

# Frame based Video Retrieval using Video Signatures

Siva Kumar Avula  
Assistant Professor  
Dept. of Computer Science & Engg.  
Ashokrao Mane Group of Institutions-Vathar

Shubhangi C Deshmukh  
Assistant Professor  
Dept. of E & TC  
Ashokrao Mane Group of Institutions-Vathar

## ABSTRACT

The World Wide Web today has grown so wide and the video-on-demand applications and video share web are becoming very popular day-by-day on the World Wide Web. An efficient video similarity search algorithm for content-based video retrieval is important in video-on-demand based services. However, there is no satisfying video similarity search algorithm showing cent percentage performance.

It is proposed here to implement a video similarity measure algorithm based on the color-features of each video represented by a compact fixed size representation known as Video Signature. This Video signature which is based on the image signature is computed on the basis of YCbCr Histogram and the sum of its weighted means. The video signatures of videos are then used to find the similar videos in-terms of visually similar frames, by using the range. This method of similarity measure is assumed to be efficient in various aspects.

## Keywords

YCbCr, Video Signature.

## 1. INTRODUCTION

The proliferation of video content on the web makes similarity detection an indispensable tool in web data management, searching and navigation. Web video clips are often stored in multiple locations, compressed with different algorithms and bit rates to facilitate downloading and streaming. Similar versions, in part or as a whole, of the same video can also be found on the web when some web users modify and combine original content with their own productions. Identifying these similar contents is beneficial to many web video applications.

As video is a complex data type, many of the proposed similarity measures are computationally intensive. On the other hand, for every new video added to the database or a query video presented by a user, similarity measurements need to be performed with possibly millions of entries in the database. Thus, it is imperative to develop fast methods in computing similarity measurements for database applications.

Finding visually similar content is the central theme in the area of Content-Based Information Retrieval. In the past decade, numerous algorithms have been proposed to identify visual content similar in color, texture, shape, motion and many other attributes. These algorithms typically identify a particular attribute of an image or a video shot.

A Number of video search engines are in existence which search for videos based on filename, file type, location, and file size. These are called as 'Text Based Video Retrieval Systems' [1]. They do not provide searching based on contents of video such as a frame, snapshot or a short video

sequence which is required in some real life applications. In the text-based video retrieval system, videos can be retrieved based on keywords, i.e., video name as an input and based on the name, videos having similar names are retrieved. For example, to search a video of a particular cricket match from a large database, suppose the input as 'ICC world cup final cricket match'. But, if the videos in the database have same string as tag or keyword, system will display many results which may contain many irrelevant videos. If the keyword for a tennis match video is assigned as 'cricket match', the irrelevant video of tennis match is retrieved when user searching for 'cricket match'. To improve the efficiency of these systems, 'Content-Based Video Retrieval Systems'[2] are developed. This project involves implementation of a Content-Based Video Retrieval Systems using video similarity algorithm which provides the similarity measure between two video sequences.

The application developed has a database of videos that are pre-processed, and it takes as input a search video. The pre-processing contains the summarization of the video and generation of Video Signature. The given search video is first summarized and represented by a Video Signature. It then identifies all valid video files decoded by the decoder from the database. Video Signatures of all these identified files are generated. The similar videos of the search file are then displayed. There is also a provision to play any selected file and the search file simultaneously. This feature enables user to visually verify the result obtained. The Video Signature can be generated using the color, texture, shape or spatial relationships of the video frames.

The proposed System, "Content Based Video Retrieval Using Video Signatures" will perform the similarity search based on Video Signatures [3] based on the YCbCr[4] values of the video frames.

## 2. RELATED WORK

Locality-Sensitive Hashing (LSH) [6] describes that, let us assume that user database is a set of vectors  $x_1, \dots, x_n$ . Given a query vector  $q$ , user is interested in finding the most similar items in the database to the query. The basic idea behind LSH is to project the data into a low-dimensional binary (Hamming) space; that is, each data point is mapped to a  $b$ -bit vector, called the hash key. If this projection is performed appropriately, system can find approximate nearest neighbors in time sub-linear in  $n$ . The hash keys are constructed by applying  $b$  binary-valued hash functions  $h_1, \dots, h_b$  to each of the database objects. In order to be valid, each hash function  $h$  must satisfy the locality sensitive hashing property:

$$Pr[h(xi) = h(xj)] = sim(xi, xj)$$

A video copy detection technique tries to find video sequence whose contents are identical to, or near-duplicates of, the source. Two indexing algorithms, the min-hashing algorithm

and the histogram pruning algorithm, are integrated based on the histogram's characteristics. The existing system gives the features of the ordinal signature and the SIFT descriptor individually for evaluation.

A unified approach is proposed to detect both unknown video repeats and known video clips of arbitrary length. Two detectors in a cascade structure are employed to achieve fast and accurate detection. In this approach very short video repeats (<1 s) and long ones can be detected by a single process, while overall accuracy remains high. Since video segmentation is essential for repeat detection, performance analysis is also conducted for several segmentation methods. Furthermore, A method to analyze video syntactical structure based on short video repeats detection is proposed.

Besides, several other feature extraction algorithms such as distinctive local descriptors, color fingerprint, DCT and visual rhythm spectrum etc were also proposed. For similarity measurement, many approaches have also been proposed such as neighborhood similarity considering local sample and label distribution.

The nearest- or near-neighbor query problems arise in a large variety of database applications, usually in the context of similarity searching. Of late, there has been increasing interest in building search/index structures for performing similarity search over high-dimensional data, e.g., image databases, document collections, time-series databases etc., The existing system solved approximate similarity search based on hashing. The basic idea is to hash the points from the database so as to ensure that the probability of collision is much higher for objects that are close to each other than for those that are far apart.

### 3. CBVR SYSTEM OVERVIEW

Two main aspects must be considered in order to implement a CBVR system:

- The way video content is represented.
- What kind of queries is allowed in the system?

#### 3.1 Video Signature Generation

As mentioned above, there is a process where the video content is summarized in a symbolic representation named *signature*. Once the video is segmented in shots, the content of each shot is processed to generate a set of concatenated symbolic representations, one per detected shot, which composes the whole video signature. There is no general agreement on the way key frames should be selected, so several approaches can be followed. Some authors choose the first frame, the last one, or one of the inner frames randomly selected. Another approach is to select the key frame taking into consideration low level features extracted from the central frames.

In the user point of view, It is preferred to make an automatic extraction of the key frame computing the differences between consecutive frames and selecting as the key frame the one that is the minimal of the whole sequence of differences inside the shot. This strategy has the advantage of being a complementary criterion of the shot detection process, so both computations can be done in a single step.

#### 3.2 CBVR System Operation

When the user introduces a query in a CBVR system, the result from the search is a list ordered according to the level of

similarity of its signature with the signatures stored in the video database. This list contains the best n matches of the query. CBVR systems accept two different inputs taking into consideration the nature of the data involved in the query: static images and sequences of images that compose a shot extracted from a video. The problem of video matching, defined as the process that allows the comparison between pairs of video sequences, is far away from the scope of this paper.

In the first case, the problem to be solved consists on finding the video that best matches the searched frame, comparing the signature from the query with respect to the signatures of the key frames extracted from the database.

The output of the system can be a sorted list of the n most similar key frames. When dealing with shots as input data, a preprocessing step is required in order to extract the key frame of the query. Then, it will be compared against the video database signatures in a similar way to the frame case. In both cases the user can select the detail of the output: video or shot level. At video level, the system shows only the video key frame, achieving only one key frame per video, but at shot level, the system returns a key frame per shot. In this case, it may appear several frames belonging to the same video. Should the users then consider the search result to be unsatisfactory, they may select as a new input one of the displayed key frames which is most similar to the original query and then restart the retrieval process. In both cases, the classification method used in the matching process has been a minimum distance classifier.

It provides a list of all query-by-video-clip methods developed in the past decade. According to different user requirements, video search intentions can be generally classified into three main categories: detecting instances of copies, identifying visual similarities of near duplicates, and retrieving entries in a broad sense of high level concept similarity.

### 4. Color and Texture Feature

COLOR and TEXTURE are among the more expressive of the visual features. Considerable work has been done in designing efficient descriptors for these features for applications such as similarity retrieval. For example, a color histogram is one of the most frequently used color descriptors that characterize the color distribution in an image. The system provides the reader with an overview of the technologies that are being considered by the MPEG-7 group for describing visual content based on its color and texture. More detailed information regarding the color and texture descriptors in MPEG-7 may be found in the references and other related MPEG documents. The color and texture descriptors that are described in this paper have undergone rigorous testing and development during the past two years, and thus represent some of the more mature technologies for content representation. These tests and development were conducted under the various Core Experiments defined by the MPEG Video group and its *Ad-Hoc* Group on Color and Texture Core Experiments.

### 5. PROPOSED SYSTEM

The proposed Frame Based Video Retrieval system is a video similarity based search engine. Similarity measure finding the similarities between two video sequences in terms of percentage of visually similar frames. According to the research paper "AN EFFICIENT VIDEO SIMILARITY SEARCH STRATEGY FOR VIDEO-ON-DEMAND

SYSTEMS” by Cao Zheng, Zhu Ming.[5], a compact video signature was computed using image histogram and spatial-temporal features of video. The video similarity is measured by the computation of the distance of video signature.

In a video similarity search system, each video in the database can be summarized into a compact fixed-size representation called as Video Signature. Then, the similarity between two video sequences can be approximated by comparing their corresponding representations. In this paper, an efficient algorithm for generation of video signatures, based on the color features of YCbCr color space is proposed. The feature extraction of image and video was achieved by the statistics based on color space. The high-dimensional feature was transformed into compact video signature and the video similarity was measured by the computation of the video signature.

YCbCr histogram can present valuable information for video matching, which can be considered as an effective video signature. [YCbCr: Y = Luma component (Brightness) Cb = Blue difference (Blue minus Luma) Cr = Red difference (Red minus Luma)].

The video signature based similarity measurement involves the following steps:

1. Generation of Video Signatures for the available videos in the database.
2. Creating a vector, to maintain the similar videos.
3. Searching for the required video, by using a query clip.
4. Generate Video Signature for the query clip.
5. Similarity is measured for videos, using video signatures.
6. Retrieval of the required video, along with similar videos.

The video signature is computed on the basis of image signature. A frame from the video is selected. Basically the color space of the image initially will be in RGB. User need to convert it into YCbCr color space. The YCbCr value can be extracted by converting RGB to YCbCr. The YCbCr holds the Y, Cb and Cr values. In Matlab, first column of the YCbCr matrix holds the Y component, second column holds Cb component and the third holds Cr value. The mean values of each component are calculated. Then the median values of the mean values are calculated. The video signature (S) is obtained by summation of the median values of each component in the proportion 2:1:3.

#### **Generating Video Signature:**

$$\bar{y} = \text{mean}(Y); \bar{cb} = \text{mean}(Cb); \bar{cr} = \text{mean}(Cr)$$

$$\tilde{y} = \text{median}(\bar{y}); \tilde{cb} = \text{median}(\bar{cb}); \tilde{cr} = \text{median}(\bar{cr})$$

$$\text{Signature } S = 2\tilde{y} + \tilde{cb} + 3\tilde{cr}$$

Here, Y holds the values of brightness component, Cb holds the blue-difference and Cr holds the red-difference. Since people are generally sensitive to red color, the biggest weight was used to multiply Cr component. In this way, the high-dimension feature is transformed into compact image signature.

### **5.1. Outline of the System**

In contrast to the features provided by existing systems, this application is expected to provide the following features:

- The application should be able to find similarity between videos irrespective of the attributes.
- The application should decide on itself, the method of estimating similarity rather than user manually specifying the attribute of comparison.
- Given a portion of a complete video, the application should identify the percentage of similarity.
- The application should use an efficient image processing algorithm in terms of time complexity.

The Video Signature is used to retrieve the similar videos in the proposed system. The Video Signature that is generated will act as the unique identity of the video. So, there is more feasibility of acquiring the more relevant videos of the query video, which yields better precision rate. And, user can reduce the time complexity. Thus, the proposed system is better than that of the existing system.

In this system, it is required to preprocess the data, i.e. videos in the database. The preprocessing includes extraction of the color (YCbCr) feature of each video and generate video signature for it. The video signature is stored in a vector. In the searching process, we have to select a query video clip and then generate a video signature for that. Once the video signature is generated, it is compared with the video signatures stored in the vector. If there are video signature values relevant to that of the query video, those videos are retrieved and can be played. As the video signature is computed using only visual data, the audio in the retrieved video may vary. The comparison of the video signatures is done on the basis of the range of set of values. All the retrieved videos will be in the same range.

The framework of the proposed system “Frame Based Video Retrieval Using Video Signatures” is shown in the below figure Fig.1. The overall system can be classified into four steps. They are:

1. Preprocessing videos
2. Generating Video Signatures & store them
3. Read & process query video clip
4. Compare & retrieve the videos

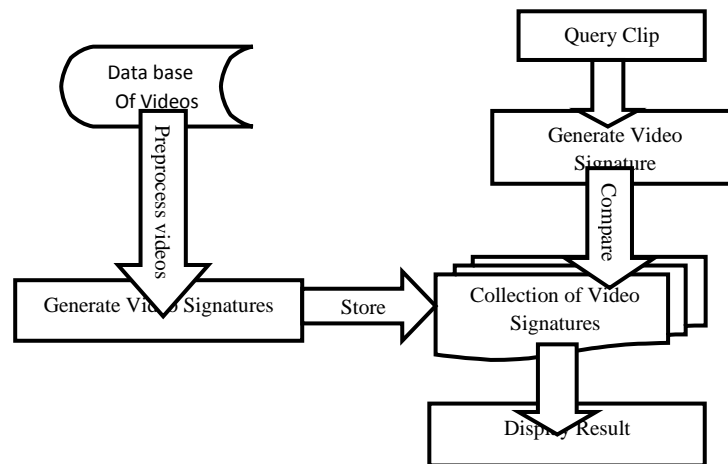


Figure 1. Overview of Frame Based Video Retrieval Using Video Signatures

Step 1: Initially, all the videos in the database are processed to make them to be in same video file format, same video codec and of same size. This is called preprocessing.

Step 2: The color features of the videos' frame data are extracted and based on those values, video signatures are generated. These generated video signatures are stored in a feature vector.

Step 3: A query video clip is selected and the color features of the query clip's frame data are extracted, and the video signature is then generated.

Step 4: By using the video signature of the query clip and the collection video signatures, a comparison is made. Then, if there are any relevant videos, they are retrieved.

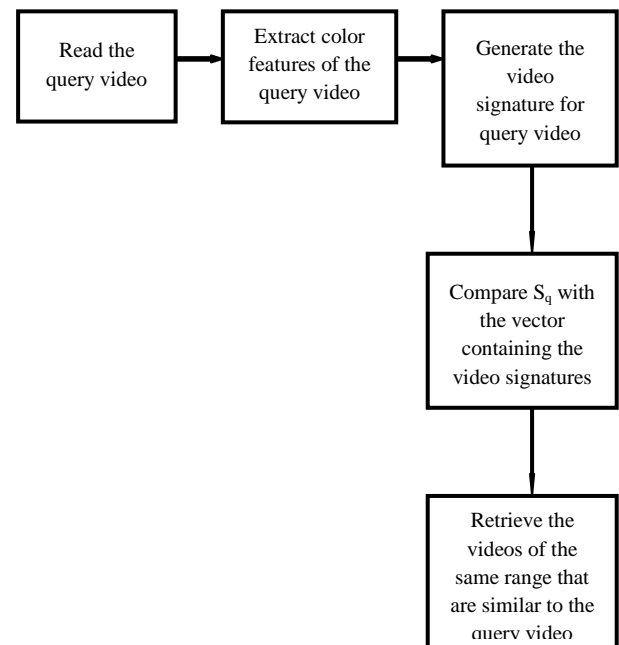


Figure 2. Process of Similarity Search

## 6. IMPLEMENTATION

Video signature is generally based on image feature extraction. It has been shown that YCbCr histogram is an effective video signature which can present valuable information for video matching. Since the volume of video database usually grows very fast, video signature should not be too complicate to compute for the computational efficiency. The video signature is computed on the basis of image signature. The ordinal measure of YCbCr histogram and the sum of weighted means were calculated for image signatures.

Extracting YCbCr Value which is converted from RGB calculate the video signatures for the database of videos using the following formulae

$$\bar{y} = \text{mean}(Y); \bar{cb} = \text{mean}(Cb); \bar{cr} = \text{mean}(Cr)$$

$$\tilde{y} = \text{median}(\bar{y}); \tilde{cb} = \text{median}(\bar{cb}); \tilde{cr} = \text{median}(\bar{cr})$$

$$\text{Signature} = 2\tilde{y} + \tilde{cb} + 3\tilde{cr} \text{ ----- ( eq )}$$

This module deals with the creation of the Graphical User Interface (UI) and integrating it with the similarity searching strategy. The GUI is to be designed in a way that the user feels ease to interact with the system. In the system, the query video should be browsed and selected. This can be done using GUI, by providing an option to browse the video file.

There should be an option to play the query video clip. This can be done by using the ActiveX controller. In this paper Video LAN-VLC plug-in is used to play the AVI video file. Similarly, GUI is providing an option to play the query video, as well as the result videos. To represent the videos, one frame image of all the videos will be displayed in thumbnail format and a play button to play the videos.

## 7. EXPERIMENTAL RESULTS

Unit testing is performed on each module as & when they are developed. Integration testing is then performed by integrating each of these modules together thus providing better efficiency & performance. Smoke testing is performed by testing the system as a whole on a daily basis and when new modules are added to the system. The factors considered in performing these tests are:

- Error handling paths
- Boundary conditions

- Data manipulation & analysis
- User interaction

The factors considered in performing integration testing are:

- Interface integrity
- Functional validity
- Performance

The main objectives of the testing our system are:

- To test whether the system can read and process the query video.
- Check whether the similarity search strategy is performing well or not.
- The retrieved videos are playing are not.
- The precision rate of the resultant retrieved videos.

### 7.1. Test Cases

A collection of 40 AVI videos has been taken, which are encoded in XVID codec. The tests are conducted with 10 query AVI video clips, in which some are of other codec. Following are the test cases and their corresponding results, in which several types of tests are performed. In some test cases, without query video being selected, the search is performed.

- Precision rate can be calculated by the following formula:

$$PR = \frac{\text{Total matching videos} - \text{No. of irrelevant videos}}{\text{Total matching videos}}$$

### 7.2. Experimental Results

Video Name	Total matching videos	Relevant videos	Irrelevant videos	Precision rate
Blackberry	3	3	0	1
Football	6	4	2	0.66
Donald duck	4	3	1	0.75
Nokia N8 advertisement	5	4	1	0.5
News	8	6	2	0.75
<b>Overall Precision</b>				<b>0.732</b>

Table 1. Experimental Results for the 40AVI Videos.

### 8. CONCLUSION

With the extensive amount of information available on the internet and the same file being replicated across different locations, a similarity measure of this kind is very essential to identify such similar contents, especially with video files. This paper is an attempt to perform this similarity search based on video signatures. The power of this method lies in the underlying fact that, it enables us to identify such similar contents irrespective of the file format, compression techniques used, algorithms behind file creation and file attributes.

Due to some constraints, this application can be implemented only to AVI files. But it is highly scalable i.e., extended further in different aspects. The following are the possible future works that can be carried out, with this paper as a start:

- Current implementation performs search at desktop level. The same can be extended to the LAN or internet as well. This is because, unlike CBIR which requires the complete video file to be transferred to the processing system, this requires just a matrix representation of the video to be stored for processing. Once the processing is complete, the matrix can be cleared by storing the video information on the processing system in files and logs for future reference.

The system can be extended its functionality to semantic derivatives also, which can retrieve best results.

### 9. REFERENCES

- [1] Jawahar.C.V., Chennupati.B., Paluri.B., & Jammalamadaka.N. (2005, December). "Video Retrieval Based on Textual Queries". *Proceedings of the Thirteenth International Conference on Advanced Computing and Communications*.
- [2] T.N.Shanmugam, & Rajendran, P. (2009). "An Enhanced Content-based Video Retrieval System based on Query Clip". *International Journal of Research and Reviews in Applied Sciences, Volume 1, Issue 3*.
- [3] S, C., & Zakhor.A. (2003). "Fast similarity search on video signatures". *IEEE International Conference on Image Processing, Vol.3*.
- [4] Sen-Cheung.S, & Zakhor, A. (2003). "Efficient video similarity measurement with video signature". *IEEE Transactions on Circuits and Systems for Video Technology, Vol.13*.
- [5] Cao, Z., & Zhu, M. (2009). "An efficient video similarity search strategy for video-on-demand systems". *2nd IEEE International Conference on Broadband Network & Multimedia Technology, IC-BNMT '09*.
- [6] P. Indyk and R. Motwani. Approximate Nearest Neighbor - Towards Removing the Curse of Dimensionality. In Proceedings of the 30th Symposium on Theory of Computing, 1998, pp. 604-613.
- [7] Yu-Hsuan Ho, Chia-Wen Lin, Jing-Fung Chen et al. Fast Coarse-to-Fine Video Retrieval Using Shot-Level Spatio-Temporal Statistics. *IEEE Trans. on Circuits and Systems for Video Technology*, 2006 vol. 16, pp.642-648
- [8] H. T. Shen, X. Zhou, Z. Huang, and J. Shao. Statistical summarization of content features for fast near-duplicate video detection. *Proc. of ACM int'l conf. on Multimedia*, 2007 pp.164–165
- [9] Ma Y-F, Zhang H-J Motion Texture: A New Motion based Video Representation *Proc. Of Int'l Conf. on Pattern Recognition*, 2002 vol.2 pp.548-551
- [10] Yan Ke Rahul Sukthankar Larry Huston An efficient parts-based near-duplicate and subimage retrieval system *Proc. of the 12<sup>th</sup> annual ACM int'l conf. on Multimedia* 2004 pp.869-876

- [11] Yang X, Tian Qi, Chang E-C A color fingerprint of video shot for content identificationin *Proc. of the 12th annual ACM int'l conf. on Multimedia* 2004 pp.276-279
- [12] Naturel X, Gros P Detecting repeats for video structuring *Multimedia Tools and Applications* 2008 vol.38 pp.233-252
- [13] Radhakrishnan R, Bauer C Content-based Video Signatures based on Projections of Difference Images
- Proc. of IEEE 9th Workshop on Multimedia Signal Processing*, 2007 pp.341-344
- [14] Gionis A, Indyk P, Motwani R Similarity search in high dimensions via hashing. *Proc. of the 25th Int. Conf. on Very Large Data Bases* 1999 pp. 518–529