



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

(Open Access, Double Blind Peer-reviewed, Refereed and Indexed Journal)

www.iasir.net

Distributed Content based Video Search Engine using Hadoop Framework

Aakash Agarwal¹, Dhananjay Uttarwar², Riyaz Kadiwar³, Vijay Katkar⁴

Department of Information Technology
Pimpri Chinchwad College of Engineering
Sector No. 26, Nigdi, Pradikaran, Pune
India

Abstract: The exponential increase in use of internet has also accelerated the generation of accessible multimedia content. So an effective approach is required for finding the relatively similar videos from a large volume of videos. Content Based Video Retrieval (CBVR) is an ongoing research area which aims to retrieve similar videos from huge databases in response to the query. Key frame identification from videos plays an important role in CBVR. In this paper, a distributed video search engine framework is proposed comprising key frame extraction, feature extraction and comparison to retrieve videos on Hadoop framework. Sequence files for key frames are used in the framework for input/output purpose. Experimentation and evaluation of system is carried out using various feature extraction algorithms.

Keywords: Content-based Video Retrieval (CBVR); Hadoop; MapReduce; Sequence file;

I. Introduction

With increase in bandwidth and storage on internet in recent years, amount of data generated in the form of text files, images, videos, etc. has increased tremendously. Finding the desired images or videos via text input is tedious. Rather using a video or an image as a query will certainly increase the output accuracy of the system. This technique is referred as Content Based Retrieval. Content Based Video Retrieval is a system consisting different techniques used to process video.

Key frames extraction from videos and feature extraction and comparison plays a vital role in Content Based Video Retrieval. Sequence files for key frames are used in the proposed system. The main purpose of creating a sequence file is to store a number of small files into one large file which reduces the extra load for transferring small file in Hadoop framework for MapReduce tasks. The sequence files formed in this system has key frames of video in key-value pair format in which the key is the video's name and the value is the feature vector of that key frame.

In this paper, a cluster running Hadoop MapReduce framework with feature extraction and comparison algorithms is used [1]. Video Search Engine signifies the video retrieval based on the key-frame [2] (image) of the video. In the proposed framework, a video as a query approach is used. Algorithms like Block Truncation Coding, Fuzzy Edge Detection and Local Binary pattern are used for feature extraction and comparison [3][4][5].

Hadoop is a highly fault tolerant distributed computing framework that allows processing of large data sets across clusters of computers using simple programming models and commodity hardware. The Hadoop framework has different basic components working in it like Hadoop Distributed File System, MapReduce programming model, Hadoop Yarn, etc. The job submitted to Hadoop is executed by dividing it into map task and reduce task. These MapReduce tasks are provided with different splits of the input data to be processed. The MapReduce is executed in three phases: Map phase, Sort/Shuffle phase and Reducer phase.

This paper is organized into 5 Sections. Section II describes the related works followed by section III which describes the system architecture and flow of framework. Section IV shows the algorithms which can be used for image feature extraction while section V shows the experimental results for the algorithms supporting the system. Section VI concludes with references.

II. Related Works

Jenq-ShiouLeu, Yun-Sun Yee and Wa-Lin Chen in [1] compared the performance of Map-Reduce on Hadoop and SQL server. They analyzed that Hadoop can save more time for query than SQL server. They also computed on how the number of nodes, number of Maps and Reduces which the maps and reduces run on can affect the system. Sudeep D. Thepade and Ashvini A. Tonge in paper [2], it is shown that key-frames play a vital role in process of Content-Based Video Retrieval (CBVR). Key-frames reduces the space required and also improve the retrieval speed.

Bharadi and Kekre in [3] proposed Content Based Image Retrieval which was based on two methods namely Modified Block Truncation Coding and Kekre Transform Pattern. From the above two methods, MBTC has been used to extract the color information while the Kekre pattern was used to extract the texture information. Wafa

barkhoda; Fardin Akhlaqian Tab; Om-Kolsoom Shahryari in paper [4], presented a fuzzy edge detection algorithm is used to find the edge in the image. Standard deviation and image gradient techniques are used.

A.Bhagyalakshmi and V.VijayaChamundeeswarii in [5] presented, images were retrieved from database of images using Local Binary Pattern (LBP) Histogram and statistical features. Hadoop framework is presented in [6] to perform parallel processing for CBIR use. Distribution of image and image data was done over a large number of nodes. Techniques used are images retrieval and indexing, indexing parallel processing of downloading images and comparing the features of images which were retrieved from different sources.

Anucha Tungkasthan and Wichian Premchaiswadi in [7] presented Hadoop Map-Reduce framework which was used with the intention to increase the performance of two functions which were query processing and data insertion. The proposed framework was used to perform distributed processing which was used for CBIR system.

Kumar in [8] proposed a Map-Reduce Framework for 3D Content Based Video Retrieval application with key frame extraction, feature extraction and matching. Threshold Based – Predictive Clustering Tree (TB - PCT) was used for the same. Dr.Sudeep.D.Thepade, Ankur.A.Mali and Krishnasagar .S. Subhedarpape in paper [9] presented Thepade's Ternary Block Truncation Coding (TTBTC) and Thepade's Sorted Ternary Block Truncation Coding (TSTBTC) are used for color content extraction of videos for CBVR. Thepade's ternary BTC has more bands than binary BTC so finer details of video pixels are extracted.

Dijana Tralic, Sonja Grgic, and Branka Zovko-Cihlar in Paper [10] presents a idea of using Cellular Automata and Local Binary Patterns for texture description. Here frame is divided into overlapping blocks and then Cellular Automata is used to learn a set of rules for every block in frame. Pixel values are first transformed to binary values using LBP

Yong Jin, Ligang Liu and Qingbiao Wu in paper [11] presented, an efficient approach for video abstraction and navigation by extracting the key-frames is used. Video is represented as sequence points in a feature space that forms polyline curve. In [12] SongqingDuan, Bin Wu, Bai Wang and Juan Yang presents, using Hadoop's MapReduce framework statistical algorithm was designed and implemented which was suitable for large-scale data processing which obtains more than ten statistical features.

III. System Architecture

The presented architecture of the system in this paper has been divided into two parts: A. Formation of database for video retrieval as in Fig.1, B. Searching for relevant videos by video as a query as in Fig. 2. These are elaborated as follows:

A. Formation of database for video retrieval

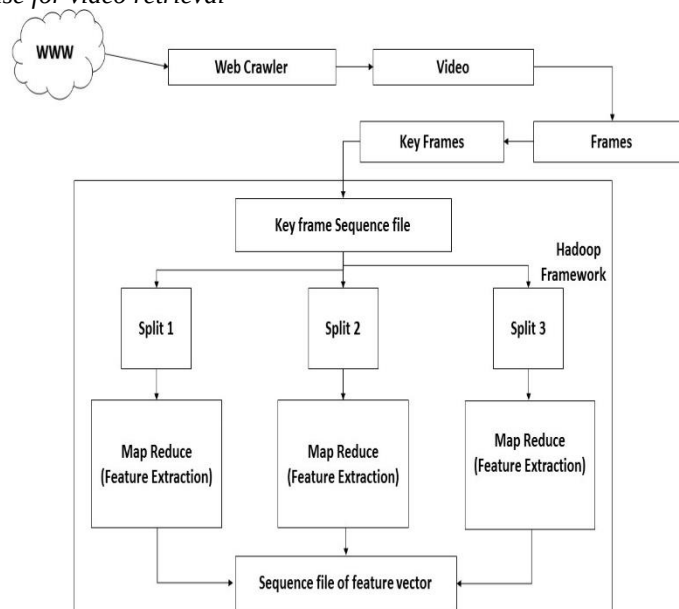


Fig.1. Formation of database for video retrieval

As shown in Fig. 1, this part creates a video database which is used by the next part.

Step 1: Web crawler is used for downloading the videos by parsing number of video links which are available on World Wide Web (WWW). The downloaded videos creates the dataset for the system.

Step 2: Extracting the frames of videos, then identifying key-frames from frames and converting them into key-frame sequence file.

Step 3: The created key-frame sequence file is split among the nodes of the Hadoop cluster.

Step 4: At each node, feature extraction algorithms are executed for retrieval of features by recreating each key-frame from the key-frame sequence file. The extracted features are written in a feature sequence file. Thus, at the end of this subtask, sequence files of feature vectors are formed.

B. Searching for relevant videos by video as a query:

After the formation of sufficient database, this part of the system aims to produce relevant results to the user query. Fig. 2, shows the architecture.

The keyframe feature sequence files formed from the first part of framework are used by this part of system. The query to the system i.e. a video is processed to extract keyframes. Feature extraction algorithms are applied to key frames to get the query feature vector. The query feature vector of such individual key frames of the query is then passed to the MapReduce tasks over the Hadoop framework.

The MapReduce tasks in this phase of system do the job of feature comparison between the query key frame features and features from the database. The feature comparison is done by calculating the Euclidean distance between the query key frame feature vector and the individual feature vectors of key frame read from the feature sequence file. Since the MapReduce uses (key, value) pair, here the video identifier acts as key while the feature vector acts as value which undergoes the process of comparison.

These results are then reproduced to single container file which is further passed as an input to the next MapReduce task. This MapReduce task is responsible to run the process of ranking the results so that videos with highest relevance are shown to the users first. The results can then be redirected to the user interface for the input user query.

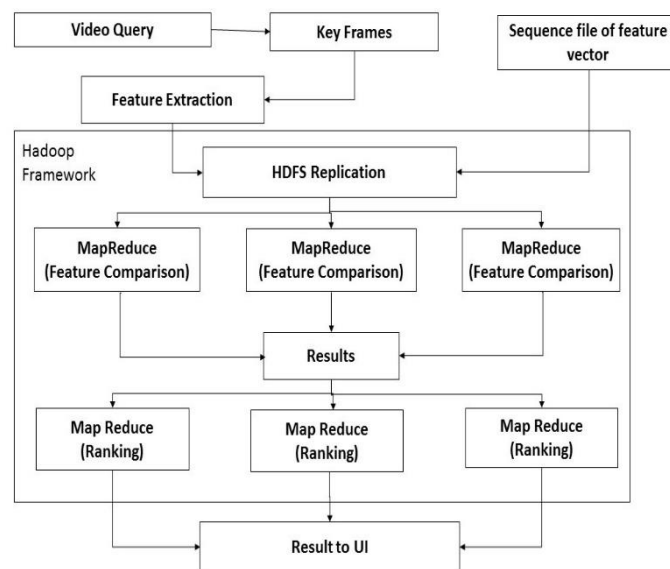


Fig. 2. Searching based on query

IV. Feature Extraction Algorithms

Feature Extraction is the process of extracting various features like color, shape and texture from an image i.e. key-frame of the query video. As features define the behavior of an image, they show its place in terms of storage taken, efficiency in classification and also in time consumption. The main goal of feature extraction algorithm is to obtain the most relevant information from the original image and represent that feature i.e. information in a lower dimensionality space. In this paper, different image feature extraction algorithms are experimented to find their suitability in the framework as in Fig. 1.

The query video is firstly converted into frames, then from frames key-frames are identified. A sequence file is created of the identified key-frames which is the input to the feature extraction algorithms. From the sequence file, each time a key-frame is extracted and features of that key-frame is calculated.

1. Block Truncation Coding:

Block Truncation Coding (BTC) is a simple image coding technique [4]. This algorithm first resizes the image into fixed width and height. Then it divides the key-frame into small non-overlapping image blocks. Each time one small block is coded. The method first computes the mean pixel value of each block and then each pixel is compared to the block mean. If a pixel is greater than or equal to the block mean, the corresponding pixel position of the bitmap will have a value of 1; otherwise value is 0. Since the image consists of three independent grey scale planes, the best way to view a color image is by applying BTC to each color plane independently. Mostly all color images are recorded in RGB space, which is the most well-known color space [3].

Algorithm:

First resize the image in 256*256 pixels.

Now let $X=\{r(i,j),g(i,j),b(i,j) \mid i=1,2,\dots,m, j=1,2,\dots,n\}$ be an $m \times n$ color block in RGB space where $m=n=256$.

$$i_b = \frac{1}{3}r(i,j) + g(i,j) + b(i,j) \quad (1)$$

The threshold is computed as the mean of each colour,

$$T1 = \frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m r(i,j) \quad (2)$$

$$T2 = \frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m g(i,j) \quad (3)$$

$$T3 = \frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m b(i,j) \quad (4)$$

The binary bitmap is computed as below,

$$bm_1(i,j) = \begin{cases} 1 & \text{if } r(i,j) \geq T_1 \\ 0 & \text{if } r(i,j) < T_1 \end{cases} \quad (5)$$

$$bm_2(i,j) = \begin{cases} 1 & \text{if } g(i,j) \geq T_1 \\ 0 & \text{if } g(i,j) < T_2 \end{cases} \quad (6)$$

$$bm_3(i,j) = \begin{cases} 1 & \text{if } b(i,j) \geq T_1 \\ 0 & \text{if } b(i,j) < T_2 \end{cases} \quad (7)$$

After the creation of the bitmap, two representative (mean) colors are then computed. The two mean colors are $MC1 = \{Cr1, Cg1, Cb1\}$ and $MC2 = \{Cr2, Cg2, Cb2\}$. Where upper-mean is calculated by,

$$C_{r1} = \frac{1}{\sum_{i=1}^n \sum_{j=1}^m bm_1(i,j)} \sum_{i=1}^n \sum_{j=1}^m bm_1(i,j)r(i,j) \quad (8)$$

$$C_{g1} = \frac{1}{\sum_{i=1}^n \sum_{j=1}^m bm_2(i,j)} \sum_{i=1}^n \sum_{j=1}^m bm_2(i,j)g(i,j) \quad (9)$$

$$C_{b1} = \frac{1}{\sum_{i=1}^n \sum_{j=1}^m bm_3(i,j)} \sum_{i=1}^n \sum_{j=1}^m bm_3(i,j)b(i,j) \quad (10)$$

And lower-mean is calculated as follow

$$C_{r2} = \frac{1}{n \times m - \sum_{i=1}^n \sum_{j=1}^m bm_1(i,j)} \sum_{i=1}^n \sum_{j=1}^m (1 - bm_1(i,j))r(i,j) \quad (11)$$

$$C_{g2} = \frac{1}{n \times m - \sum_{i=1}^n \sum_{j=1}^m bm_2(i,j)} \sum_{i=1}^n \sum_{j=1}^m (1 - bm_2(i,j))g(i,j) \quad (12)$$

$$C_{b2} = \frac{1}{n \times m - \sum_{i=1}^n \sum_{j=1}^m bm_3(i,j)} \sum_{i=1}^n \sum_{j=1}^m (1 - bm_3(i,j))b(i,j) \quad (13)$$

2. Fuzzy Edge Detection

Fuzzy edge detection is an algorithm used to detect the edges present in the image. Standard deviation and image gradient (sobel operator) are used for identifying edges. Firstly, the standard deviation of a pixel is computed with respect to its neighboring eight pixel values. Further sobel operator is used to calculate the image gradient of the pixel.

P ₁	P ₂	P ₃
P ₄	P ₅	P ₆
P ₇	P ₈	P ₉

Fig.3. Mask to compute standard deviation

Fuzzy rule is applied by taking the threshold value of standard deviation and image gradient to find whether the pixel is part of edge or not. If the pixel value is more than threshold, its value is changed to 255 and is considered an edge candidate otherwise the value is changed to 0. Thus, such pixels together form a fuzzy image.[4]

3. Local Binary Patterns

Local Binary Pattern (LBP) is an image descriptor technique in which extract the features on texture spectrum of an image [6]. Integrating the results from LBP into a histogram formation enhances the accuracy in results for classification and comparison. To apply LBP, at first, the color image is converted to gray-scale image. Then each pixel of the gray-scale image is compared with each of eight neighbouring pixels in a particular fashion i.e, clockwise or anticlockwise. If the value of center pixel is greater than its neighboring pixel then '0' is written else '1' is written. Thus, for each pixel, a binary pattern is formed of length 8. Now, based on the occurrence of each number from these patterns, a histogram is formed which can also be seen as 256 dimensional feature vector. This feature vector then can be used for comparison of two images based on distance obtained from the Euclidean distance formula.

V. Results

In this paper, Experimentation are performed on intel i5 5th gen processor having 8GB RAM using Java and opencv library for feature extraction algorithms for the search engine framework. For the evaluation of performance of algorithms, precision and recall are used. Videos of shinchan, tennis, kapil sharma show, news and cricket are downloaded from internet and used for experimentation.

$$\text{Precision} = \frac{\text{no. of relevant images retrieved}}{\text{Total no. of images retrieved}} \quad (14)$$

$$\text{Recall} = \frac{\text{no. of relevant images retrieved}}{\text{Total no. of relevant images in the database}} \quad (15)$$

It can be observed from Fig. 4 that crossover of 45.25% is obtained using BTC.

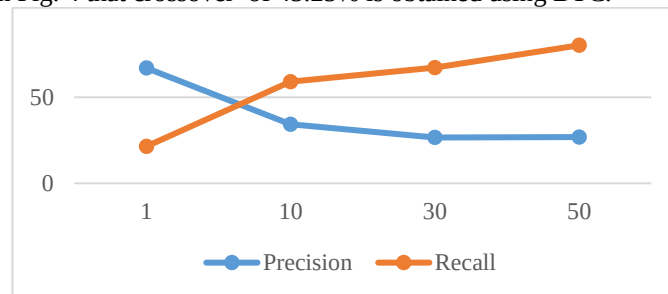


Fig. 4: Experimental results using BTC

It is shown from Fig. 5 that crossover of 42.05% is obtained by using fuzzy edge detection.

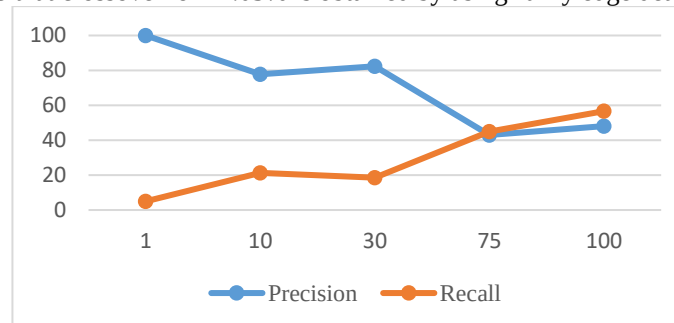


Fig. 5. Experimental results using fuzzy algorithm

It is shown from Fig. 6 that crossover of 52.05% is obtained by using Local Binary Pattern algorithm

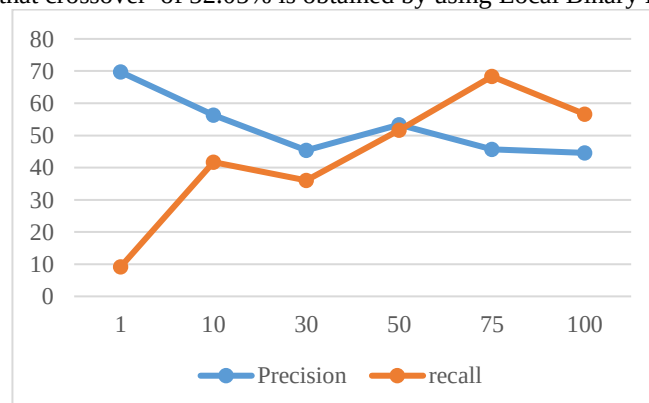


Fig.6. Experimental results using Local Binary Pattern algorithm

VI. Conclusion

A distributed framework for design of content based video search engine using Hadoop MapReduce framework is presented in this paper. It is observed that the relevance of results varies depending on the threshold provided to different algorithms like Block Truncation Coding, Fuzzy edge detection and Local Binary Patterns. The experimentation results show that highest crossover point (52.05%) is achieved using Local binary pattern for threshold distance 1200.

VII. References

- [1] Jenq-ShiouLeu, Yun-Sun Yee and Wa-Lin Chen, "Comparison of Map-Reduce and SQL on Large-scaleData Processing", International Symposium on Parallel and Distributed Processing with Applications
- [2] Sudeep D. Thepade and Ashvini A. Tonge, "An Optimized Key Frame Extraction for Detection of Near Duplicates in Content BasedVideo Retrieval", International Conference on Communication and Signal Processing, April 3-5,2014, India
- [3] H B Kekre and V A Bharadi, "Fusion of Modified Block Truncation Coding and Kekre Transform Patterns for Colour& Texture Based CBIR", 2012 International Conference on Communication, Information & Computing Technology (ICCICT)
- [4] Wafa barkhoda; Fardin Akhlaqian Tab; Om-Kolsoom Shahryari
"Fuzzy Edge Detection Based on Pixel's Gradient and Standard Deviation values" 2009 IEEE
- [5] A.Bhagyalakshmi and V.VijayaChamundeeswari, "Image retrieval using Color and Texture Binary Patterns", 2015 International Conference on Green Computing and Internet of Things (ICGCIoT)
- [6] WichianPremchaiswadi, Anucha Tungkatsathan, and SarayutIntarasema, "Improving Performance of Content-Based ImageRetrieval Schemes using Hadoop MapReduce", 2013 IEEE
- [7] Anucha Tungkasthan and WichianPremchaiswadi, "A Parallel Processing Framework using MapReducefor Content-Based Image Retrieval", 2013 Eleventh International Conference on ICT and Knowledge Engineering
- [8] C. Ranjith Kumar and Dr. S. Suguna, "3D CBVR: AN EFFICIENT VIDEO RETRIEVAL USING MAPREDUCE FRAMEWORK WITH TB-PCT", 2016 International Conference on Inventive Computation Technologies (ICICT)
- [9] Dr.Sudeep.D.Thepade, Ankur.A.Mali and Krishnasagar.S.Subhedarpage, "Content Based Video Retrieval using Thepade's Ternary Block Truncation Coding and Thepade's Sorted Ternary Block Truncation Coding with various Color Spaces", International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS).
- [10] Dijana Tralic, Sonja Grgic, and Branka Zovko-Cihlar, "Video Frame Copy-Move Forgery Detection Based on Cellular Automata and Local Binary Patterns", 2014 X International Symposium on Telecommunications (BIHTEL)
- [11] Yong Jin, Ligang Liu and Qingbiao Wu, "iFrames: A Multi-Level Keyframe Extraction andNavigation Tool for Videos", 2011 12th International Conference on Computer-Aided Design and Computer Graphics
- [12] SongqingDuan, Bin Wu, Bai Wang and Juan Yang, "Design and Implementation of Parallel Statistical Algorithm based on Hadoop's MapReduce Model", Proceedings of IEEE CCIS2011
- [13] Sridhar Vemula, Christopher Crick, "Hadoop Image Processing Framework", Big Data (BigData Congress), 2015 IEEE International Congress
- [14] S. D. Ruikar and Rohit S. Kabade, "Content Based Image Retrieval by Combining FeatureVector", IEEE WiSPNET 2016 conference
- [15] VishakhaSoni, Sandeep Kumar Mathariya and RanuSoni, "User Friendly Approach for Video Search Technique Using Text and Image as Query", 2014 Conference on IT in Business, Industry and Government (CSIBIG)
- [16] Akash K Sabarad, Mohamed HumairKankudti, Meena S M, and Moula Husain, "Color and Texture Feature Extraction using Apache Hadoop Framework", 2015 International Conference on Computing Communication Control and Automation