Multi feature content based video retrieval using high level semantic concept

Hamdy K. Elminir
Head of communication dep Misr Academy for Engineering & technology.

Mohamed Abu ElSoud
Computer Science Dept.
Faculty of computers and information sciences.
Mansoura university, Egypt

Sahar F. Sabbeh
Information System Dept.
Faculty of computers and information sciences.
Benha university, Egypt

Aya Gamal
Computer Science Dept.
Faculty of computers and information sciences.
Mansoura university, Egypt

Abstract- Content-based retrieval allows finding information by searching its content rather than its attributes. The challenge facing content-based video retrieval (CBVR) is to design systems that can accurately and automatically process large amounts of heterogeneous videos. Moreover, content-based video retrieval system requires in its first stage to segment the video stream into separate shots. Afterwards features are extracted for video shots representation. And finally, choose a similarity/distance metric and an algorithm that is efficient enough to retrieve query – related videos results. There are two main issues in this process; the first is how to determine the best way for video segmentation and key frame selection. The second is the features used for video representation. Various features can be extracted for this sake including either low or high level features. A key issue is how to bridge the gap between low and high level features. This paper proposes a system for a content based video retrieval system that tries to address the aforementioned issues by using adaptive threshold for video segmentation and key frame selection as well as using both low level features together with high level semantic object annotation for video representation. Experimental results show that the use of multi features increases both precision and recall rates by about 13% to 19 % than traditional system that uses only color feature for video retrieval.

Keywords -
Content based video retrieval, High level semantic features, video partitioning, feature extraction, video parsing, and object annotation.

I. Introduction

The value of video is partially due to the fact that significant information about many major aspects of the world can only be successfully managed when presented in a time-varying manner. Today, a great challenge in information retrieval is to manage various nontraditional types of data, such as spatial objects, video, image, voice, text and biological data types [1, 2].
Content-based video retrieval (CBVR) is a technique used for retrieving similar video from a video database, CBVR systems appear like a natural extension of Content-based Image Retrieval (CBIR) systems. The video takes into consideration four different levels which are frame, shot, scene, and story level [3].
In frame level, each frame is treated separately as static image, set of contiguous frames all acquired through a continuous camera recording make shot level, set of contiguous shots having a common semantic significance make scene level and the complete video object is story level. A typical structure of video is shown in Fig.1.

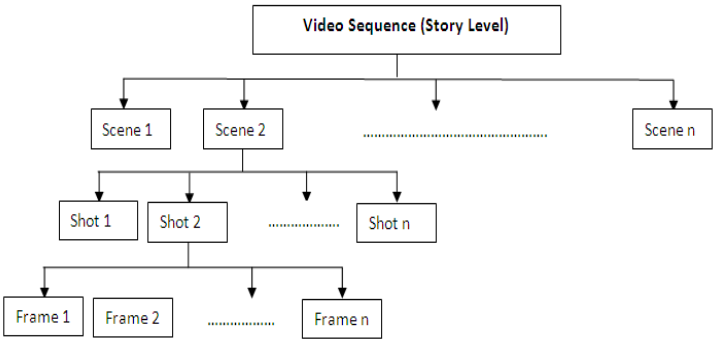


Fig.1 Common video structure

From this four level, the smallest basic meaningful unit can represent video's scenario is a shot. So to perform video search and retrieval process it will require the support of tools which can detect and isolate such meaningful shot segments in any video source [4]. The significant qualitative difference in content is easily apparent to human and according to this difference it will be easy to determine shot boundaries or video segments. If that difference can be expressed to computer by a suitable metric, then a segment boundary can be declared whenever that metric exceeds a given threshold. Hence, establishing such metrics and techniques for applying them is the first step for the automatic partitioning of video packages. Once video segmented, each segment determine key frames that will represent it, after that extract for each key frame color, shape and texture feature that represent its content and apply object annotation to reduce semantic gap, which refers to the discontinuity between the simplicity of features that can be currently computed automatically and the richness of semantics in user queries posed for video search and retrieval.
With this information, proposed system developed that is capable of accurately segmenting a wide range of video and apply video retrieval in satisfied manner. This paper is organized as follows. In Section 2, presents the proposed system methodology that contain, video segmentation, key frame selection, feature extraction ,apply object annotations to achieve high level semantic concept ,the matching process and automatic selection of the adaptive threshold. Section 3 discusses proposed frame work. Section 4 discusses experimental results. Finally conclusions and future work represented in section 5.

2. Proposed System Methodology

The proposed content based video retrieval system is divided into two phases offline and online phase. In offline phase crawler first navigates through a set of URL seeds searching for video files to construct video database. Once those videos are collected, they are preprocessed. The preprocessing phase starts by dividing video into segments based on a threshold value. However, appropriate threshold values selection is a key issue in applying segmentation and comparing changes between two frames feature values. Thresholds must be assigned that tolerate variations in individual frames while still ensuring a desired level of performance. Most of the existing methods use global pre-defined thresholds, or adaptive threshold. Heuristically chosen global thresholds is inappropriate because experiments have shown that the threshold for determining a segment boundary varies from one shot to another which must be based on the distribution of the frame-to-frame differences of shots. That's why using adaptive threshold [5] during this phase was more reasonable than global threshold. Afterwards key frame(s) is selected to represent each segment. The next step aims mainly to extract features that will represent video. Extracting video features, the proposed system mixes both low and high level features for video representation in order to bridge the gap between them. The feature extraction step begins by extracting content- related low level features, namely, color, texture and shape. Then, high level semantic features are extracted using object annotation. Offline phase results in a database of videos annotated and represented by their content – related features.

The next phase is online phase which includes submission of user query image. This image is preprocessed in the same way by extracting its low and high level features. The extracted features are then compared to features of stored videos in database. The matched videos are then retrieved and ranked according to the nearest to user query image.

3. Proposed system components

Fig.2 presents the main components of the proposed framework. The proposed content based video retrieval system works in two phases online and offline phases. The main components of the proposed system are crawler , video segmentation and frame selection module , feature extraction module that includes both low and high level features and both matching and retrieval module that retrieves, ranks and presents them to user.

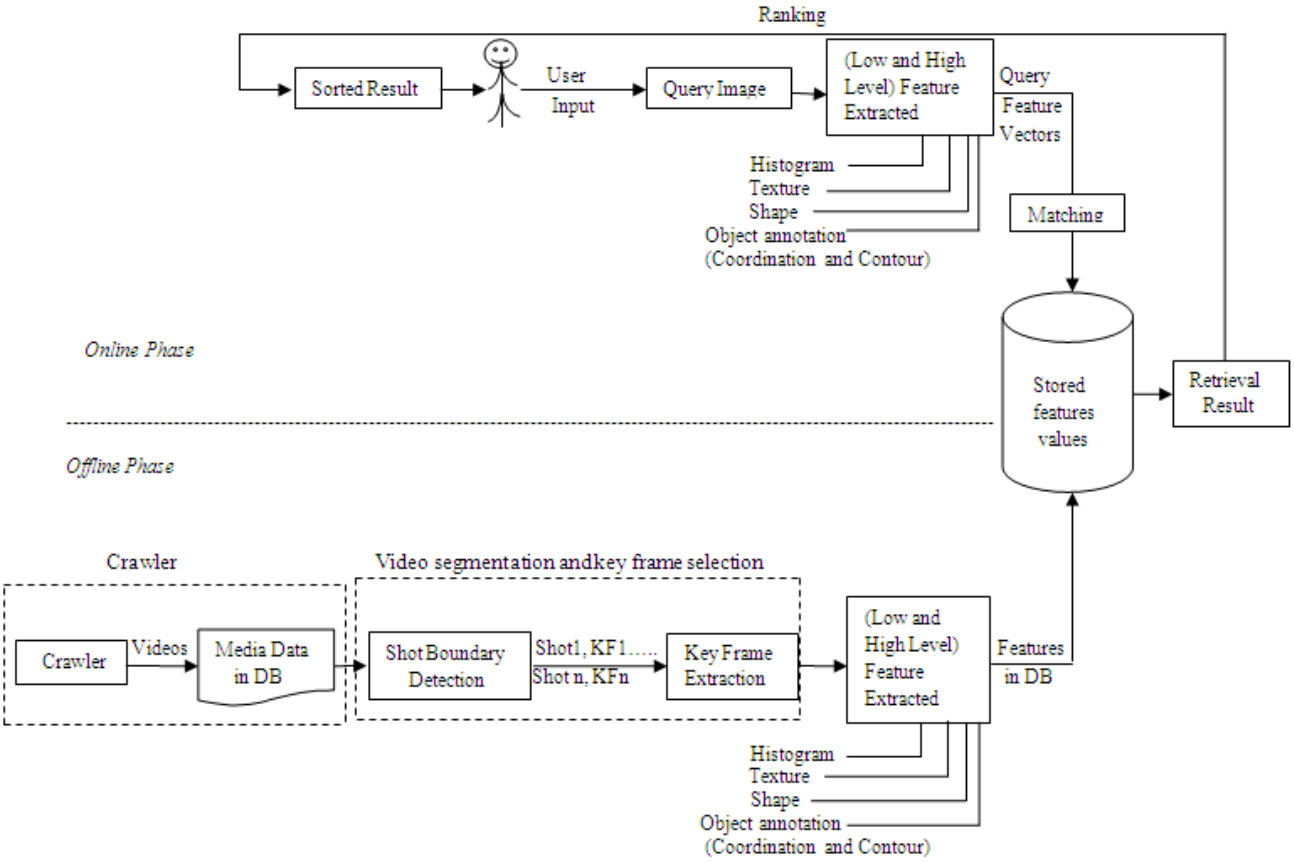


Fig.2 Proposed system Framework.

3.1 Crawler

A crawler is an automated program that methodically scans or crawls through Internet pages for searching and downloading purposes. Alternative names for a crawler include web spider, web robot, and web crawler. There are many purposes for which crawlers are used but the prime use is to download from Internet pages. A crawler needs a web address as a starting point in order to download videos from the website, and then these videos is stored in data base as media data.

3.2 Video segmentation and key frame selection

Video segmentation, or shot change detection, involves identifying the frame(s) where a transition takes place from one shot to another. This transition occurs when the absolute difference of mean blocks between two consecutive frames exceed a threshold value. In cases where this change occurs between two frames, it is called a cut or a break. Identifying breaks or cuts subdivides the entire video into shots for which key frames need to be identified. If large camera motion is present in a single shot, then two frames that are spaced well apart within this shot may be quite dissimilar. In such cases, more than one key frame may be required. Choosing key frames of scenes allows us to capture most of the content variations, due at least to camera motion, while at the same time excluding other key frames which may be redundant. The ideal method of selecting key frames would be to compare each frame to every other frame in the scene and select the frame with the least difference from other frames in terms of a given similarity measure. Obviously, this requires extensive computation and is not practical for most applications. On the other hand, choosing the first frame seems to be the natural choice, as all the rest of the frames in the scene can be considered to be logical and continuous extensions of the first frame, but it may not be the best match for all the frames in the scene [6].

3.3 Feature Extraction

Feature extraction is very crucial step in video retrieval system to describe the video frame with minimum number of descriptors. This includes the extraction of low level features, namely (color, shape and texture) and high level features, namely (object annotation).

3.3.1 Low Level Features

The basic visual features of images include color, shape and texture .Many research efforts, the use of only one low-level feature is still not powerful enough to represent frame content. Some features can achieve relatively good performance if combined to each other [7].

3.3.1.1 Color feature

The method that used to apply color feature extraction is a histogram. The principle behind this method is that two frames having an unchanging background and unchanging objects will show little difference in their respective histograms. Let Hi (j) denote the histogram value for the ith frame, where j is one of the G possible grey levels (The number of histogram bins can be chosen on the basis of the available grey-level resolution and the desired computation time.) Then the difference between the ith frame and its successor will be given by the following formula [4]:

$$HDi = \sum_{j=1}^{G} |Hi(j) - Hi+1(j)| \quad (1)$$

Where
G is the number of grey levels.
j is the grey value,
i is the frame number,
And H (j) is the value of the histogram for the grey level j.
If the overall difference HDi is larger than a given threshold T, a segment boundary is declared. This equation used for grey-level frames and to use it with color frames we first convert the color intensities into grey levels.

3.3.1.2 Texture feature

Texture, like color, is a powerful low-level descriptor for image search and retrieval applications .It is a fundamental feature which provides significant information about the spatial arrangement of color or intensities in an image or identifying objects or regions of interest in an image [8]. Texture could be defined in simple form as repetitive occurrence of the same pattern. Texture could be defined as something consisting of mutually related elements. Another definition of texture claims that, an image region has a constant texture if a set of its local properties in that region is constant, slowly changing or approximately periodic [9]. Since they are computed over gray levels, color images of the database are first converted to 256 gray levels. The method that used to extract the texture features is entropy, which is a statistical measure of randomness can be used to characterize the texture of the input image.
The value of entropy can be calculated as [10]:

$$ENT = \sum_{K=1}^{M} Pk \log 1/Pk \quad (2)$$

Where
ENT=Entropy of I/P,
M=Total no. of samples,
P=Probability of I/P occurrences.

3.3.1.3 Shape feature

In order to identify shape in a given image, edge detection techniques are used. The various gradient operators used for edge extraction are Sobel, Prewitt, Roberts and Canny.
Despite being well known to many as the optimal edge detector [11], canny detector's performance was tested against the former edge detecting algorithms. While visual results weren't enough to proved its efficiency, that's why peak signal to noise ratio (PSNR) measure was used to provide a statistical method for its performance. Fig.3 (a, b) both visual and performance measures assure the fact that canny is more suitable to choose in this phase.

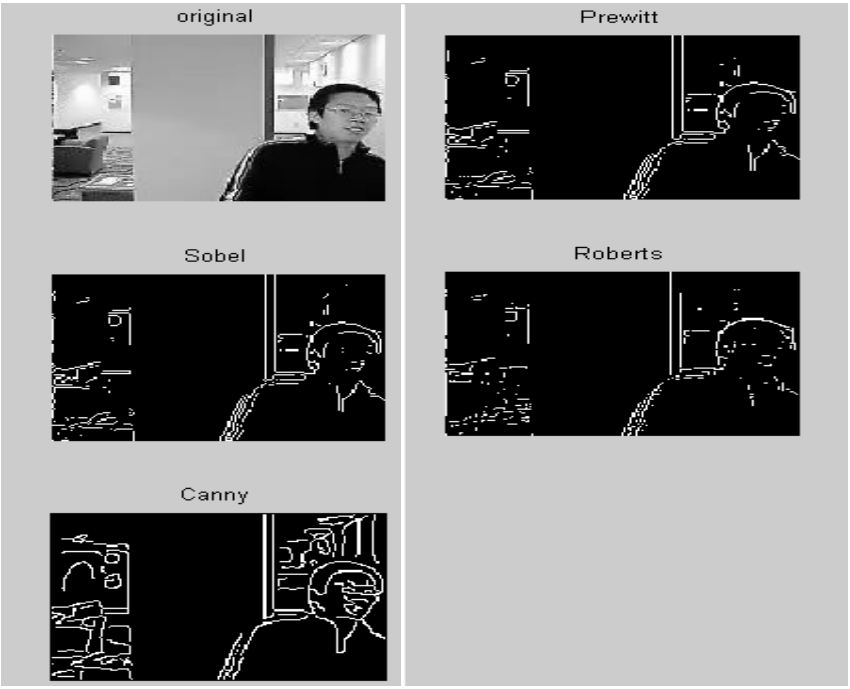


Fig (3.a) Visual Comparison of various edge detection Algorithms

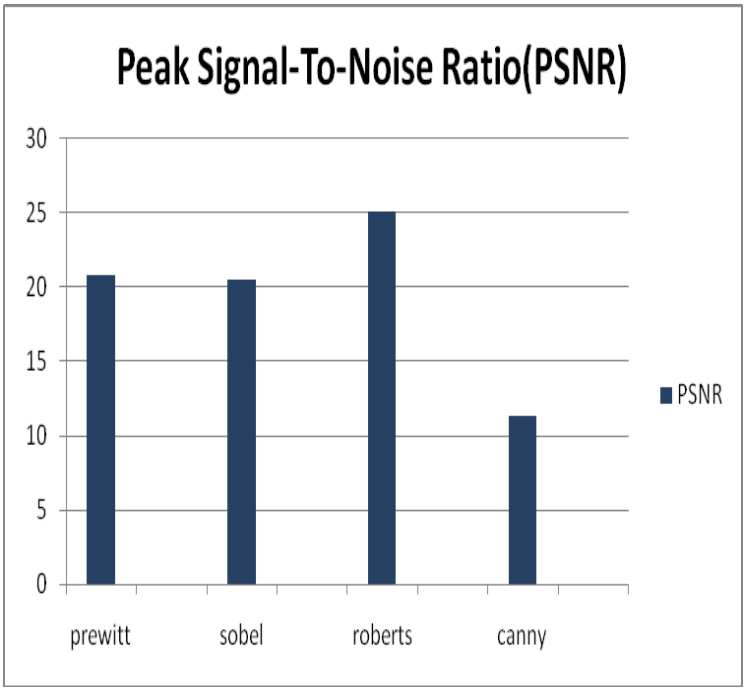


Fig (3.b) Performance Measures for Edge Detection Techniques

3.3.2 High-level Features

Features extraction using computer vision techniques are mostly based on low-level features. These features are not enough to retrieve satisfied result, because humans tend to use semantic objects to provide conceptual entities of visual content. To reduce the semantic gap between low and high level features, object annotation is often used. In this paper, graphical annotations are used to highlight regions or objects of interest. Object classes are learned from a set of labeled training images in LabelMe database [12].These dataset contains spatial annotations of thousands of object categories in hundreds of thousands of images.

4. Experimental Results

The proposed system has been validated using several kinds of video sequences. We report here some results obtained on a part of a video sequence utilized for retrieval, its performance was compared to the performance of a video retrieval system based only on color feature. Both systems were experimented using a database of 30 videos against 4 different queries. In order to

evaluate the quality of the proposed system, recall and precision rates of the retrieved results against manual human opinions are used. Recall is a measure of how well the proposed system performs in finding relevant items, while precision indicates how well it performs in not returning irrelevant items and F-measure is an average of the formers. Recall, Precision and F-measure are shown in formulas (3) and (4) as defined in [13].

$$\text{Precision} = \frac{|\{relevant\ videos\} - \{retrieved\ videos\}|}{|\{retrieved\ videos\}|} \quad (3)$$

$$\text{Recall} = \frac{|\{relevant\ videos\} - \{retrieved\ videos\}|}{|\{relevant\ videos\}|} \quad (4)$$

Table (1, 2) and Fig (4, 5) show the experimental results. The results showed that in the first case, where only color feature is used, both precision and recall were about 60% in average. Whereas, testing the multi-feature system resulted in 79% precision and 73% recall in average. These results proved that using multi-features increases precision and recall by about 13 and 19% with respect to the first system.

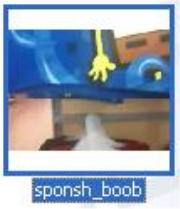
Query Image	Retrieved and Relevant Video	Precision	Recall
	 sponsh_boob	0.75	0.75
	 Tom_And_Jerry_Train	0.6000	0.50
	 abo_esma3el	0.5567	0.6667
	 PLEASE_NO	0.50	0.50

Table 1 Experimental results of video retrieval system based on Color Feature Only.

Query Image	Retrieved and Relevant Video	Precision	Recall
	 All_of_my_money_Sponsh	0.9091	1.0
	 Tom_and_Jerry_bump	0.6000	0.75
	 abo_esma3el	0.6667	0.6667
	 Tom_Cruise_interview_bout_Jerry_Maguire	1.0	0.50

Table 2 Experimental results of proposed video retrieval system based on both low and high level features

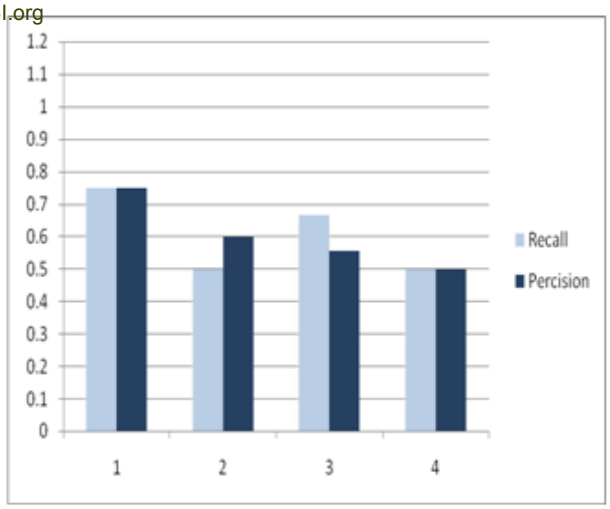


Fig.4 Proposed system evaluation results Using Color Feature Only

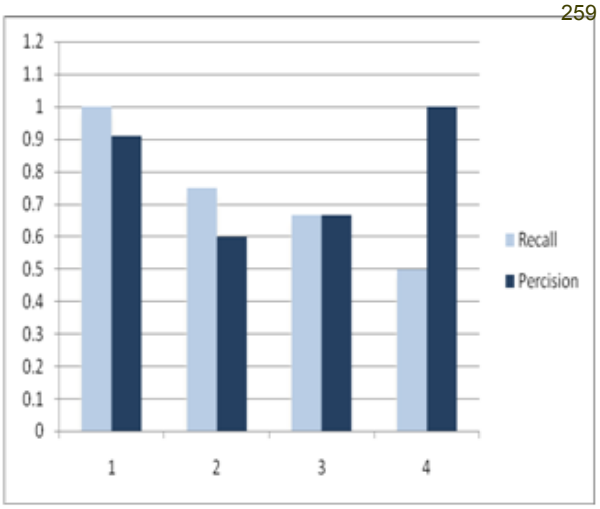


Fig.5 Proposed system evaluation results based on both low and high level features

5. Conclusion and future work

This paper presented the implementation of the proposed content based video retrieval system. This system tries to bridge the semantic gap between low and high level features using semantic object annotation. Every video in the database is segmented into several shots. For each shot, one or more key frames are selected, and then a features vector for each key frame is computed. The sequences of feature vectors are stored in the feature database. User's query image is also extracted its features. Then, the proposed system uses a dynamic programming approach to compute the similarity between the sequence of feature vectors of the query image and each sequence of feature vectors in the feature database. Finally videos are ranked according to their similarity and only videos with similarity higher than a predefined threshold are returned to user. Testing the proposed system against older systems resulted in a raise in precision and recall by about 19% and 13% respectively. Future work includes performing more experimental results using a large scale video set as well as the attempt to obtain user's feedback rates and use those rates as a ranking factor.

6. Bibliography

[1] Chavez, E., Navarro, G., Baeza-Yatesand R., Marroquin, J. L., "Searching in metric spaces", ACM Computing Surveys, (2001), 33(3): 273-321.

[2]Xu, W., Briggs, W. J., Padolina, J., Liu, W., Linder, C. R. & Miranker, D.P., "Using MoBIoS' Scalable Genome Joins to Find Conserved Primer Pair Candidates Between Two Genomes", ISMB04, Glasgow, Scottish (2004).

[3] Shweta Ghodeswar1, B.B. Meshram, Content Based Video Retrieval using Entropy , Edge Detection , Black and White, 2nd International Conference on Computer Engineering and Technology (2010),Pages V6-272 - V6-276

[4] H.Zhang, Kankanhalli & Smoliar, Automatically partitioning of full-motion video. Multimedia Systems, 1993,1(1), 321-339.

[5] Arasanathan Anjulan and Nishan Canagarajah Video Scene Retrieval Based on Local Region Features ICIP , IEEE 2006,PP 3177 - 3180.

[6] P. Geetha and V. Narayanan, A Survey on Content based video retrieval, journal of Computer Science2008,Volume 4, Issue 6,PP 474-486

[7] Hui Yu,Mingjing Li,Hong-Jiang Zhang,Jufu Feng, Color Texture Moments For Content – Based Image Retrieval, Image Processing International Conference, vol.3,2002,PP 929 - 932

[8] Haralick, Robert M, Textural Features for Image Classification, IEEE Systems, Man, and Cybernetics Society, Volume: 3 Issue:6,2007,pp610 – 621

[9] M.C. Padma, P.A.Vijaya, Entropy Based Texture Features Useful for Automatic Script Identification, M.C. Padma et al. / (IJCSE) International Journal on Computer Science and Engineering,Vol. 02, No. 02, 2010, 115-120

[10] Dr. H.B.Kekre, Sudeep D. Thepade, Image Retrieval using Texture Features extracted from GLCM, LBG and KPE, Vol. 2, No. 5, Oct, 2010, pp.1793-8201

[11] H.B.Kekre,S.Thepade, Image Retrieval with Shape Features Extracted using Gradient Operators and Slope Magnitude Technique with BTC, Volume 6– No.8, Sep 2010

[12] K. P. M. C. Russell, A. Torralba and W. T. Freeman. Labelme: a database and web-based tool for image annotation. International Journal of Computer Vision, 77:157{173, May 2008.
[13] F. McSherry and M. Najork. "Computing Information Retrieval Performance Measures Efficiently in the Presence of Tied Scores", ECIR 2008, LNCS 4956, pp. 414–421, 2008. C_Springer-Verlag Berlin Heidelberg 2008.
[14] I.K. Sethi, I.L. Coman, Mining association rules between low-level image features and high-level concepts, Proceedings of the SPIE Data Mining and Knowledge Discovery, vol. III, 2001, pp. 279–290.
[15]A. Mojsilovic, B. Rogowitz, Capturing image semantics with low-level descriptors, Proceedings of the ICIP, September 2001, pp. 18–21.
[16]X.S. Zhou, T.S. Huang, CBIR: from low-level features to highlevel semantics, Proceedings of the SPIE, Image and Video Communication and Processing, San Jose, CA, vol. 3974, January 2000, pp. 426–431.
[17] Shradha Gupta, Neetesh Gupta, Shiv Kumar, Evaluation of Object Based Video Retrieval Using SIFT, ISSN: 2231 2307, Volume-1, Issue-2, May 2011
[18] Arasanathan Anjulan, Nishan Canagarajah ―Object based video retrieval with local region tracking, Signal Processing: Image Communication 22 (2007) 607–621.
[19] P. Geetha, Vasumathi Narayanan, A Survey of Content-Based Video Retrieval, Journal of Computer Science 4 (6): 474-486, 2008 ISSN 1549-3636 © 2008 Science Publications.