1st International Conference on Recent Trends in Engineering & Technology, Mar-2012
Special Issue of International Journal of electronics, Communication & Soft Computing Science & Engineering, ISSN: 2277-9477

Content Based Video Retrieval Using Ranking Correlation, Motion and Color

Ms. Kainjan S. Kotecha Mr.Mahesh R. Sanghavi Dr Mrs. A.M. Rajurkar Dr. Rajeev Mathur

Abstract - The content-based video retrieval technologies are unable to satisfy the human requisites. A vital reason is the lack of content depiction that enables to form an association between the visual features and semantic conception in video.The fundamental research in content – based video retrieval is resulting in evolution of technologies to inevitably break down video, extract as well as represent content attributes from video source. Feature extraction is one of the primary progressions for analysis of video content. In the proposed scheme the features extracted are the motion, color as well as average intensities. The average intensities are calculated and ranked using ordinal measures. A video retrieval system that helps to find the similarity between video clips is proposed. The query is in the form of a query by video clip example.

Index Terms
Content based video retrieval, Ordinal measure, Color distribution, Spearman's rank correlation coefficient, Levenshtein distance.

I. INTRODUCTION

Now-a-days, users find real difficulty in managing an enormous amount of multimedia data. Due, to this the user's entry experience is suffered a lot. This hindrance is made extinct with the help of Content-based video retrieval.Since the former era, the issue of Content-based retrieval has become an important one.

Multimedia information consists of four different types of media, which are text, audio, images and video. One of the main research areas in this direction is about the retrieval of video sequences [1]. Finding the similarities of video sequences can be easily done by a human, using its complex visual perception system [2]. But, when the search is considered to be automatic along with the retrieval system situations grow complex. There are some methods that can be used to find the similarities of video sequences [3] [4], where each one has its own advantage and disadvantages [5].

When two video clips are compared to find a match between them, two important points need to be considered, first to find similarity between the successive frames , secondly, the similarity between the ranking order of the frames in the two video clips. The similarity between the above two if exist then , we can conclude the two video sequences to be matching.

Grounded on the fact, this paper proposes a video retrieval system to match the video sequences based on their similarity. In this paper, we present our work on the extraction and application of motion feature and color feature which is the most distinctive character of video. Hence, we propose to work on the extraction and application of motion and color feature, which are considered as the most distinctive features of video.

Our main target is to find the approximate similarity between a query video and a part of another video sequence available in a database. Althrough the progress of the work, the matching is to be done considering it as an approximate similarity. The remainder of the paper is organized as follows. Section II describes the feature extraction from a frame. Feature extraction as proposed in[3] considers only average intensity and motion. But introducing color feature increases the confidence rate of the system. We suggest the system because it gives more accuracy over [3]. Section III describes the algorithm for findingmatch between the videos considered as string. The matching results provided are not exact but approximate. Section IV is to compare two video sequences based on results obtained as per previous section. Section V and VI illustrate some empirical results and conclusion, with a brief discussion of our approach.

II. LITERATURE REVIEW

Deficiency in text-based retrieval method gave rise to content-based image retrieval methods. Content Based Image Retrieval, relies on the characteristics of low level features such as color, shape, texture. Quite a few models are now commercially available like Query By Image Content (QBIC)[6], Virage[7], Excalibur, Attrasoft and others.

Gong, Zhang [8] allows us to specify an example image (digitized photo) as the example of colors with spatial distribution, where a representative plane is divided into nine (3×3) subplanes.

Problems in image-based retrieval methods have encouraged increasing interest in the development of content-based video retrieval methods. As video is a collection of image frames in motion, it gives more descriptive information. Hence, a proper approach to video retrieval is made.

QBE with motion examples is an approach to retrieve intrinsic features of video data, i.e. the motion of objects appearing in video.The motion is typically represented using a motion vector (x,y) which indicates the displacement of a pixel or a pixel block from the current location due to motion.

Motion estimation techniques[9] form the core of video processing applications. Motion estimation extracts motion information from the video sequence. However, the most popular technique for extracting the motion vector is Block Matching Algorithm [9]. Some of the algorithms are Three Step Search [9], New three Step Search [10], and DiamondSearch [11].

A generic approach for managing video data is first to segment a video into groups of related frames called "shots" by means of shot detection or scene break detection [12], [13]. After identifying the shot boundaries, one or more key frames or representative frames can be extracted for each group of frames (GOF) or video shot. The visual contents of these key frames are then used to represent the video shots for indexing and retrieval [14]. Key frame selection is therefore an important procedure for video management. Paul Browneand Alan F. Smeaton [15] provide Video Retrieval Using Dialogue, Key frame Similarity and Video Objects.

Bhatt and Nayar [16] proposed ordinal measures of association for image correspondence in the context of stereo. By using distance metrics between two rank permutations, ordinal measures are defined. These measures are independent of absolute intensity scale and invariant to monotone transformations of intensity values like gamma variation between images.These measures serve as a general tool for image matching that are applicable to other vision problems such as motion estimation and texture-based image retrieval.

Content based video retrieval system can thus be designed using primitive features. The proposed scheme uses ordinal measure(rank permutations) for average intensity, motion and color features that when compared with above is invariant to the changes, as well as since it considers all frames for comparison instead of only key frames. This feature ensures no data loss.

III. FEATURE EXTRACTION

The proposed retrieval model is as in Fig1. Our suggested features for extraction from blocks of the video frames include, average of intensity values, the motion direction and color feature of each block.

In the suggested system, ordinal measure is used to characterize the video sequences. Bhat and Nayar introduced the ordinal measure [16], and used the averages of intensity values to produce it. In our work we used motion feature and color feature in addition to the intensity values to generate the ordinal measure.

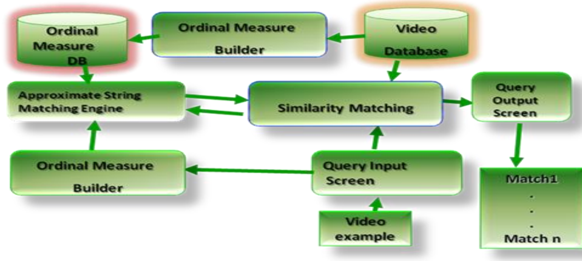


Fig. 1 Proposed Video Retrieval model

Ordinal measure is based on ranks rather than feature values themselves, and order of the extracted features will be used instead of their values.First of all the videos are converted into frames .Every video frame is then divided into n*n non overlapping windows (Fig. 2), then, the desired features will be extracted from each block.

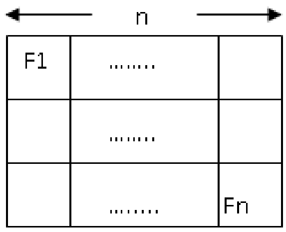


Fig. 2 Divide video frame into blocks

Consider a video frame of 180 × 120 as shown in Fig. 3a) Dividing it into 3 × 3 window, we get a block-size as 60 × 40 depicted in Fig. 3b). Average intensity of each of these 9 blocks is calculated with that of (1).

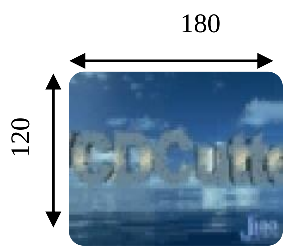


Fig. 3a) Video example

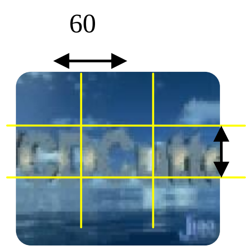


Fig. 3b) Divide into 3×3 blocks

$$Average\ Intensity = \frac{\sum_{i=0}^{i=m\times n} I_i}{m \times n} \qquad (1)$$

Where,
Ii = Intensity values in the block of frame

m × n = size of one block of video frame.

In the similar manner if video frame is divided into 3×3 blocks, 9 average intensity values are obtained as illustrated in the example below. Consider the following example in Fig. 4 (a) of a 3 × 3 window$R$ with average intensity obtained as I_1:

11652	7894	5423
19363	19327	17078
1808	9600	8680

5	2	1
9	8	7
6	4	3

(a) (b)

Fig 4 (a) Average intensity values of one frame (b) Ranking of average intensities of the frame

Now these values are ranked as shown in Fig. 4 (b) and then a string of feature vector for one frame is formed as F_1 = {5,2,1,9,8,7,6,4,3}. In this manner average intensities for all frames are ranked and combined together to form one feature vector as F_i, which is a combination of all frame values.

As an important cue in understanding video content, motion and color has been a study topic ever since computer vision and image processing research started. Motion estimation and color distribution is a conventional method to extract motion information from two consecutive frames.

Motion directions[22] may be found by comparing the successive frames, in video sequences using block matching algorithms. To reduce the amount of data, and make the results more useful, motion vectors will be quantized in 8 directions. Direction and angles of these candidate vectors are shown in Fig.5.

Motion vector is calculated using variable shape search method [17] to reduce the number of computations and find exact motion. Fig. 6 shows motion vector calculation using VSS.

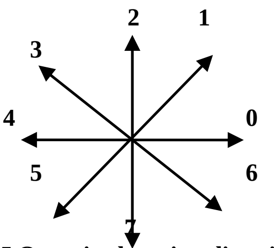
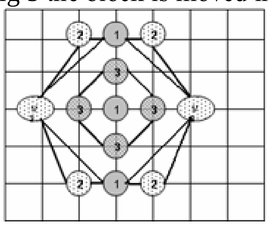
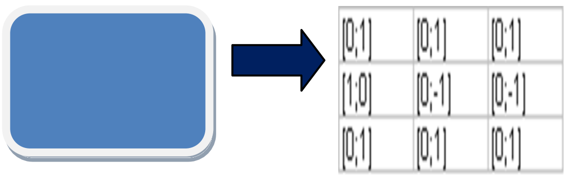


Fig. 5 Quantized motion directions

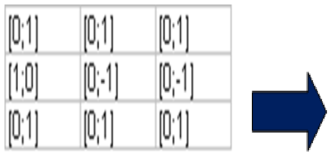
Consider for example the block of video frame compared with all the blocks of next video frame and its evaluated motion vector in Fig 6a. These motion vector are then ranked to form ordinal measures as depicted in Fig 6b.Here, [0;1] indicates that the block has no displacement in x-direction and 1 indicates 1 block displacement in y-direction. Therefore, according to Fig 5 the block is moved in ‘0’ direction.



a



b



0	**0**	**0**
6	**4**	**4**
0	**0**	**0**

c

Fig. 6. a)Variable shape search method b) Video block of frame and its motion vector.c) Motion vector and its ranked motion vector.

The motion vector is then formed after ranking as F1 = {0,0,0,6,4,4,0,0,0} for one video frame. In this manner motion vector for all frames are computed and ranked and combined together to form one feature vector as Fm, which is a combination of all frame values.

After the extraction of the average gray level and the motion direction for each block of a video frame, color feature of each block of video frame is also calculated. Color distribution of the frame is computed by 3D - (HSV) histogram [18]. The Fig 7.shows the query sample frame and the HSV histogram of that frame. First of all the histogram is obtained. Then, the number of bins in each direction (i.e., HSV space) is duplicated by means of interpolation.

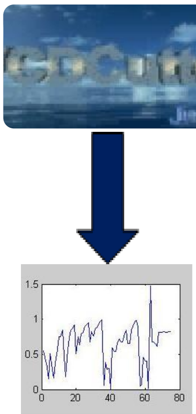


Fig. 7 The3D – HSV Histogram of sample frame.

Three set of feature values (say as F {f1,f2….fn,n) will be generated for each frame. In the next step, the order of each of the mentioned values is computed.

This is based on its location on ascending order of the set of sorted values of the same type for a specific frame. Finally we generate three strings, one for average intensities and one for motion directions, and one for color. These strings look likeX=x1, x2 ,xn.n which are composed of rank values for each of the blocks in a frame. The generated strings could be used as representer for each frame of the video sequence.

By generating the rank values of every video sequence in the database, we will have a representer set. But for the retrieval, the distance between rank permutations must be calculated. In this research, we suggest Spearman's rank for calculating distance between rank permutations.

Spearmen's rank correlation [19] finds how much two video frames are similar. By using formula in (2).

$$rank = 1 - \frac{6 \times \sum d^2}{n(n^2 - 1)} \qquad (2)$$

Spearmen's rank correlation coefficient

III.

IV. APPROXIMATE STRING MATCHING ALGORITHM

There are different approaches for approximate string matching, but the main used is algorithm proposed by Navaro and Baeza [20]. Levenshtein distance [21] between two strings is calculated to consider the no. of errors in between two strings. This algorithm works on the fact that two strings would be similar with 'ε' errors, then we can split each one into (ε+1) substrings; and we can be sure that it matches with exactly at least one of the substring.

A pseudo code of the suggested system can be found at Fig. 8. This is a recursive algorithm, and as can be found, the minimum similarity rate for the rank correlation is assumed to be α (α is the measure that two permutations of the rank values will be considered similar to each other, based on the spearman's rank correlation coefficient).

As an example if we consider A=2, j=2,a1=a2=1, then the pattern string will be divided into two halves (halving also the number of errors). In this case, the proposed video retrieval system states that at least one half must match with $k/2$ errors. This splitting, recursively continues by halving the sub patterns and number of the errors, until we reach the pieces that are to be searched directly, with no errors. This system introduced by [22] is known to be one of the fastest approximate string matching algorithm. Search cost for this algorithm is O(kn / m) . If we consider kn/m be small enough, then the cost becomes O(αn), which is linear, and is better than other similar algorithms.

```
Search ( X[1…m] , Y[1…n] ,k )
  1. Begin
  2. if(k>0) then
       Return ( Search (X[1…m] , Y[1…n/[2] ] ,
     k-1) AND (Search (X[1…m] , Y[n/[2] + 1…..n
     ], k-1));
  3. else
 "Search for an exact match of Y[1…n] in X[1…m]
 according to Spearmen's correlation formula"
  4. Loop from i = 1 to m-n
        similarity = false;
  5. Loop j = 1 to n
  6.    if(rank-correlation(X [j] ,Y[i] )<α then
           similarity = false;
       else
           similarity = true;
       end if
  7. Next j
  8. if similarity = true then break for
  9. Next I
  Return (similarity);
    End if
  End
```

Fig.8 Approximate String Matching Algorithm

V. APPROXIMATE STRING MATCHING ALGORITHM OVER VIDEO

To perform a query by example, with video examples, we should have a similarity matching algorithm, to approximately match video sequences. If a video sequence be considered as a string, then the video frames will be presented as words of that string. So, a textual approximate string matching algorithm over the ordinal measure sequences along with color is used to compare two sequences of the video frames.

During this process Spearmen's correlation coefficient is used for finding the similarity between two frames. This approach has many advantages over similar approaches. It is worth to mention that, different resolutions of the same copy of the video will be characterized in the same way. The proposed video retrieval system uses all frames to characterize the video. Thus the accuracy rate is increased.

If the key frames are used, then most of the information available in video frames may not be considered for characterizing the video. However, proposed video retrieval system has an advantage on this problem by using all the frames.

The algorithm proposed is shown as belows :-

1) Consider a .avi video file as query input.
2) Convert video(.avi) into frames.
3) Calculate average intensity of all frames of query video and rank them. Thereafter, the ranked average intensities of all frames are combined together to form feature vector *Fi* of average intensity.
4) Calculate motion vector of all frames of query video and rank them. Thereafter, the ranked motion vector values of all frames are combined together to form feature vector *Fm* of motion.
5) Compute the color histogram of all frames of query video and compare them with all frames using Bhattacharya coefficient of all videos in the database as discussed in section 3.4 and store the result.
6) Compare the combined feature vector Fm of the motion of query video with that of the combinedmotion feature vector of all videos in database using algorithm discussed in Fig. 8 and store its result.
7) Compare the combined feature vector *Fi* of the average intensity of query video with that of the combined average intensity feature vector of all videos in database using algorithm discussed in Fig. 4.3 and store its result.
8) Combine all the three results obtained in step 5, 6, 7. And goto step 9.
9) If the database video matches with the query results of step 8 , it is stored as a matched video along with its similarity measure.
10) Sort the videos according to their similarity measure
and display the sorted result.

VI. Empirical Results

This section first focuses on the results of scheme proposed in [22] that uses only ordinal measure for average intensity and motion. Due to less effectiveness in the old scheme a new scheme is proposed. The results of new proposed scheme content-based video retrieval using ordinal measures for average intensity, motion and color features are then shown. This approach uses offline process. The user has to specify the query avi video filename as input.

For implementing this scheme MATLAB 7.1 is used as a front-end and MS-Access is used as a backend. MATLAB 7.1 is used for feature extraction. The similarity algorithm explained in Fig 8, uses the feature values stored in database for comparing it with the feature values extracted as a result of query video, which is applied here for retrieval. This scheme is a simple user command-line interface where the user has to specify the query avi video filename as input.

Prior to this 8 avi videos with their feature values are computed and an offline database of these features is built. The following section shows the way to compute these features when block size specified is 33. The experiment is carried out on the following 8 videos stored in database as shown in Table I. These videos are downloaded as well as some of the videos are developed by us.

Table I. Video database

Sr. No.	Name of Video File	Preview of Video file	No. of Frames	Size of Frame	Time
1	Fox.avi		61	320x240	4 sec
2	Foxrain.avi		61	128x96	4 sec
3	test10.avi		20	176x144	1 sec
4	test11.avi		35	320x240	2 sec
5	test12.avi		17	320x240	1 sec
6	test14.avi		53	250x188	3sec
7	Testdemo.avi		10	180x120	1 sec
8	test.avi		30	180x120	2 sec

Applying algorithm discussed in section 5, the video retrieval results are shown in this section for block size of 3×3 for test12.avi. As shown in Fig. 9. test12.avi when given as query video retrieves total 4 videos and out of them 3 i.e test10.avi, test11.avi, and test12.avi are relevant combining motion, average intensity and color features. Hence, precision rate becomes 75% in this case.

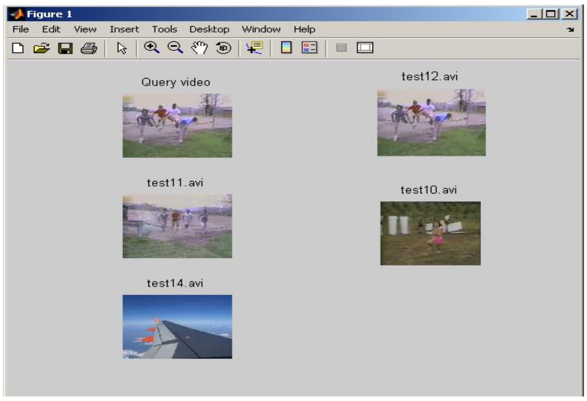


Fig. 9 Retrieved results for test12.avi

The retrieval performance is measured by using recall and precision. These measures are the standard measures used t compute the performance. Recall measures the ability of retrieving all or similar relevant videos from the database. It is a ratio of relevant videos retrieved and the total number of relevant videos in database. Precision measures the retrieval accuracy and is defined as the ratio between the number relevant retrieved videos and the total number of retrieved videos. The equations for the two are as follows :

$$Recall = \frac{No.\,Of\ Videos\ Retrieved\ And\ Relevant}{Total\ No.\,Of\ Relevant\ Videos\ in\ \ database}$$

$$Precision = \frac{No.\,Of\ Videos\ Retrieved\ And\ Relevant}{Total\ No.\,Of\ retrieved\ Videos}$$

Using above two measures, the performance is measured as shown in Table II. The same is graphically represented in Fig. 10. Looking to these tables as well as the graph, the 3rd column in Table II specifies the precision rates of the system proposed in [22]. Compared to these, the recall and precision rates of the new proposed scheme is better than old scheme increasing the search accuracy and also the effectiveness of the new proposed algorithm discussed in section.

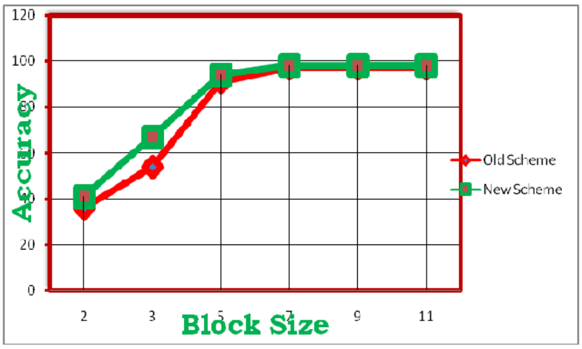


Fig 10 Graphical representation of Performance measure

VII. Conclusion

In this paper we present our current work developing a video retrieval system. This system works in a query by example manner. Query examples are video files and similarity matching is done by approximate techniques. Approximate matching is made possible by using an approximate string matching algorithm. Ordinal measure is used to characterize frames of the videos of the database and also queries. A promotion is made on ordinal measures by using motion features beside sum of intensities.

The future work will be on performing query by pictures on video database, with the same method. Also we plan to increase the search accuracy by using spatial features of the blocks of the video frames, besides current features.

Table II Recall, Precision rates of old scheme and proposed scheme

Sr.No	Block Size	CBVR With Average intensity, and motion	Proposed Scheme.. CBVR with average intensity, motion and color.	
		Precision Rate	**Precision Rate**	**Recall Rate**
1	2×2	34%	**41%**	**92%**
2	3×3	60%	**67%**	**100%**
3	5×5	91%	**94%**	**100%**
4	7×7	97%	**98%**	**100%**
5	9×9	97%	**99%**	**100%**
6	11×11	97%	**99%**	**100%**

References

[1] N. Dimitrova, H.J. Zhanhg, B. Shahrary,I. Sezan, T. Huang, A. Zakhor "Applications of Video-Content Analysis and Retrieval", *IEEE Multimedia,* Vol. 9, No. 4, 2002.

[2] Jiang, Y., Olson, I. R., & Chun, M. M. (2000). "Organization of visual-short term memory." *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 26(3), 683-702.

[3] Z. GharibanMasihi, N. MoghadamChakari, " Content based video retrieval using approximate string matching" ,*EuroCon2005,* Serbia and Montenegro,Belgrade, Nov 22 – 25 2005.

[4] S. Antani, R. Kasturi, R. Jain, "A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video", *Pattern recognition* 35 (2002) 945-965.

[5] M.K. Mandal, F. Idris, S. Panchanathan, "A critical evaluation of image and video indexing techniques in the compressed domain", *Image and Vision Computing 17* (1999), 513-529.

[6] M. Flickner, H Sawhney, W. Niblack, J. Ashley,Q Huang, B. Dom, M.Gorkani, J Hafhe, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by Image and Video Content: the QBIC system," ***IEEE** Computer*, **1995.**

[7] A. Gupta and R. Jain, " Visual information retrieval," *communications of the ACM,* vol. 40, no. 5, 1997

[8] Y. Gong, H. Zhang, H.C. Chuan, and M. Sakauchi, "An Image Database System with Content Capturing and Fast Image Indexing Abilities," *Proc. Int'l Conf. Multimedia Computing and Systems*,pp. 121-130, May 1994.

[9]Milind Phadtare, "Motion estimation techniques in videoprocessing",*Electronics Engineering Times*, August 2007

[10] Renxiang Li, Bing Zeng, and Ming L. Liou, "A New Three-Step Search Algorithm for Block Motion Estimation", *IEEE Trans. Circuits And Systems For Video Technology,* vol 4., no. 4, pp. 438-442, August 1994.

[11] Shan Zhu, and Kai-Kuang Ma, " A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation", *IEEE Trans. Image Processing*, vol 9, no. 2, pp. 287-290, February 2000.

[12] K.-W. Sze, K.-M. Lam, and G. Qui, "Scene cut detection using the colored pattern appearance model," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona,

[13] C.-L. Huang and B.-Y. Liao, "A robust scene-change detection method for video segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no.12, pp. 1281–1288, Dec. 2001.

[14] H. S. Chang, S. Sull, and S. U. Lee, "Efficient video indexing scheme for content-based retrieval," *IEEE Trans.Circuits Syst. Video Technol.*,vol. 9, no. 8, pp. 1269–1279, Dec. 1999.

[15] Paul Browne, Alan F. Smeaton, "Video Retrieval usingdialogue, Key frame similarity and video objects.",*IEEE Comp Mag.*, 2005.

[16] Dinkar N. Bhat and Shree K. Nayar, "Ordinal Measures for Image Correspondence" , *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. 20, No. 4, April 1998.

[17] Liu Hao, Zhang Wen- Jun, CaiJun ," A fast block-matching algorithm based on variable shape search" , *Journal of Zhejang University SCIENCE A* 2006-07(2), 194-198

[18] L. Kotoulas , I Andreadis , "Color histogram content based image retrieval and hardware implementation", *IEEE Proc.-Circuits Devices Syst.*, Vol. 150, No. 5, October 2003

[19]Spearman, C., "The Proof and K measurement of Association between Two Things", American Journal of Psychology, (1904)15,72-101.

[20] Navaro G. , Baeza-Yates R. , "Very fast and simple approximate string matching", Information Processing Letters 72 (1999) 65-70

[21] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals", DokladyAkademiiNauk SSSR, 163 (1965), no.4, 845--848. in Soviet Physics Dokl., 10 (1966), no. 8, 707--710.

[22] Z. GharibanMasihi, N. MoghadamChakari, " Content based video retrieval using approximate string matching",EuroCon2005, Serbia and Montenegro,Belgrade, Nov 22-252005

AUTHOR'S PROFILE

Kainjan S. Kotecha received a M.E in Computer Science from S.R.T.M.U , Nanded. She is currently an Associate Professor in SNJB's COE, Chandwad , She has published her technical contribution in IJCA, IJCSE Journal and also an International Conference. She has a teaching experience of nine years in the field of computer engineering .Her research interest includes algorithms, similarity searching in videos, video processing and digital data security.

Mr. Mahesh R. Sanghavi has secured ME in Computer Science and Engineering. Currently he is pursuing PhD from Jodhpur National University, Jodhpur. He is also working as an Associate Professor & Head in Department of Computer Engineering at SNJB'S College Of Engineering, Chandwad. He has a teaching experience of nine years in the field of computer engineering. He has published papers in national and international conferences. His research interests include Image processing and digital data security.

Dr Archana Rajurkar received degree of BE (Computer engineering) and ME (Instrumentation) from Marathwada university Aurangabad and Ph.D (Computer science and Engineering) from IIT Roorkee, India. She joined as a faculty member in M.G.M.'s College of Engineering, Nanded in 1991. Currently She is working as Professor and Head in the Department of Computer Science and Engineering, M.G.M.' College of Engineering, Nanded, India. Her research interests include Content-Based Image and Video Retrieval, Multimedia and Image Databases, Computer Vision and Pattern Recognition

Dr. Rajeev Mathur has completed his BE, MTech in Computer Science & Engineering and Ph.D from Jai Narayan Vyas University, Jodhpur, India. Currently he is working as Professor and Director of the Computer Science Department in Lachoo Memorial College of Science & Technology, Jodhpur, India. His research interests include Data Mining, Image Processing & Pattern Recogniton.