A content based video retrieval scheme

Payel Saha
TCET, Thakur Village,
Kandivali (E), Mumbai -101
09819412736
payel.saha@thakureducation.org

Sudhir Sawarkar
D.M.C.E.
Airoli-708
09819768930
sudhir_sawarkar@yahoo.com

ABSTRACT

The proposed system retrieves similar video clips for a query video clip from a collective set of videos. The first step is to partition a long video sequence into several video shots, i.e. shot segmentation, where each shot is the basic unit for video retrieval. In the next step, the system extracts ‘motion features’ for every video shot and the extracted ‘motion features’ will be stored in the feature library. Then, the same features will be extracted for a query clip and compared with the features in the feature library. With the aid of Kullback- Leibler distance similarity measure, the comparison will be carried out. Finally the videos will be retrieved from the videos collection on the basis of Kullback- Leibler distance.

Categories and Subject Descriptors

Image Processing and Computer Vision

Keywords

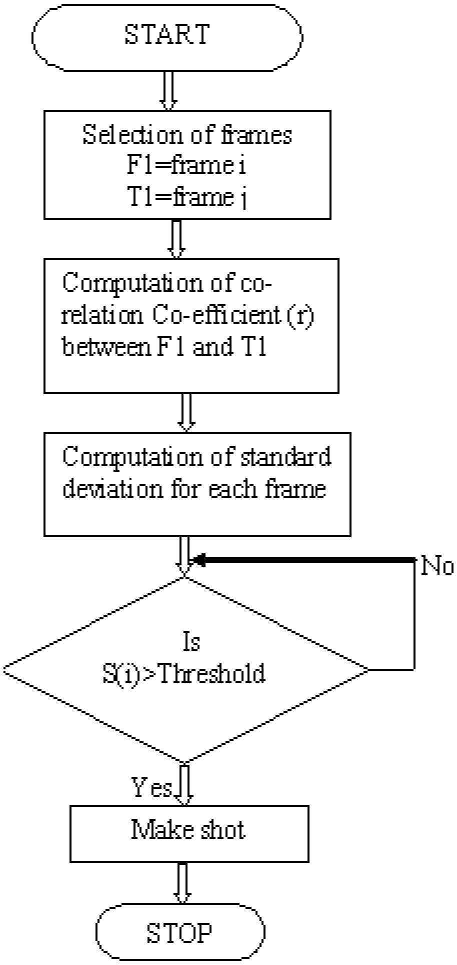
Video retrieval, discrete cosine transform, shot segmentation, co-relation co-efficient, Mean and standard deviation, motion estimation, block matching, feature selection

1. INTRODUCTION

Traditionally, the initial step in a majority of available content-based video analysis techniques is to segment a video into elementary shots, each of them constituting a sequence of consecutive frames recording a video event or scene continuous in time and space. These elementary shots are organized to form a video sequence during video sorting or editing with either cut transitions or gradual transitions of visual effects such as fades, dissolves, and wipes.
Here a 2-D Correlation Coefficient technique has been used for video shot segmentation. Moreover, discrete cosine transform, mean and standard deviation has been carried out over the video sequence to segment the video shots.
The video database is a collection of video sequences. The individual videos are split into separate shots followed by the tracking of the video objects across frames within every shot. The first step is to partition a long video sequence into several video shots, i.e. shot segmentation, where each shot is the basic unit for video retrieval. In the next step, the system extracts motion feature for every video shot and the extracted video feature are stored in the feature library. Then, the same features (aforesaid) are extracted for a query clip (single clip) and are compared with the features in the feature library. With the aid of Kullback-Leibler distance similarity measure, the comparison is carried out. Finally the videos are retrieved from the video collection on the basis of Kullback-Leibler distance.

2. Shot segmentation (SS):

Shot segmentation is applied on video database. A shot can be defined as a sequence of frames taken by a single camera without any significant change in the color content of consecutive images. A number of researchers utilized robust techniques on basis of the color histogram comparison to accomplish this function.


START
Selection of frames
F1=frame i
T1=frame j
Computation of co-relation Co-efficient (r) between F1 and T1
Computation of standard deviation for each frame
No
Is
S(i)>Threshold
Yes
Make shot
STOP

2.2 Correlation coefficient

A similar process will be carried out in the 2nd frame of the video sequence followed by the computation of correlation coefficient between frame 1 and 2 by means of the following formula.

$$r = \frac{\sum_m \sum_n (A_{mn} - \overline{A})(B_{mn} - \overline{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \overline{A})^2)(\sum_m \sum_n (B_{mn} - \overline{B})^2)}}$$

……………….(1)

where

$$\overline{A} = mean^2(A)$$

$$\overline{B} = mean^2(B)$$

Once correlation features of the frames are determined, the 1st frame of the video sequence is replaced with the 2nd frame. This process is carried out for every single frame in a video sequence. The feature library stores the correlation features of all the frames. An M -dimensional feature vector a_i, is computed for each frame f_i. The matrix is obtained with a_i as a column.

$$A = [a_1]……………[a_j].$$

…………..(2)

Mean and standard deviation

We construct the MxN feature matrix A with the aid of such a feature vector as a column. The following formulas aid n the calculation of the mean and standard deviation of the correlation features.

$$\overline{x_j} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}$$

$$\sigma_j = \sqrt{\frac{\sum_{i=1}^{M} (u_{ij} - \mu_j)^2}{M-1}}, 1 \le j \le N$$

…………..(3)

Eventually, we find the Shot/Cut changes took place in the video sequence on basis of the value of mean and standard deviation.

Motion Estimation:

Motion is the most significant feature in video which represents two dimensional temporal change of video content despite the conventional image features including color, texture and shape. It is possible to distinguish video and images in terms of motion. Numerous applications including motion based segmentation and structure from motion, utilize the motion information. This sub-section describes the estimation of motion. A number of significant applications in the areas of computer vision and video processing also employ the process of estimation of motion. Motion compensation based video coding is one application which encompasses motion estimation technique as

$$X(k)=\sum_{j=1}^{N} x(j)\,w_N{}^{(j-1)(k-1)}$$

where $w_N = e^{(-2\pi i)/N}$

……………(5)

The $L2$ -norm is also referred as the Euclidean Norm. For a function, the $L2$ -norm is defined as.

$$\|\Phi\|^2 \equiv \Phi.\Phi \equiv \left\langle \Phi \mid \Phi \right\rangle \equiv \int_a^b [\Phi(\mathrm{x})]^2 dx$$

……………………(6)

The aforesaid procedure is repeated for every block in the frame followed by the indexing of motion vector of the first frame and in turn followed by the application of the same process on 2nd frame and the frame adjoining it. Similarly the process is executed repeatedly for all the video frames in the sample video sequence. Later, a threshold value is set for video sequence. All

a direct appliance..Motion can aid to find interesting objects in the video.Our project recognizes an approach for motion estimation.

Let us regard a sample video sequence comprising some of set frames. Primarily the color frames are transformed into grey scale followed by the Selection of 1st and 2nd frame from the sample video sequence. The non-overlapping blocks of size 8x8 are extracted from the both the frames. Later, the blocks in the first frame are evaluated against the blocks in the second frame through FFT and $L2$ -norm distance. At first, FFT is applied to the blocks. Subsequently, the difference between the two blocks is determined on basis of

the measured distance of the frames are compared by keeping this threshold value in mind and the sample video sequence is classified into static and motion object, to form the motion vector.

Assumptions:

Block size = 8;

Thres is Predefined threshold value

Ff is First frame in a shot;

Nf is Next frame in a shot

BFf is Block of first frame;

BNf is Block of next frame MV

Mv is Motion vector

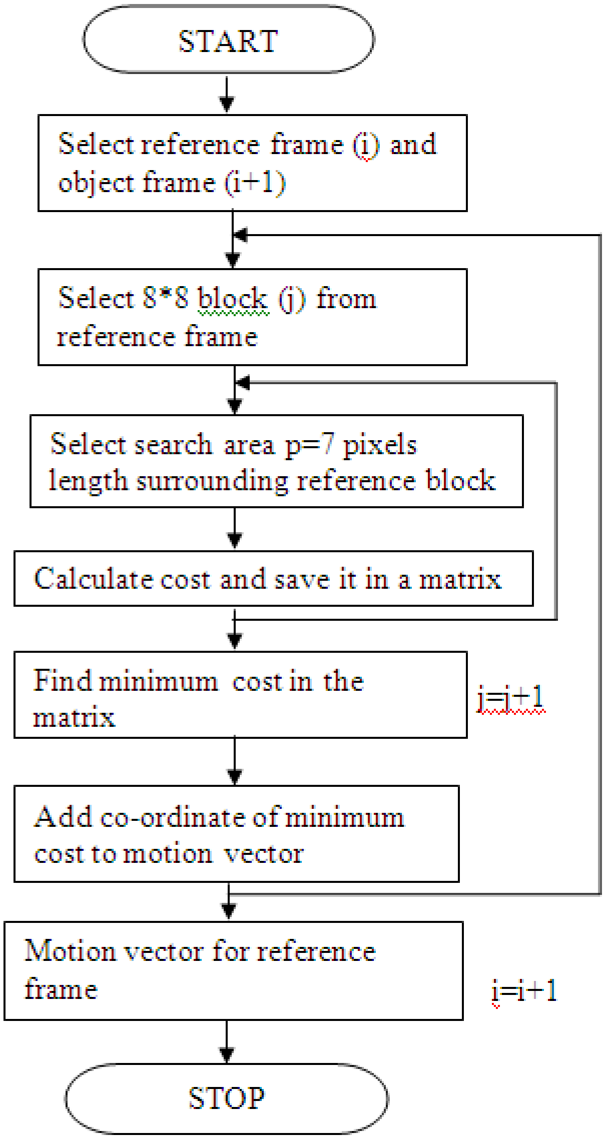
3. Block matching for motion vector extraction

To extract the motion intensity, the motion of the moving objects has to be estimated first. This is called motion estimation. The commonly used motion estimation technique in all the standard video codec is the block matching algorithm (BMA). Block matching is used to retrieve an initial estimate of the image displacement. To obtain a dense displacement field, matching with adaptive block sizes was implemented. In this typical algorithm, a frame is divided into blocks of M × N pixels or, more usually, square blocks of N2 pixels. Then, we assume that each block undergoes translation only with no scaling or rotation. The blocks in the first frame are compared to the blocks in the second frame. Motion Vectors can then be calculated for each block to see where each block from the first frame ends up in the second frame.

Motion information in digital video usually comes in the form of vector fields. A motion vector is estimated by extracting a macro block (an N X M block of adjacent pixels) from a frame and then seeking its best match in another frame in the future, which can be the very next frame or n frames away. A motion vector thus provides us with a description of the movement of a certain part of the frame and the object the block belongs to. The vector field can either be dense, where the surrounding block of each pixel is used for estimation, thus assigning a vector on every pixel, or sparse, where the macro blocks are non-overlapping, thus assigning one vector per block. This raw motion information is often of low accuracy and offers little insight as to the content of the video as is, but is, in most cases, the basis for the motion descriptors of a video shot.

To extract the motion intensity, the motion of the moving objects has to be estimated first. The commonly used motion estimation technique in all the standard video codec is the block matching algorithm (BMA). Block matching is used to retrieve an initial estimate of the image displacement.

Flow Chart


START
Select reference frame (i) and object frame (i+1)
Select 8*8 block (j) from reference frame
Select search area p=7 pixels length surrounding reference block
Calculate cost and save it in a matrix
Find minimum cost in the matrix
j=j+1
Add co-ordinate of minimum cost to motion vector
Motion vector for reference frame
i=i+1
STOP

To obtain a dense displacement field, matching with adaptive block sizes was implemented. In this, a frame is divided into blocks of M × N pixels or, more usually, square blocks of N2 pixels. Here it is assumed that each block undergoes translation only with no scaling or rotation. The blocks in the first frame are compared to the blocks in the second frame. Motion Vectors can then be calculated for each block to see where each block from the first frame ends up in the second frame.

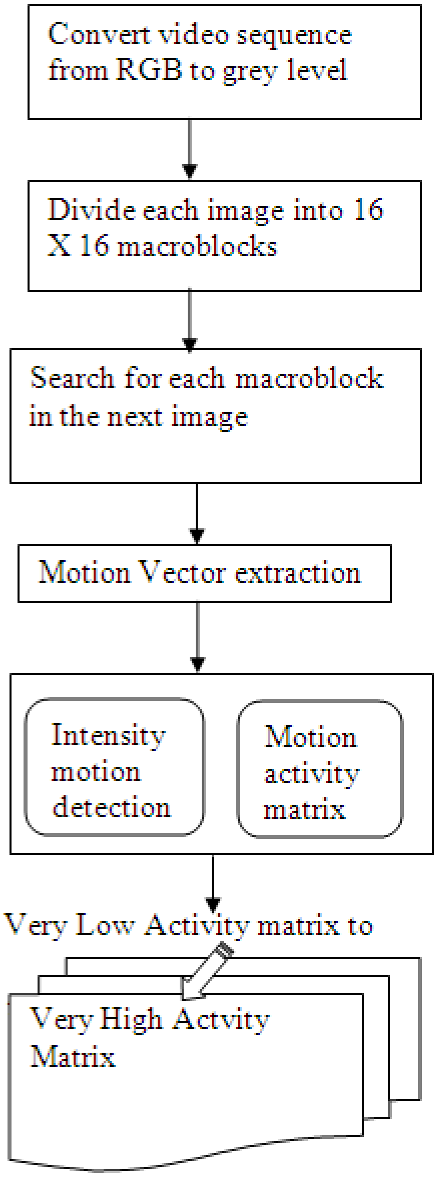

Convert video sequence from RGB to grey level
Divide each image into 16 X 16 macroblocks
Search for each macroblock in the next image
Motion Vector extraction
Intensity motion detection
Motion activity matrix
Very Low Activity matrix to
Very High Actvity Matrix

Fig 4.3.1 Different steps for motion intensity extraction

The basis of any video segmentation method consists in detecting visual discontinuities along the time domain. During this process, it is required to extract visual features that measure the degree of similarity between the frames in a given shot. This measure is related to the difference or discontinuity between frame n and n+j where j>= 1.

The main idea underlying the methods of segmentation schemes is that images in the vicinity of a transition are highly dissimilar. It then seeks to identify discontinuities in the video stream. The general principle is to extract a comment on each image, and then define a distance (or similarity measure) between observations. The application of the distance between two successive images, the entire video stream, reduces a one dimensional signal, in which the peaks are sought (resp. hollow if similarity measure), which correspond to moments of high dissimilarities. In this project, the key frames were extracted based on detecting a significant change in the activity of motion. First the motion vectors between images i and image i+2 are extracted and used to calculate the intensity of motion. The process is repeated until the last frame of the video is reached and the difference between the intensities of successive motion is compared to a specified threshold.

3.1 Feature Extraction:

Motion is the key feature representing temporal information of videos. The second step is to extract the motion features for each video shot. To minimize the dimensionality of the data, feature extraction which extracts preferably compact and discriminative features of data is employed. Motion estimation is performed with the help of Fast Fourier transform (FFT) and L2-norm distance. The feature library stores the extracted features. In the proposed system, the similar videos are retrieved on the basis of a specified query clip. Therefore, the motion features is extracted for a query video clip and compared with the features in the feature library. The similarity measure is employed to compare the query features and the features in the feature library. For similarity measure calculation the proposed system makes use of Kullback-Leibler distance method calculation. On the basis of the calculated Kullback-Leibler distance similar videos will be retrieved from the collection of video. After applying motion estimation the extracted feature is stored in the feature library.

3.2 Similarity Measure:

A query video is taken and its features are extracted based on above steps and compared with features stored in feature library. All the extracted features will be stored in a Mat file. Based on the input query clip, the related videos are retrieved from the video database. The ‘Motion features’ are evaluated for the query clip and compared against the features in the feature library. With the help of Kullback –Leibler distance which is employed as a similarity measure, the comparison of the features is achieved. Kullback and Leibler in 1951 studied from a statistical perspective, a measure of information that implicated two probability distributions associated with the same experiment. To determine the difference between two distinct probability distributions (over the same event space) the Kullback-Leibler divergence measure is used. The subsequent equation describes the KL divergence of the probability distributions P,Q on a finite set of X.

$$D_{KL}(P\|Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)}$$

Owing to the fact that *KL* divergence is a non-symmetric information theoretical measure of distance of *P* from *Q*, it is not specifically a distance metric. Therefore the following different symmetric Kullback-Leibler divergences i.e., Kullback-Leibler Distances (*KLD*) will be employed here. There are various applications including language models, query expansion, and categorization which have utilized *KL* and *KLD*. In addition, they have also been employed in natural language and speech processing applications on the basis of statistical language modeling, and in information retrieval, for topic identification.

Steps for Similarity Measure:

Let P:- □Query clip feature vector

Q:-□ Feature library 1st feature vector

n:-□ Element of vector

N:- □Normalized factor of Q

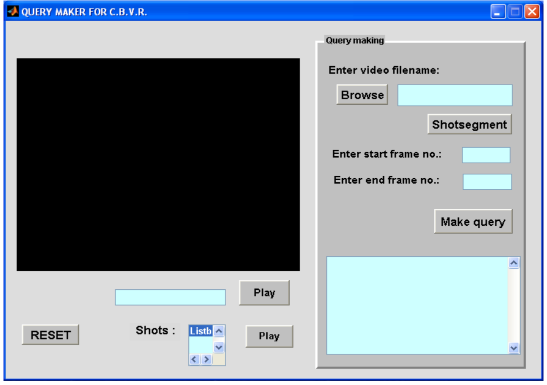
Then find ((Q>0) & (F>0)) and store that in F1. Then similarity measure is carried out using [6]:

$$D_{KL} = \sum F(F_I) \log \frac{N * F(F_I)}{Q(F_I)}$$

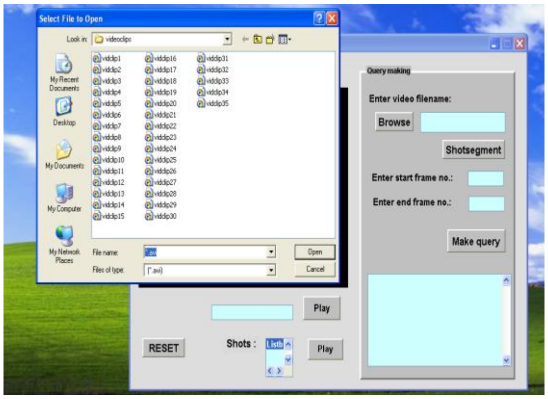
4. Results

Steps for Query Making

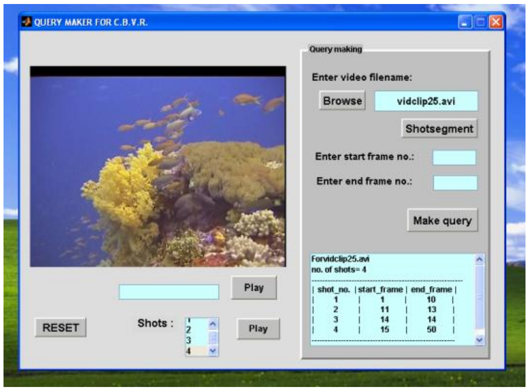
Step1: Open Query Maker GUI



Step 2: Press Browse button and select the video for which the query needs to be made from the database.



Step 3: Press Shot segment button. This will make different shots of the video.

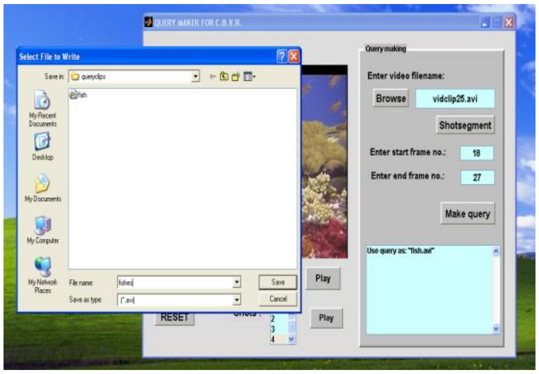


vidclip25.avi

```
Forvidclip25.avi
no. of shots= 4
----------------------------------------------
| shot_no. | start_frame | end_frame |
|     1    |      1      |     10    |
|     2    |     11      |     13    |
|     3    |     14      |     14    |
|     4    |     15      |     50    |
----------------------------------------------
```

Shots : 1 2 3 4

Step 4: Enter the start frame number and the end frame number. Save the query file by any filename.



File name: fishes

Save as type: (*.avi)

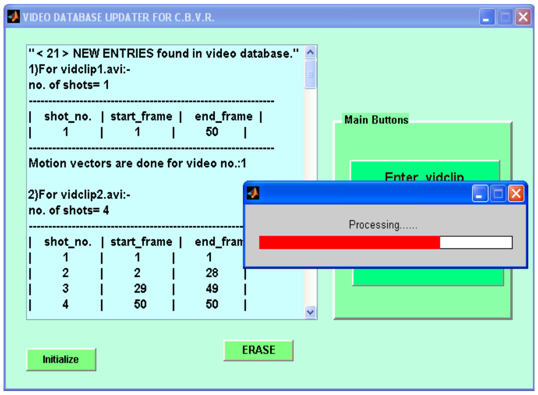
Enter video filename: vidclip25.avi

Enter start frame no.: 18

Enter end frame no.: 27

Use query as: "fish.avi"

Video database updating and updating motion vector library for new videos.



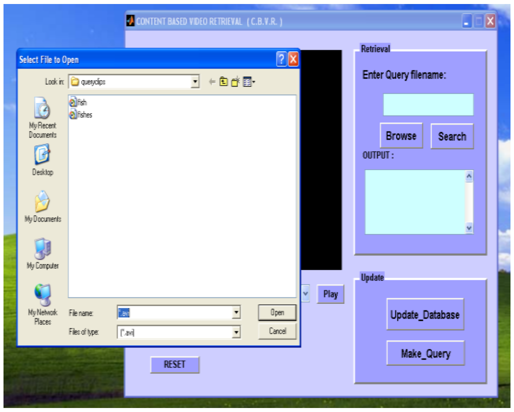
VIDEO DATABASE UPDATER FOR C.B.V.R.

```
"< 21 > NEW ENTRIES found in video database."
1)For vidclip1.avi:-
no. of shots= 1
------------------------------------------------------------
|   shot_no.   | start_frame |    end_frame  |
|       1      |      1      |       50      |
------------------------------------------------------------
Motion vectors are done for video no.:1

2)For vidclip2.avi:-
no. of shots= 4
------------------------------------------------------------
|   shot_no.   | start_frame |    end_fram
|       1      |      1      |       1
|       2      |      2      |       28      |
|       3      |      29     |       49      |
|       4      |      50     |       50      |
```

Processing......

Query video file as input.



CONTENT BASED VIDEO RETRIEVAL (C.B.V.R.)

Select File to Open

Look in: queryclips

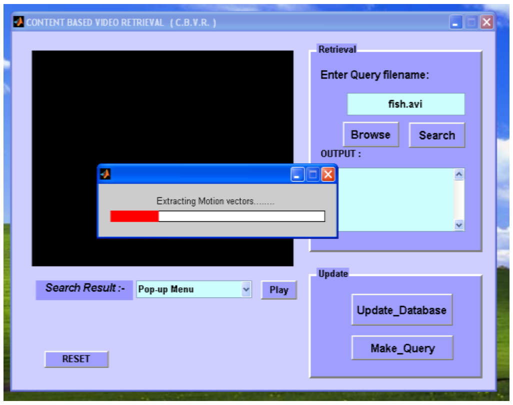
fish
fishes

File name: *.avi

Files of type: (*.avi)

Enter Query filename:

Extracting motion vectors for query video file.



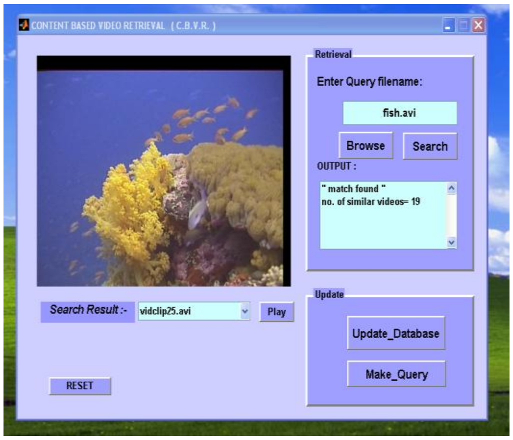
CONTENT BASED VIDEO RETRIEVAL (C.B.V.R.)

Enter Query filename: fish.avi

Extracting Motion vectors........

Search Result :- Pop-up Menu

Search results for query video file.



CONTENT BASED VIDEO RETRIEVAL (C.B.V.R.)

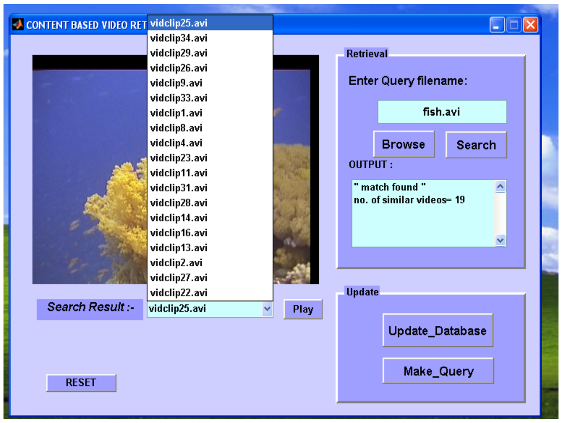
Enter Query filename: fish.avi

OUTPUT :

```
" match found "
no. of similar videos= 19
```

Search Result :- vidclip25.avi

List of similar videos for query video clip from database



CONTENT BASED VIDEO RET

```
vidclip25.avi
vidclip34.avi
vidclip29.avi
vidclip26.avi
vidclip9.avi
vidclip33.avi
vidclip1.avi
vidclip8.avi
vidclip4.avi
vidclip23.avi
vidclip11.avi
vidclip31.avi
vidclip28.avi
vidclip14.avi
vidclip16.avi
vidclip13.avi
vidclip2.avi
vidclip27.avi
vidclip22.avi
```

Enter Query filename: fish.avi

OUTPUT :

```
" match found "
no. of similar videos= 19
```

Search Result :- vidclip25.avi

5. Conclusion

Content-based retrieval of visual information is an emerging area of research which has been in limelight amongst the researchers and experimenters, recently. The proposed scheme of content based video retrieval system facilitates the segmentation of the elementary shots in the long video sequence. Subsequently, the extraction of motion vector features of the video sequence is performed and the feature library is employed for storage purposes. The Kullback-Liebner distance similarity measure is employed for successful comparison between the features in the feature library and the features of the query clip extracted in a similar manner. The computed Kullback-Liebner distance serves as the basis for the effective retrieval of the similar videos from the video database.

6. References

1. T.N.Shanmugam, Priya Rajendran, “Effective Content-Based Video Retrieval System Based On Query Clip”, Proceeding of the 2nd International Conference On Advanced Computer Theory and Engineering, vol.2 no.5, pp.1095-1102, September 2009.

2. Yang Liu, Weiqiang Wang, Wen Gao, Wei Zeng, "A novel compressed domain shot segmentation algorithm on H.264/AVC", in Proc. of International Conference on Image Processing, ICIP 2004, 24-27 October 2004.

3. Pennebaker, William B., and Joan L. Mitchell, JPEG: Still Image Data Compression Standard, Van Nostrand Reinhold, 1993.

4. K. Otsuka, T. Horikoshi, S. Suzuki and M. Fujii, “Feature Extraction of Temporal Texture Based on Spatio-Temporal Motion Trajectory”, Proceedings of the 14th International Conference on Pattern Recognition, ICPR‟98, pp.1047-1051, Aug. 1998.

5. Society of Motion Picture and Television Engineers, "Television - Signal Parameters - 1125-Line High-Definition Production", SMPTE 240M-1999.

6. B. Bigi, “Using Kullback-Leibler Distance for Text Categorization”, In Proceedings of the ECIR-2003, volume 2633 of Lecture Notes in Computer Science, pp. 305-319, Springer-Verlag, 2003

7. J. Meng, Y. Juan, S.F. Chang, Scene Change Detection in a MPEG Compressed Video Sequence, SPIE Symposium on Electronic Imaging: Science and Technology - Digital Video Compression: Algorithms and Technologies, SPIE Vol. 2419, San Jose, Feb. 1995.

8. Vikrant Kobla, Daniel Dementhon, David Doermann, “Detection of slow-motion replay sequences for identifying sports videos”, University of Maryland.

9. B.V.Patel, A.V. Deorankar, B.B.Meshram, “Content based video retrieval using entropy, edge-detection, black and white color features”, Proceedings of 2nd International Conference on Computer Engineering and Technology, Vol 6, 978-1-4244-6349-7/10 2010 IEEE Proceedings.

10. Ramin Zabih, Justin Miller, Kevin Mai, “Video browsing using edges and motion” 1063-6919196 $5.00 1996 IEEE Proceedings.