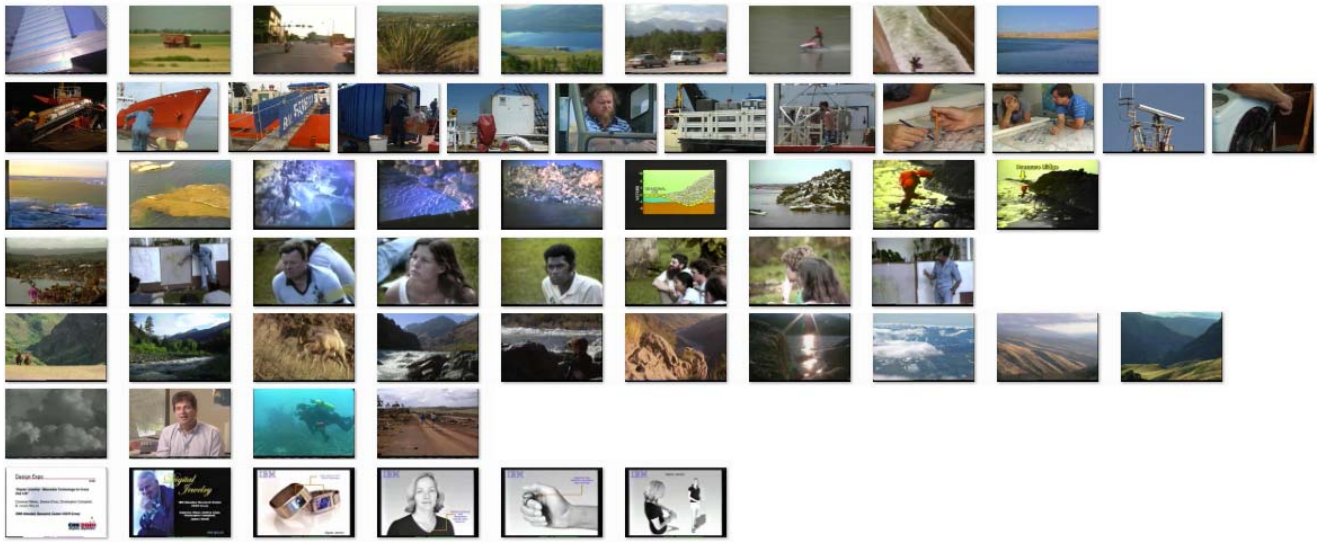

Fig. 6. Preview of generated summaries of seven test video sequences (one video per row)

REFERENCES

[1] Truong, B.T., Venkatesh, S., Video abstraction: A systematic review and classification. *ACM Transactions on Multimedia Computing Communications and Applications*, 3 (2007), No. 1, 1-37.

[2] Mundur, P., Rao, Y., Yesha, Y., Keyframe-based video summarization using Delaunay clustering. *International Journal on Digital Libraries*, 6 (2006), No. 2, 219–232.

[3] Hadi, Y., Essannouni, F., Thami, R. O. H., Video summarization by k-medoid clustering. Proceedings of the ACM symposium on Applied Computing - SAC '06, (2006) New York, USA: ACM Press, 1400–1401.

[4] De Avila, S. E. F., Lopes, A. P. B., da Luz, A., Albuquerque Araújo, A., VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, 32 (2011), No. 1, 56–68.

[5] Furini, M., Geraci, F., Montangero, M., Pellegrini, M., VISTO: visual storyboard for web video browsing. Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR) (2007), 635–642.

[6] Furini, M., Geraci, F., Montangero, M., Pellegrini, M., STIMO: STIll and MOving video storyboard for the web scenario. *Multimedia Tools and Applications*, 46 (2007), No. 1, 47–69.

[7] Lee, S., and Chang D.Y., Robust Video Fingerprinting for Content-Based Video Identification. *IEEE Transactions on Circuits and Systems for Video Technology*, 18 (2008), No. 7, 983-988.

[8] Świerz, M., Iwanowski, M., Image features based on morphological class distribution functions and its application to binary pattern recognition, *Przegląd Elektrotechniczny*, 88 (2012), No. 2, 1, 132-135.

[9] Eidenberger, H., Statistical analysis of content-based MPEG-7 descriptors for image retrieval. *Multimedia Systems*, 10 (2004), No. 2, 84–97.

[10] Deselaers, T., Keysers, D., Ney, H., Features for Image Retrieval: A Quantitative Comparison. DAGM Symposium Symposium for Pattern Recognition, (2004), 228–236.

[11] Manjunath, B. S., Ohm, J. R., Vinod, V. V., Yamada, A., Color and Texture descriptors, *IEEE Trans. Circuits and Systems for Video Technology*, 11 (2001), No. 6, 703–715.

[12] Manjunath, B. S., Salembier, P., Sikora, T., *Introduction to MPEG-7.* (2007), San Francisco CA: Wiley.

[13] Yusoff, Y., Christmas, W., Kittler, J., Video Shot Cut Detection Using Adaptive Thresholding, British Machine Vision Conference, (2000).

[14] Yeo, B. L., Liu. V., Rapid scene analysis on compressed video. *IEEE Tranactions on Circuits and Systems for Video Technology*, 5 (1995), No. 6, 533–544.

[15] The Open Video Project Online: http://www.openvideo.org

[16] Zhang, X-W., Zhu, T., Shu, X-C., Xiao, Q., Analysis and evaluation for the Scalability Mechanisms of H.264/SVC, *Przegląd Elektrotechniczny*, 87 (2011), No.7, 235-239.

***Authors**:*
Stevica Cvetkovic,MSc, email: stevica.cvetkovic@elfak.ni.ac.rs;
Marko Jelenkovic, MSc, email: virtus@elfak.rs;
Prof. Sasa V. Nikolic,PhD, email: sasa.nikolic@elfak.ni.ac.rs;
Faculty of Electronic Engineering, University of Nis, Aleksandra Medvedeva 14, 18000 Nis, Serbia.