Content based Video Retrieval using Latent Semantic Indexing and Color, Motion and Edge Features

Kalpana S Thakare
Assistant Professor, Dept. of IT,
Sinhgad College of Engineering
Pune

Archana M Rajurkar, PhD.
Head, Dept. of CSE,
MGM College of Engineering,
Nanded

R R Manthalkar, PhD.
Head, Dept. of EnTC SGGS
Institute of Engg. & Tech.
Nanded

ABSTRACT
Optimal efficiency of the retrieval techniques depends on the search methodologies that are used in the video processing system. The use of inappropriate search methodologies may make the processing system ineffective. Hence, an effective video segmentation and retrieval system is an essential pre-requisite for searching a relevant video from a huge collection of videos. In this paper we propose a video retrieval system based on the integration of various visual cues. In contrast to key-frame based representation of shot, our approach analyzes all frames within a shot to construct a compact representation of video shot. In feature extraction step we extract quantized color, motion and edge density features. A similarity measure is defined using LSI (Latent semantic indexing) to locate the occurrence of similar video clips in the database. Our approach is able to fully exploit the spatio-temporal contents of the video. Experimental results indicate that the proposed algorithm is effective and outperforms some existing technique. The detailed result analysis and graphs supports the effectiveness and correctness of the system.

General Terms
Image/Video Processing, Video Retrieval

Keywords
Video retrieval, video database, video matching, similarity measure.

1. INTRODUCTION
Recent developments in computer technology, especially in the area of storage technology, have led to a considerable growth in the quantity and quality of multimedia databases. This has provided a good platform for the demand for information retrieval from these large databases. In general, multimedia contents comprise audio and video signals without indexes. So, conventional text-based information retrieval methods cannot be directly employed to multimedia contents. Due to the high cost, manual indexing to these databases is generally impractical [2]. The traditional text based search experiences the subsequent drawbacks: (1) Manual annotations are time consuming and costly to implement. With the increase in the number of media in a database, the complexities in determining the required information also increases. To manually annotation of all attributes of the media content is a difficult task [4]. (2) Manual annotations fall short in handling the difference of subjective perception. The phrase "a picture is worth a thousand words" describes that the textual description is not necessary for representing subjective perception. Acquiring the entire attributes of the content of any media is unachievable [2]. For this reason, we require a good search technique for Content-Based Video Retrieval System (CBVR). In other words, content-based is defined as the search which will examine the original image contents. Here, content relates to colors, shapes, textures, or any other information that can be obtained from the image directly [5]. Recently, CBVR system has been widely studied. In CBVR, vital information is automatically taken out by employing signal processing and pattern recognition techniques for audio and video signals [2]. Digital video needs to efficiently store the index, store, and retrieve the visual information from multimedia database. Video has both spatial and temporal dimensions and video index should capture the spatio-temporal contents of the scene. In order to achieve this, a framework mainly works into three basic steps. Shot segmentation, Feature extraction and finally similarity match for effective retrieval of the query clip. This approach has established a general framework of image retrieval from a new perspective. The query example may be an image, a shot or a clip. A shot is a sequence of frames that was continuously captured by the same camera, while a clip is a series of shots describing a particular event. Our query statement is formulated as: given a sample clip, find all occurrences of similar (or relevant) video clips in the database. Current techniques for content-based video retrieval can be broadly classified into two categories.

1) Frame sequence matching [6] Mohan 1998, proposed a scheme that matches videos based on similarity of temporal activity, it finds similar "actions". Furthermore, it provides precise temporal localization of the actions in the matched videos. Video sequences are represented as a sequence of feature vectors called fingerprints. The fingerprint of the query video is matched against the fingerprints of videos in a database using sequential matching.

[7]Tan et al. 1999, here author achieves the compact shot representation by integrating the color and spatial features of individual frame. In the video matching step, a shot similarity measure is defined to locate the occurrence of similar video clips in the database. [8]Naphade et al. 2000, proposed original two-phase scheme for video similarity detection. For each video sequence, they extract two kinds of signatures with different granularities: coarse and near Coarse. In the second phase, the query video example is compared with the results of the first phase according to the similarity measure of the near signature. They achieve better quality results than the conventional approaches. Many authors [9] Hoad & Zobel 2003,[10] Ren & Singh 2004,[11] Kim & Vasudev 2005,[12] Toguro et al. 2005 discussed and designed their models using the concept of frames sequence matching.

2) Key-frame based shot matching: [14] Jain et al. 1999, implemented the algorithm using key-frames of abrupt

transitions. They extracted image features (color, texture and motion) around the key frames. For each key frame in the query, a similar value is obtained with respect to the key frames in the database video. Consecutive key frames in the database video that are highly similar to the query key frames are then used to generate the set of retrieved video clips.

[15]Lienhart et al. 2000 proposed an efficient algorithm for video sequence matching using the modified Hausdorff distance and the directed divergence of histograms between successive frames. to effectively match the video sequences with a low computational load, author uses the key frames extracted by the cumulative directed divergence and compare the set of key frames using the modified Hausdorff distance. The same approach of key frame based shot matching is used by [13]Liu et al. 1999, [16]Kim & Park 2002, [17]Diakopouos & Volmer 2003,[18] Peng et al. 2003, [19]Luo et al. 2007.

The first approach of frame sequence matching is derived from the sequential correlation matching that is widely used in the signal processing domain. These methods usually focus on frame-by-frame comparison between two clips in order to find sequences of frames that are consistently similar. The common drawback of these techniques is the heavy computational cost of the exhaustive search. Although there exist some techniques ([20] Kashino et al. 2003) to improve the linear scanning speed, their time complexity still remains at least linear to the size of database. Additionally, these approaches are susceptible to alignment problem when comparing clips of different encoding rates. In second category, each video shot is represented by a key-frame compactly. To reduce computational cost, video sequence matching is achieved by comparing the visual features of key-frames. The problem with these approaches lies in that they all leave out the temporal variations and correlation between key-frames within an individual shot. Also, it is not clear as to which image should be used as the key-frame for a shot. To strike a good balance between searching accuracy and computational cost, in this paper, we propose an integrated approach for shot matching. In contrast to previous approaches, our approach analyzes all frames within a shot to extract more visual features for shot representation. Because there does not exists a single visual feature for the best representation of video content, we integrate several visual features to capture the spatio-temporal information more accurately.

The next section of this paper describes the shot segmentation method. Visual features used in our work are extracted and are described in section 3. Then, the proposed similarity measure and video matching algorithm are described in Section 4. The performance evaluation of our approach is reported in Section 5. Finally, some concluding remark is given in Section 6.

2. SHOT SEGMENTATION

In the process of shot segmentation, the entire video clips are separated into 'chunks' or video shots. Consider the database video clips as $v_i;\ \ 0 \le i \le N_v - 1$, in which each clip is constituted of $f_{ij};\ \ 0 \le j \le N_{f_i} - 1$ frames of size $M \times N$. In other words, the shot segmentation can also be defined as the grouping of consecutive frames based on the captured shots. In the proposed retrieval system, the shot segmentation is performed by applying biorthogonal wavelet transformation to every frame of a video clip and then by calculating the L2-norm distance between every frame. Firstly, all the frames of every i^{th} video clip is transformed to biorthogonal wavelet transformation domain as

$$F_{ij}(x,y) = X(x,y)\, f_{ij}(x,y) X^{-1}(x,y);$$
$$0 \le x \le M-1,\ 0 \le y \le N-1 \qquad (1)$$

where, F_{ij} is the frame f_{ij} in the wavelet domain and X is the transformation matrix of the biorthogonal wavelets. Then, the frames f_{ij} and f_{ij-1} are chosen as same shot when it satisfies the condition $L2_{ij} \le S_T$. If $L2_{ij} > S_T$, then the frames f_{ij} and f_{ij-1} belongs to different shots, where, S_T is the threshold to separate shots and $L2_{ij}$ is the L2 norm distance between f_{ij} and f_{ij-1} that can be determined as

$$L2_{ij} = \sqrt{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} \left(|F_{ij}(x,y) - F_{ij-1}(x,y)| \right)^2} \qquad (2)$$

By checking all the consecutive frames, they are separated based on its belonging shots. Hence, N_s number of shots is obtained for every i^{th} video clip and $f_{ikl}^{(j)}$ be the frames that belongs to the k^{th} shot of i^{th} video clip. Once the frames of all the database images are segmented based on shots, they are subjected to the process of feature extraction.

3. FEATURE EXTRACTION

In the process of feature extraction, here, some dominant features such as Motion feature, Quantized color feature and Edge density are determined. The feature extraction process is described as follows.

3.1 MOTION FEATURE EXTRACTION

Motion features are any components in the video clip that exhibits motion i.e. the components that shows movement in the consecutive frames. They are extracted by dividing the frames of a particular shot into several blocks and then by identifying the blocks that exhibits motion in the consecutive frames. Hence, each frame of the k^{th} shot is sub-divided into n_b blocks (each of size $m \times n$) and then the SED is determined for the blocks of two consecutive frames as follows

$$SED_{ikl}^{(j)}(c) = \sum_{x=0}^{m-1}\sum_{y=0}^{n-1} \left(b_c^{(j)}(x,y)_{ikl} - b_c^{(j)}(x,y)_{ik(l+1)} \right)^2 \qquad (3)$$

Based on the SED, the blocks that has moving object are identified and their index are appended as follows

$$[b_M]_{ikl}^{(j)} << b_{c_{ik(l+1)}}^{(j)} \quad ; \ if \quad SED_{ikl}^{(j)}(c) < M_T \tag{4}$$

Thus obtained blocks $[b_M]_{ikl}^{(j)}$ for every l^{th} frame belongs to k^{th} shot of a video clip is constituted by a moving object. Hence, the blocks are stored as the motion feature of the i^{th} video clip and they are stored in the feature database as a vector.

3.2 COLOR FEATURE EXTRACTION

Color quantization or color image quantization is a process that lessens the number of individual colors employed in an image or frame of a video clip, generally with the intent that the new image should be as visually identical to the original image. Here, with the aid of the color quantization process, the color features are extracted from the shot segmented video clips. To accomplish this, firstly, all the frames of every shot of the video clip is converted from RGB color space to $L*a*b$ color space as follows

$$L* = \begin{cases} 116\ \sqrt[1/3]{Y/Y_0} - 16 & ; \ if \quad Y/Y_0 > 0.008856 \\ 903.3\ \dfrac{Y}{Y_0} & ; \ otherwise \end{cases} \tag{5}$$

$$a* = 500(\psi(X/X_0) - \psi(Y - Y_0)) \tag{6}$$

$$b* = 200(\psi(Y/Y_0) - \psi(Z - Z_0)) \tag{7}$$

In Eq. (5), Eq. (6) and Eq. (7), X_0, Y_0 and Z_0 are the tristimulus values of the reference white. The $\psi(t)$ and the XYZ model of the RGB color space are obtained as model of the RGB color space are obtained as

$$\psi(t) = \begin{cases} \sqrt[3]{t} & ; \ if \ t > 0.008856 \\ \dfrac{7.787\ t + 16}{116} & ; \ otherwise \end{cases} \tag{8}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = T \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

where, T is the Transformation Matrix and generally, it can be given as

$$T = \begin{bmatrix} 0.412435 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \tag{9}$$

The color space converted frames are then divided into blocks as done earlier. Then, each block of every frame is subjected to DCT transformation as

$$B_c^{(j)}(u,v)_{ikl} = \alpha_u \alpha_v \sqrt{\frac{2}{m}} \sqrt{\frac{2}{n}} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} b_c^{(j)}(x,y)_{ikl} \cos\left(\frac{\pi u}{2m}(2x+1)\right) \tag{10}$$

where,

$$\alpha_u = \begin{cases} \sqrt{1/2} & ; \ if \quad u = 0 \\ 1 & ; \ if \quad 1 \le u \le m-1 \end{cases} \tag{11}$$

$$\alpha_v = \begin{cases} \sqrt{1/2} & ; \ if \quad v = 0 \\ 1 & ; \ if \quad 1 \le v \le n-1 \end{cases} \tag{12}$$

The obtained blocks $B_c^{(j)}(u,v)_{ikl}$ which are in DCT domain are scanned in zigzag fashion. While, zigzag scanning of a block, the first N_c elements of the block are extracted as $[b_c^{zigzag}]_{ikl}^{(j)}$ and it is stored as the quantized color feature of the block.

3.3 EDGE DENSITY EXTRACTION

The edge density feature is an attribute of a video clip that can indicate the clip frames by means of magnitude of the edge of any object present in the clip. To extract the feature, firstly, the shot segmented video clip is resampled so that the frames of the shot segmented video clip accomplish the size of $M_r \times N_r$. The resampled frames of the shot segmented video clip are subjected to gray scaling operation and so that every frame of the shot segmented video clips that are in RGB color space is converted to gray scale. Then, two pixel distances are determined in Eq. (13) and Eq. (17) as follows

$$d_{1_{ikl}}^{(j)} = \left| f_{gray}^{(j)}(x,y) - f_{gray}^{(j)}(x-1,y-1) \right|; \quad 1 \le x \le M_r - 1,\ 1 \le y \le N_r - 1 \tag{13}$$

Once the distance is calculated, an edge preserving operation is performed based on the obtained distance and so three classes of edges are obtained as follows

$$E_{1_{ikl}}^{(j)}(x,y) = \begin{cases} G_{\max} & ; \ if \quad d_{1_{ikl}}^{(j)}(x,y) > G_{LT} \\ G_{\min} & ; \ otherwise \end{cases} \tag{14}$$

$$E_{2_{ikl}}^{(j)}(x,y) = \begin{cases} G_{\min} & ; \ if \quad d_{1_{ikl}}^{(j)}(x,y) > G_{LT} \\ G_{\max} & ; \ otherwise \end{cases} \tag{15}$$

$$E_{3_{ikl}}^{(j)}(x,y) = \begin{cases} G_{\max} & ; if \ d_{1_{ikl}}^{(j)}(x,y) > G_{HT} \\ G_{\min} & ; otherwise \end{cases} \tag{16}$$

$$d_{2_{ikl}}(x,y) = \left| f_{gray}^{(j)}(x,y+1) - f_{gray}(x,y) \right|;$$
$$0 \le x \le M_r - 2 \text{ and } 0 \le y \le N_r - 2 \quad (17)$$

$$E_{1_{ikl}}^{(j)}(x,y) = \begin{cases} G_{\max} \ ; \ if \ \ E_{1_{ikl}}^{(j)}(x,y) = 0 \ \ and \ \ d_{2_{ikl}}^{(j)}(x,y) > G_{LT} \\ G_{\min} \ ; \ otherwise \end{cases} \quad (18)$$

$$E_{2_{ikl}}^{(j)}(x,y) = \begin{cases} G_{\min} \ ; \ if \ \ E_{1_{ikl}}^{(j)}(x,y) = 0 \ \ and \ \ d_{2_{ikl}}^{(j)}(x,y) > G_{LT} \\ G_{\max} \ ; \ otherwise \end{cases} \quad (19)$$

$$E_{3_{ikl}}^{(j)}(x,y) = \begin{cases} G_{\max} \ ; \ if \ \ E_{3_{ikl}}^{(j)}(x,y) = 0 \ \ and \ \ d_{2_{ikl}}^{(j)}(x,y) > G_{HT} \\ G_{\min} \ ; \ otherwise \end{cases} \quad (20)$$

The edges $E_{1_{ikl}}^{(j)}$, $E_{2_{ikl}}^{(j)}$ and $E_{3_{ikl}}^{(j)}$ obtained in Eq. (14), Eq. (15) and Eq. (16) are based on the Eq. (13) and based on Eq. (14), Eq. (16) and Eq. (17), the edges $E_{1_{ikl}}^{(j)}$, $E_{2_{ikl}}^{(j)}$ and $E_{3_{ikl}}^{(j)}$ are obtained. The final edge obtained in the Eq. (20,) $E_{3_{ikl}}^{(j)}$ is subjected to determine the edge density. This can be accomplished by determining the edge density matrix as follows

$$\rho_{E_{ikl}}^{(j)}(p,q) = C_{ikl}^{\max}\left(\frac{n_b p}{2} + q\right) + 1;$$
$$0 \le p \le n_{b/2} - 1, \ 0 \le q \le n_{b/2} - 1 \quad (21)$$

where, $C_{ikl}^{\max}\left(\frac{n_b p}{2} + q\right)$ is the number of $G_{\max}$ values present in the c^{th} block ($c = \frac{n_b p}{2} + q$)of the edge $E_{3_{ikl}}^{(j)}$. It is to be noted that the $E_{3_{ikl}}^{(j)}$ is the edge obtained for the l^{th} frame of the k^{th} shot that belongs to the i^{th} video clip. Thus obtained $\rho_{E_{ikl}}^{(j)}$ is stored as the edge density feature vector of the corresponding video clip.

4. RETRIEVAL OF VIDEO CLIPS BASED ON QUERY CLIPS

In the retrieval, the database video clips that are similar to the query clip are retrieved by means of measuring the similarity in between the query clip and the database video clips. When a query clip is given to the proposed retrieval system, all the aforesaid features are extracted as performed for the database video clips. Then, with the aid of LSI, similarity is measured between every database video clip and the query clip.

Before we perform the LSI based similarity measure, the transpose of each feature vector extracted for every video clip is determined so as to obtain the feature vector as column vector. The obtained column vectors for the motion feature of all the database video clips are concatenated and then by appending zeros in the necessary locations, a feature matrix is generated. Then, the column vectors of the next feature, color, are appended just below the particular location of the feature matrix. In other words, the column vector for color feature of the 0th video clip is concatenated below the element of the 0th column of the feature matrix. Similarly, feature vectors for all the video clips are performed. The same process is repeated for the final feature vector, edge density. Hence, a feature matrix A of size $\hat{N} \times N_v$ is obtained. When a query clip is given, all the aforesaid features are extracted. Then, the feature vector is converted to column vector and then all the feature vectors are concatenated below (as stated above). Hence, a column feature vector $N_q \times 1$ is obtained for the query clip.

In the first process of LSI based similarity measure, the A is subjected to SVD decomposition. Using the SVD theorem, the matrix A is decomposed as

$$A_{\hat{N} \times N_v} = U_{\hat{N} \times \hat{N}} \ \ S_{\hat{N} \times N_v} \ \ V^T_{N_v \times N_v} \quad (22)$$

where, $S \in R_{m \times n}$ is a diagonal matrix with nonnegative diagonal elements known as the singular values, $U \in R_{m \times n}$ and $V \in R_{m \times n}$ are orthogonal matrices. The columns of matrices U and V are labeled as the left singular vectors and the right singular vectors respectively and they can be given as

$$U^T U = I_{\hat{N} \times \hat{N}} \quad (23)$$

$$V^T V = I_{N_v \times N_v} \quad (24)$$

A query vector co-ordinate is determined for the query vector as follows

$$q_{co} = q^T \times U \times S^{-1} \quad (25)$$

where, q^T is the transpose of the query vector. Based on the query vector, similarity is determined as

$$d_{3_x} = \frac{1}{N_v} \sum_{y=0}^{N_v - 1} \frac{q_{co}^T(x).\left(V^T(x,y)\right)}{\left(q_{co}^T(x)\right)'.\left(V^T(x,y)\right)'} \ ;$$
$$0 \le x \le N_v - 1 \quad (26)$$

where, d_{3_x} is the similarity coefficient determined for the x^{th} database video clip. Then, N_r numbers of similar

database video clips are retrieved based on d_{3_x} . This can be accomplished by retrieving the video clips that has maximum d_{3_x} . Hence, the database video clips that are similar to the query clip are retrieved based on the LSI.

5. RESULTS AND DISSCUSSIONS

The proposed retrieval system is implemented in the MATLAB platform (version 7.10) and tested using the database video clips of MPEG-2 format. To evaluate the performance of the proposed approach, we set up a database that consists of 50 videos. The genres of videos include home video, news, sports, movies and documentaries. The testing with different genres of videos would ensure that the overall performance of the algorithm is not biased toward a specific video category.

The frame results obtained in the intermediate process of the proposed CBVR system is depicted in the following figures

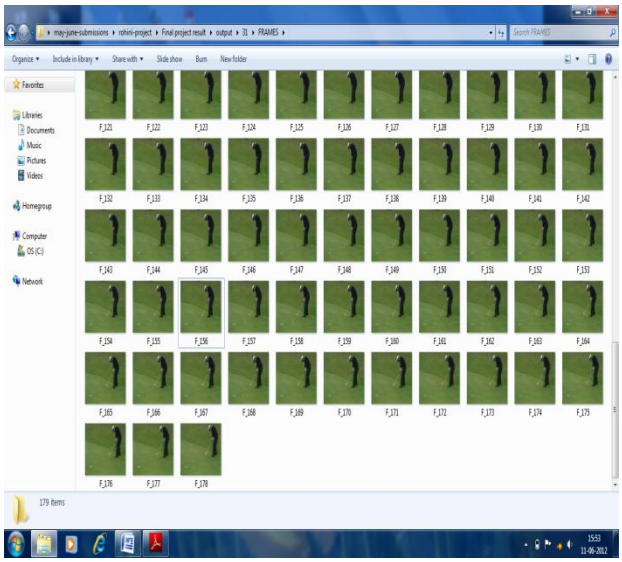
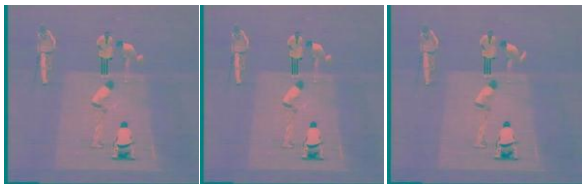
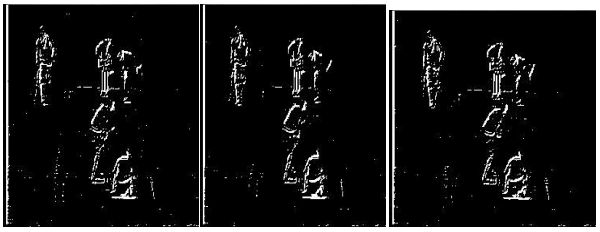


Fig 1: Frames of query video 31

(a)



(b)



(c)

Fig 2: Sample output from color space conversion: (a) frames in RGB color space (b) frames in $L^*a^*b^*$ color space(c)frames in edge detection

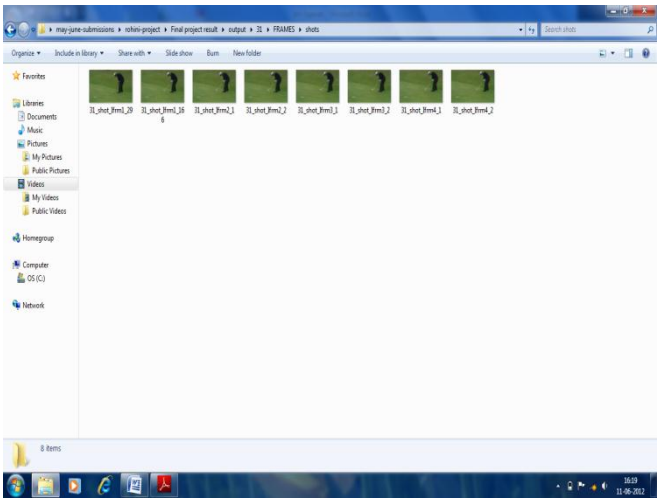


Fig 3: Shots detected for query video 31

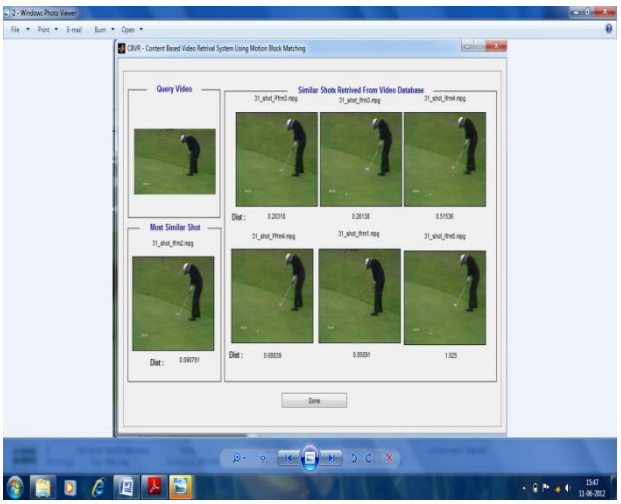


Fig 4: Retrieval results for query video 31

The performance of video retrieval is usually measured by the following two metrics:

$$recall = \frac{DC}{DB} \quad (27)$$

$$precision = \frac{DC}{DT} \quad (28)$$

where *DC* is the number of similar clips which are detected correctly, *DB* is the number of similar clips in the database and *DT* is the total number of detected clips. The ground truth of database, i.e., the decision whether a video clip is similar or not, is determined by human perception.

We compare our implemented approach, with the well known video retrieval algorithm proposed by Jain et al.(Jain et al. 1999). They implemented the algorithm using key-frames of abrupt transitions i.e. Hard cuts, they extracted image features (color, texture and motion) around the key frames. For each key frame in the query, a similar value is obtained with respect to the key frames in the database video. Consecutive key frames in the database video that are highly similar to the query key frames are then used to generate the set of retrieved video clips.

Table 1: Performance comparison on some queries

Query No.	Our approach		Jain's approach	
	Recall	**Precision**	**Recall**	**Precision**
17	58	72	45	63
22	71	77	65	75
23	58	100	53	75
25	63	85	60	63
26	77	57	55	46

The performance of Jain's algorithm may be limited by the following factors:

- Instead of using the all the frames that make a shot, only the image features of key-frame are used to represent the whole shot content. Only few key frames cannot possibly contain the complete visual cues of the video shot.
- The color description is based on traditional histogram which does not capture spatial layout information of each color.
- The video similarity is measured by the Euclidean distance between feature histograms. However, two different bins may represent perceptually similar features but are not compared in this measure. It has been shown that video similarity measured by LSI and SVD is more effective than histogram Euclidean distance for image retrieval.

.

6. CONCLUSION AND FUTURE SCOPE

We have presented a new video shot representation and a video similarity measure to achieve video retrieval task. Unlike key-frame based representation of shot, the proposed approach analyzes all frames within a shot to extract more visual features for shot representation. Our approach integrates color, motion and edge features to fully exploit the spatio-temporal information contained in video. Thus, the proposed system is able to resemble human similarity perception to some extent. Experimental results indicate that the proposed approach is effective and feasible in retrieving and ranking similar video clips. Finally, our future work should incorporate other video features, such as audio and text, for assessing video similarity. The work can also expanded for fade, dissolve and wipe gradual transitions.

7. ACKNOWLEDGMENTS

Our thanks to the experts, who have contributed towards development of the template.

8. REFERENCES

[1] Boycott, B.(2001), Color Vision, Cambridge University Press, Cambridge, U.K.

[2] Petkovic, Milan, Jonker, Willem,(2003)"Content-based video retrieval", Kluwer Academic Publishers, Boston, Monograph, 2003, 168 p., Hardcover ISBN: 978-1-4020-7617-6

[3] Arnold, W., M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain,(2000) "Content-Based Image Retrieval at the End of the Early Years", In proceddings of IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.22, pp.1349 - 1380, 2000.

[4] Chia-Hung Wei, Chang-Tsun Li,(2004) "Content–based multimedia retrieval - introduction, applications, design of content-based retrieval systems, feature extraction and representation" , 2004

[5] John Eakins, Margaret Graham,(1999) University of Northumbria at Newcastle, "Content-based Image Retrieval" (JISC Technology Applications Program Report 39 -1999deo Browsing Strategies.

[6] Mohan, R.(1998), Video sequence matching, *in* 'Proceedings of International Conference on Acoustic, Speech and Signal Processing', pp. 3697–3700.

[7] Tan Y. Kulkarni S., & Ramadge, P. (1999), A framework for measuring video similarity and its application to video query by example, *in* 'International Conference on Image Processing', pp. 106–110.

[8] Naphade, M., Yeung, M. & Yeo, B. (2000), A novel scheme for fast and efficient video sequence matching using compact signature, *in* 'SPIE Conference on Storage and Retrieval for Media Database', pp. 564–572.

[9] Hoad, T. & Zobel, J. (2003), Fast video matching with signature alignment, *in* 'ACM SIGMM International Workshop on Multimedia Information Retrieval', Berkeley, CA, pp. 262–269.

[10] Ren, W. & Singh, S. (2004), Video sequence matching with spatio-temporal constraints, *in* 'International Conference on Pattern Recognition', pp. 834–837.

[11] Kim, C. & Vasudev, B. (2005), 'Spatiotemporal sequence matching for efficient video copy detection', IEEE Transactions on Circuits and Systems for Video Technology **15**(1), 127–132.

[12] Toguro, M., Suzuki, K., Hartono, P. & Hashimoto, S. (2005), Video stream retrieval based on temporal feature of frame difference, *in* 'Proceedings of International Conference on Acoustic, Speech and Signal Processing', Volume 2, pp. 445–448.

[13] Liu, X., Zhung, Y. & Pan, Y. (1999), A new approach to retrieve video by example video clip, *in* 'ACM International Conference on Multimedia', pp. 41–44.

[14] Jain, A., Vailaya, A. & Wei, X. (1999), 'Query by video clip', *Multimedia Systems* **7**, 369–384.

[15] Lienhart, R., Effelsberg, W. & Jain, R. (2000), 'VisualGREP: A systematic method to compare and retrieve video sequences', *Multimedia Tools and Applications* **10**(1), 47–72.

[16] Kim, S. & Park, R. (2002), 'An efficient algorithm for video sequence matching using the modified Hausdorff distance and the directed divergence', *IEEE Transactions on Circuits and Systems for Video Technology* **12**(7), 592–596.

[17] Diakopouos, N. & Volmer, S. (2003), 'Temporally tolerant video matching', *in* 'ACM SIGIR Workshop on Multimedia Information Retrieval', Toronto, Canada.

[18] Peng, Y. & Ngo, C. (2004), Clip-based similarity measure for hierarchical video retrieval, *in* 'ACM SIGMM International Workshop on Multimedia Information Retrieval', pp. 53–60.

[19] Luo, H., Fan, J., Satoh, S. & Ribarsky, W. (2007),Large scale news video database browsing and retrieval via information visualization, *in* 'ACM symposium on applied computing', Seoul, Korea, pp. 1086–1087.

[20] Kashino, K., Kurozumi, T. & Murase, H. (2003), 'A quick search method for audio and video signals based on histogram pruning', *IEEE Transactions on Multimedia* **5**(3), 348–357.

[21] Sikora, T. (2001), 'The MPEG-7 visual standard for content description - An overview', *IEEE Transactions* on Circuits and Systems for Video Technology **11**(6), 696–702.