A COMPLETE FRAMEWORK FOR VIDEO TEMPORAL SEGMENTATION

Ruxandra ȚAPU[1], Bogdan MOCANU[2], Teodor PETRESCU[3]

În acest articol se propune un algoritm complex de segmentare a fluxurilor video in scene, care debutează cu identificarea schimbărilor de plan (acuratețea detecției peste 95%, precizia de identificare peste 90% şi timpul de calcul redus) şi continuă cu crearea unui rezumat static al secvenței de imagini prin folosirea metodei originale de extragere a imaginilor cheie "în salturi"(cu 27% mai rapidă decât o tehnică de referință pentru performanțe echivalente). În final, prin utilizarea clusterelor, ce integrează un set de constrângeri temporale, imaginile cheie selectate sunt utilizate pentru gruparea planelor video în scene (acurateţe şi precizie de peste 73.7% şi 84.6% respectiv).

In this paper we propose a complete high level segmentation algorithm of video flows into scenes. In the first phase we identify shot boundaries with an accuracy of more than 95% in recall and 90% in precision rates at reduced computational time. In a second stage, a storyboard is created by using a leap keyframe extraction method, at 27% faster rate than the reference method for equivalent performances. Finally, the detected keyframes feed a shot clustering algorithm which integrates a set of temporal constraints that generates video scenes with an average precision and recall rates of 73.7% and 84.6% respectively.

Keywords: shot boundary detection, keyframe, shot merging, clustering

1. Introduction

The continuous growth of the available information stored, transmitted and exchanged over the Internet is challenging the scientific community in developing new and efficient tools to reliable access, browse and retrieve multimedia content. Existing commercial and industrial video search engines are currently based solely on textual annotations, which consist of attaching some keywords to each individual item in the database. However, such an approach is tedious in terms of the manual annotation effort required. In addition, the process is strongly influenced by: the subjective interpretation of the content since various people

[1] Assist., Faculty of Electronics, Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania, e-mail: ruxandra_tapu@comm.pub.ro
[2] Assist., Faculty of Electronics, Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania, e-mail: bcmocanu@comm.pub.ro
[3] Prof., Faculty of Electronics, Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania, e-mail: teodor.petrescu@electronica.pub.ro

may perceive differently the semantics of a same image/video document and thus associate different keywords with the content. Finally, the multi-lingual aspects cannot be treated in a straightforward manner.

Moreover, when considering the specific issue of video indexing, the description exploited by the actual commercial searching engines (*e.g.,* Youtube, Daily Motion, Google Video) adopt a monolithic and global video description, treating each document as a whole. Such an approach does take into account neither the informational and semantic richness, specific to video documents, nor their intrinsic spatio-temporal structure. As a direct consequence, the resulting granularity level of the description is not sufficiently fine to allow a robust and precise access to user-specified elements of interest (*e.g*. objects, scenes, events…). Within this framework, video segmentation and structuring represents a key and mandatory stage that needs to be performed prior to any effective description/classification of video documents.

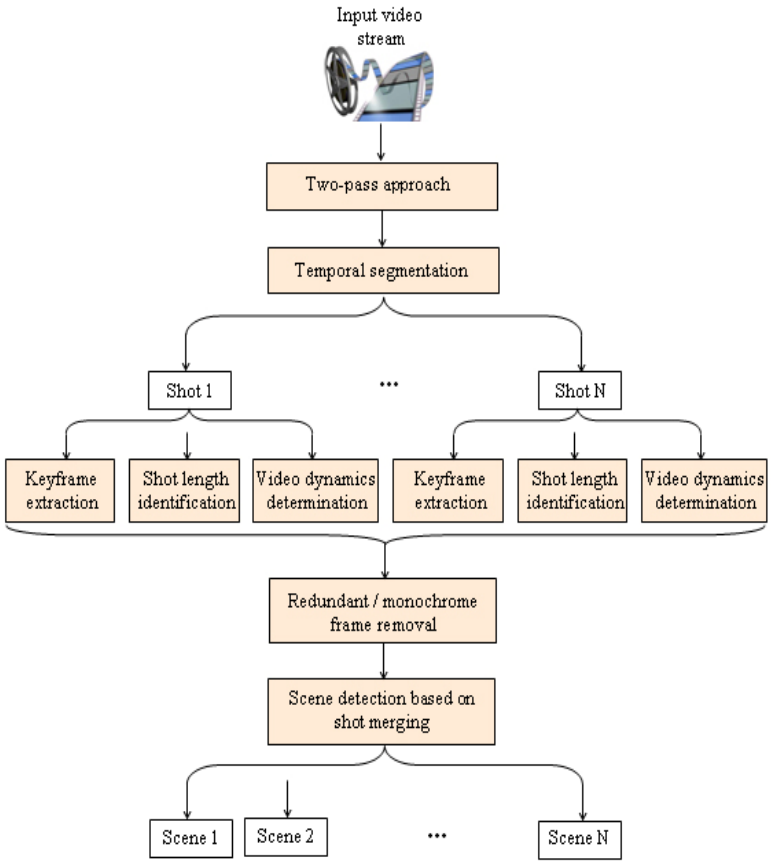

Fig. 1. Scene change detection algorithm

The video segmentation process can be divided into the following phases: shot boundary detection (or temporal segmentation into shots - a sequence of successive frames taken from the moment a camera starts recording until it stops), automatic image sequence abstraction that provide concise information, with static or moving images, about the video content while conserving the original message and shot clustering based on similarity constraints in order to construct semantically pertinent scenes (Fig. 1).

Traditionally, a scene is defined as group of video shots that are correlated according to the semantic content. A scene needs to respect three continuity rules: space, time and action. [1]. However, in some circumstances these constraints may not hold, as in the case of scenes with large camera/object motion. Elaborating methods for pertinent and automatic scene identification is still an open issue of research.

The rest of this paper is organized as follows. In Section II we briefly present the shot boundary detection system adopted. After a brief recall of the most important video abstraction techniques, Section III describe the keyframe selection procedure as well as the temporally-constrained shot grouping algorithm proposed. In Section IV we present and discuss the experimental results obtained. Finally, Section V concludes the paper and opens some perspectives of future work.

2. Temporal segmentation system

We start our analysis by dividing the video into shots using our previous work presented in [2]. The algorithm is an enhanced version on a graph partition model, combined with a non-linear scale-space filtering. In order to reduce the computational complexity an initial segmentation step of the input stream is performed with the help of a sliding window that selects a constant number of N frames from the original video signal centered on the current frame.

In the shot boundary context each frame of the video sequence is represented as a vertex in a graph structure, connected with each others, by edges. The weight of each edge is computed as the chi-square distance between color histograms represented in the HSV color space. For segmentation, we retained the min–max algorithm [3] that aims at minimizing the cut while maximizing the association measures.

In our method the detection efficiency is further increased by performing an analysis on the first order discrete derivatives of the local minimum vector, which allows deriving a relative change ratio measure, instead of an absolute one. In addition, a scale space evaluation of discrete derivative at different resolution levels is determined in order to remove false alarms introduced by large object displacement or camera movement.

We focused next in reducing the global computational complexity by implementing the two-pass approach also introduced in [2]. In a first step, the algorithm detects time intervals which can be reliably considered as belonging to the same shots. Abrupt transitions considered as certain are also detected in this stage. In a second step, the optimal multiresolution graph partition algorithm is further performed only for the remaining uncertain time intervals.

3. Keyframe selection and shot merging

Video abstraction techniques aim at providing concise representation of a multimedia document, represented with the help of still or moving images, while conserving the original message. Two different types of summaries can be used in order to characterize image sequences: static storyboard and video skimming. The fist type also encountered in the scientific literature as still abstract is given to a set of representative images (known as keyframes) selected from the original movie that represent its informational content. Video skimming, also called moving abstract, is a collection of image sequences incorporating several audiovisual cues to represent the content in a video stream condensed and succinct and can be further classified into: highlights and summary sequence. In this paper, we focus our attention on developing a new technique of selecting salient images (key frames) from all the frames of an original video in order to obtain a representative video summary.

3.1. *Keyframe selection*

The main objective of keyframe selection is to determine, for each detected shot, a set of images that might represent in a pertinent manner the associated content. The keyframe selection process is highly important for video indexing applications. On one hand, keyframes can be exploited for video summarization purposes [4, 5]. On the other hand, they may be further used for high level shot grouping into scenes [6, 7].

One of the first attempts to automate the extraction process was to choose as a key frame the first, the middle or the last picture appearing after each detected shot boundary [8] or a random image within a shot. However, while being sufficient for stationary shots, one frame does not provide an acceptable representation of the visual content in dynamic sequences. Therefore, it is necessary to implement more complex methods that can be able to adapt the number of key frames to the visual content variation of the corresponding shot [9].

The challenge in automatic key-frames extraction is given by the necessity of adapting to the underlying semantic content, maintaining, as much as possible, the original message while removing all redundant information. In [10] the extraction process relay on the color and motion features. When selecting the first frame encountered after a shot boundary as a key frame, as presented in [10], [11]

a possible disadvantage is given by the probability of that frame belonging to a transitional effect, reducing very much it's representative power. When a temporal segmentation is performed on a video stream, for gradual transitions, in most of the cases a shot change is identified within the actual transition. So, selecting the first frame afterwards is not an optimal solution.

A clustering algorithm is the natural solution to solve the problems described above. Even so all clustering algorithms have weak points related to the threshold parameters which control the cluster density and the computational cost.

A mosaic-based approach can generate, in an intuitive manner, a panoramic image of all informational content existed in a video stream. The summarization procedure in this case is based on the assumption that there is only one dominant motion among all the others various object motions found in the sequence [12] [13]. Mosaic-based representations of shot / scene include more information and are visually richer than regular key frame approaches. However, creating mosaics is possible solely for videos with specific object or camera motion, such as pan, zoom or tilling. In the case of movies with complex camera effects such as a succession of background/foreground changes, mosaic-based representations return less satisfactory results due to physical location inconsistency. Furthermore, mosaics can blur certain foreground objects and thus, the resulted image cannot be exploited for arbitrary shape object recognition tasks.

In our case, we have adopted a key-frame representation. Initially, a first keyframe is selected for each shot. By definition, this frame is located at *N* (*i.e.,* the window size used for the shot boundary detection) frames away after a detected transition, in order to make sure that the selected image is not part of a gradual effect.

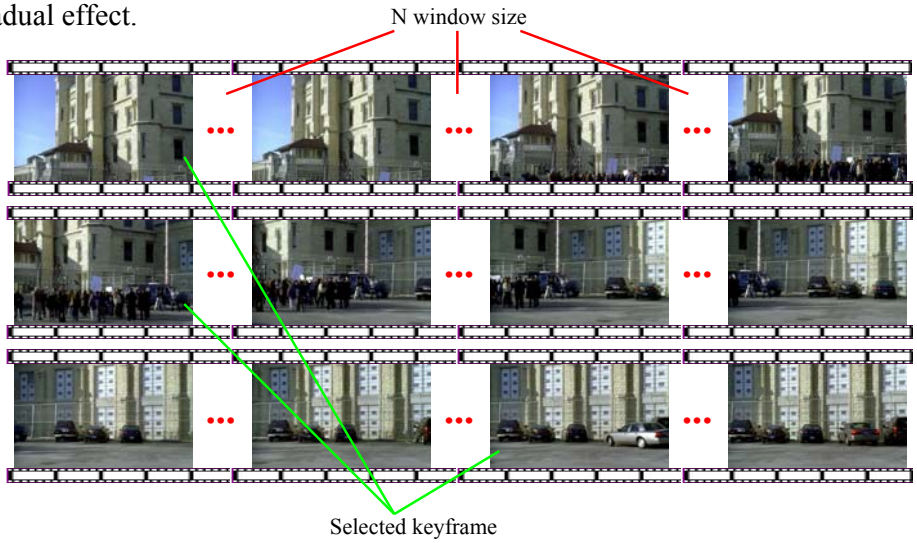

Fig. 2. Keyframe selection based on the leap-extraction method

However, for dynamic shots with relatively important amount of motion, only one frame is not sufficient to adequately represent the content of a video shot. In this case, multiple images need to be selected, based on the visual variation, for a finer shot characterization. Here, we propose a leap-extraction method that involves only the frames spaced by integer multiples of the analysis window size (N) used (Fig. 2), instead of considering the whole set of frames of a shot as in the case of the reference method [7].

These frames are further compared with the existing shot keyframes set already extracted. Based on the amount of visual content variation (expressed as the chi-square distance in the HSV color space) a new frame is selected as a keyframe if its visual content differ significant (above a fixed threshold) from all the frames previously extracted.

Let us note that the analysis is performed only upon a reduced number of frames, by taking advantage of the shot boundary detection algorithm. By computing the graph partition within a sliding window, the method ensures that all the relevant information will be taken into account. Let us also note that the number of detected keyframes set per shot is not fixed *a priori*, but automatically adapted to the content of each shot. In Fig. 3 we presented a complex shot with lot of visual content variation for which our algorithm selects 3 keyframes.

Fig 3. Keyframe extraction and shot boundary detection

An additional preprocessing step eliminates all the monochrome and redundant images from the selected set of keyframes assuring that the story board captures all informational content of the original movie without any irrelevant images, which influence directly the representative power of the summary.

The key-frames thus extracted are then exploited for grouping shots into scenes.

3.2. ***Shot grouping into scenes***

The principle consists of clustering different shots into the same scene based on both visual and temporal criteria. More precisely, the algorithm can be described in the following steps:

Step 1: The first shot of a film is automatically assigned to the first scene.

Step 2: For each of the following shots (s_{cur}) the algorithm computes the visual dissimilarity with all the anterior scenes located at a temporal distance smaller or equal to the width of a temporal analysis window. We introduce a novel approach of adaptively determining the window size (*dist*) to depend on the video stream content variation and set proportional to the average number of frames per shot:

$$dist = \frac{Total\ number\ of\ \ frames}{Total\ number\ of\ \ shots} \tag{1}$$

We selected as similarity measure between two keyframes the chi-square distance of HSV color histograms. HSV space offers a serious of advantages compared to other color spaces: the color given by H and S is decupled from the intensity and we can manipulate the color independent. When using the HSV space the image representing regions are more homogenously and compact [16]. Furthermore in this space the color distance is easier to understand and interpret.

Step 3: If a shot is identified to be similar to a scene it will be clustered in that scene, together with all the intermediary shots between them (Fig. 4). A novel shot - scene similarity is developed:

$$\forall\ S_k \qquad SceneShotSim(s_{cur}, S_k) = \frac{n_{matched}}{n_{k,p} \cdot N_k \cdot n_{cur}} \tag{2}$$

where N_k is the number of shots included in scene S_k, n_{cur} is the number of keyframes of the considered shot and $n_{matched}$ represents the number of similar (above a threshold T_g) keyframes from the current shot s_{cur} and analyzed scene S_k while $n_{k,p}$ is the number keyframe included in shot s_p.

Finally, the current shot s_{cur} is identified to be similar to the scene S_k if:

$$SceneShotS\ im(S_k, s_{cur}) \geq 0.5 \tag{3}$$

In this case, the current shot s_{cur} will be clustered in the scene S_k. In the same time, all the shots between the current shot and the scene S_k will be also affected to scene S_k and marked as *neutralized*. Let us note that the scenes to which initially belonged such neutralized shots disappear (in the sense that they are merged to the scene S_k). The list of detected scenes is then updated.

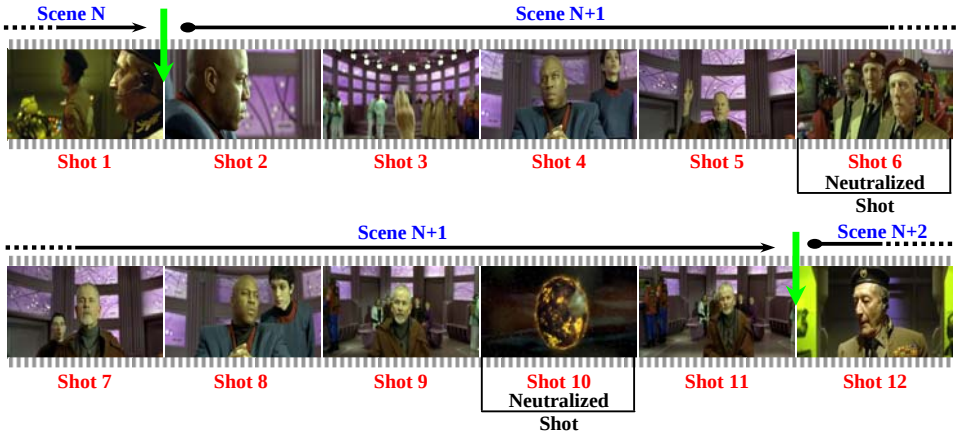

Fig. 4. Shots grouping based on visual contain similarity

Step 4: If a shot contains a keyframe very similar (with a similarity at least two times bigger than the grouping threshold T_g) with any other keyframe of a scene located at a temporal distance inferior to parameter *dist*, it will be grouped in the scene. In this step, key-frames from both neutralized and non-neutralized shots will be taken into account

Step 5: If a shot is not found to be similar with any scene in the above conditions, a new scene associated to the considered shot will be created.

Step 6: At the end, scenes containing only one shot are attached to the previous scenes. In the case of the first scene, this is grouped with the following one.

We developed a new method of establishing the grouping threshold T_{group} adaptively depending on the input video stream visual content variation as the average chi-square distance between the current keyframe and all anterior keyframes located at a temporal distance smaller then *dist*.

4. Experiments and result

In order to validate our method we have considered a set of videos from the TRECVID 2001 and 2002 evaluation campaigns (http://trecvid.nist.gov/), which are freely available over the Internet (www.archive.org and www.open-video.org). The selected videos are documentaries that vary in the production date

and style, and include various types of transitions and motions caused by both large object displacement and camera movement.

In addition, we have included in the test set 7 sitcoms and Hollywood movies for which the scene boundaries have been manually identified by human observers. The following extra videos in the database have been considered: Seinfeld (SF), A Beautiful Mind (BM), Terminator 2 (T2), Top Gun (TG), Gone in 60 seconds (G-60), Golden Eye (GE). Such films include lots of action and scene changes and have been used previously for the evaluation of the scene detection algorithms introduced in [7].

Table 1

Computation time and gain for classic and leap keyframe extraction strategy.

	Video title	Video duration (s)	Leap extraction Time (s)	Classical extraction Time (s)	Gain (%)
TRECVID	NAD55	871	153	186	17.7
	NAD57	417	72	98	26.5
	NAD58	455	102	140	27.1
	UGS01	1337	292	356	17.9
	UGS04	1620	328	412	20.4
	UGS05	1297	299	368	23.1
	UGS09	1768	355	493	27.9
	23585a	617	121	154	27.2
	10558a	833	158	192	21.5
	06011	997	186	217	16.6
	08401	423	78	93	19.2
	08024	1119	242	309	27.7
Hollywood	SF	1313	312	388	19.6
	BM	8100	1978	2879	31.3
	T2	8812	2256	3105	27.3
	TG	6612	1523	2118	28.1
	G-60	6787	1605	2196	26.9
	GE	7878	2004	2779	27.8
	TOTAL	**51256**	**12064**	**16483**	**26.8**

Table 1 presents the computational time (The algorithms were run on a Pentium IV machine with 3.4 GHz and 2 Go RAM, under a Windows XP SP3 platform) necessary to extract representative frames for each detected shot, in both cases: when applying our leap-extraction strategy for selecting keyframes (*cf.* Section III), and the state-of-the-art method [7, 8] based on direct comparison of all adjacent frames inside a shot.

As evaluation metrics, we have considered the traditional recall (R) and precision (P) measures [1], defined as follows:

$$R = \frac{D}{D + MD} \quad \text{and} \quad P = \frac{D}{D + FA} \tag{4}$$

where *D* is the number of the detected shot boundaries, *MD* is the number of missed detections, and *FA* the number of false alarms. Ideally, both recall and precision should be equal to 100%, which correspond to the case where all existing shot boundaries are correctly detected, without any false alarm. In order to evaluate our scene extraction algorithm, we have considered the above dataset which include a variety of movies with different shooting styles. The ground truth for each new movie was set base on the reference paper [7].

The Hollywood videos allow us to make a complete evaluation of our method with the state of the art algorithms [7], which yield recall and precision rates at 60,1% and 79,3%. For this corpus, our precision and recall rates are of 73,7% and 84,6% respectively, which clearly demonstrates the superiority of our approach in both parameters (Table 2).

Table 2

Scene detection algorithm performance

	Video name	Grund truth Scenes	Rasheed *et at.*					Our proposed technique				
			D	F	MD	P (%)	R (%)	D	F	MD	P (%)	R (%)
Hollywood movies	SF	28	23	4	5	85.2	82.1	25	1	3	96.1	85.7
	BM	18	15	13	3	53.6	83.3	16	8	2	66.7	88.8
	T2	36	27	12	9	69.2	75	31	8	5	79.4	86.1
	TG	23	18	8	5	69.2	78.3	18	5	5	78.2	78.2
	G-60	39	29	28	10	50.9	74.4	32	15	7	68.1	82.1
	GE	25	22	22	3	50	88	21	14	4	60	84
Total		**169**	**134**	**87**	**35**	**60.1**	**79.3**	**143**	**51**	**26**	**73.7**	**84.6**

Finally, the experiments demonstrate the robustness of our method, regardless the film genre. In Fig. 5 we presented a video stream for which the scene detection was realized using the method proposed in this paper.

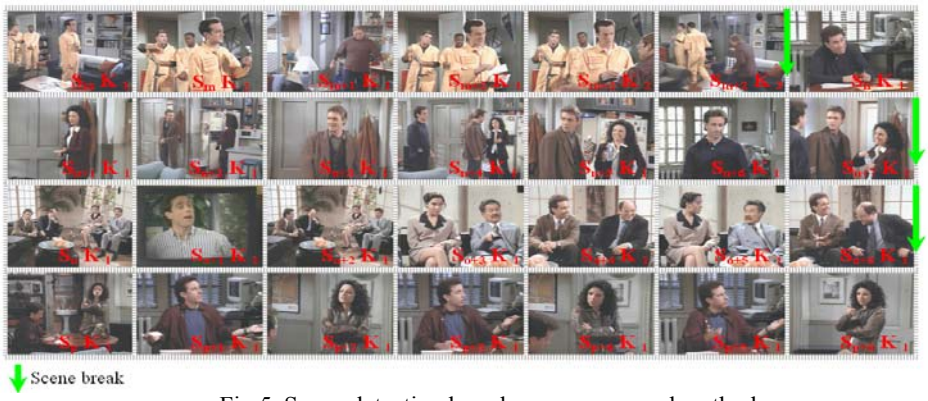

Fig.5. Scene detection based on our proposed method

5. Conclusion and perspectives

In this paper, we have introduced a new complete methodological framework for high level video temporal structuring and segmentation, with significant improvements in each step: in shot boundary detection, keyframe extraction and scene identification based on shot merging.

Our major contribution was directed on developing a new technique that develops fast static storyboards based on a set of representative images selected from each shot of the original movie and extracted based on the leap extraction method presented in Section III that capture the visual content variation. In the same section we introduced also a complete scene change detection method based on temporal constraints clustering. By implementing our keyframe selection method proposed in this paper we have increase the efficiency, our approach makes it possible to reduce the computational time with more than 26.8% at equivalent performances.

Regarding the shot merging into scenes strategy, by exploiting the observation that shots belonging to the same scene have similar visual features, we have adopted a grouping method based on temporal constraints that uses adaptive thresholds based on the input video dynamics. The technique is superior to state of the art methods and captures the global similarities rather the local ones. The experimental evaluation proposed validates our approach, with precision and recall rates around 73.7% and 84.6%, respectively.

For future work and perspective we will concern the integration of our method within a more general framework of video indexing and retrieval applications, including object detection and recognition methodologies. On one hand, this can further refine the level of description required in video indexing applications. On the other hand, identifying similar objects in various scenes can be helpful for the scene identification process. Finally, we intend to integrate within our approach motion cues that can be useful for both reliable shot/scene/keyframe detection and event identification.

R E F E R E N C E S

[1] *Z. Rasheed, M. Shah*, Scene detection in Hollywood movies and TV shows, IEEE Conference on Computer Vision and Pattern Recognition (CVPR'03), **vol. 2**, 2003, pp. 343–348

[2] *R. Tapu, T. Zaharia, F. Preteux*, A scale-space filtering-based shot detection algorithm, IEEE 26-th Convention of Electrical and Electronics Engineers in Israel, IEEEI 2010, pp.919-923.

[3] *C. Ding, X. He, H. Zha, M. Gu, H.D. Simon*, A Min-Max Cut Algorithm for Graph Partitioning and Data Clustering, Proc. Int'l Conf. Data Mining, Nov. 2001, pp. 107-114

[4] *M. Guironnet, D. Pellerin, N. Guyader, P. Ladret*, Video summarization based on camera motion and a subjective evaluation method, EURASIP Journal on Image and Video Processing, 2007, pp. 12 - 26

[5] *G. Ciocca, R. Schettini*, Dynamic key-frame extraction for video summarization, In Internet Imaging VI, **vol. 5670** of Proceedings of SPIE, , San Jose, California, USA, January 2005, pp. 137–142

[6] *A. Hanjalic*, *L.-Q. Xu*, Affective video content representation and modeling, IEEE Trans. Multimedia, **vol. 7**, no. 1, Feb, 2005, pp. 143-154

[7]. *Z. Rasheed, M. Shah*, Detection and representation of scenes in videos, IEEE Transactions on Multimedia, **Vol. 7(6)**, Dec. 2005, pp. 1097 – 1105

[8] *B. Shahraray, D.C. Gibbon*, Automatic generation of pictorial transcripts of video programs, in Proc. IS&T/SPIE Digital Video Compression: Algorithms and Technologies, San Jose, 1995, pp. 512–519

[9] *A. Hanjalic*, *H.J. Zhang*, An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis, IEEE Transactions on Circuits and Systems for Video Technology, **vol. 9**, no. 8, Dec. 1999

[10] *H. Zhang, J. Wu, D. Zhong, S.W. Smoliar*, An integrated system for content-based video retrieval and browsing, Pattern Recognit., **vol. 30**, no. 4, 1999, pp. 643–658

[11] *X. Fu, J. Zeng*, An Improved Histogram Based Image Sequence Retrieval Method, Proceedings of the International Symposium on Intelligent Information Systems and Applications (IISA'09), pp. 015-018, 2009

[12] *A. Aner, J.R. Kender*, Video summaries through mosaic-based shot and scene clustering, In Proc. European Conf. Computer Vision, 2002, pp. 388–402

[13] *A. Rav-Acha, Y. Pritch, D. Lischinski, S. Peleg. Dynamosaics*, Video mosaics with non-chronological time", In CVPR, Washington, DC, 2005, pp. 58–65

[14] *A. Hanjalic*, Shot-Boundary Detection: Unraveled and Resolved?, IEEE Trans. Circuits and Systems for Video Technology, **vol. 12**, no. 2, 2002, pp. 90-105

[15] *B. Truong, S. Venkatesh*, Video abstraction: A systematic review and classification, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), **vol. 3**, no. 1, pp. 3-es, February 2007

[16]. *H. Dasari, C. Bhagvati*, Identification of printing process using HSV color space, In ACCV, Lecture Notes On Computer Science, **vol. 3852**, pp. 692-701, Springer 2006.