International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 12, December 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

CONTENT BASED IMAGE RETRIEVAL USING HADOOP AND HIPI

B.R.Kavitha[1], P.Kumaresan[2] , Ramya Govindaraj[3]

[1,2,3]Assistant Professor(Senior),SITE,

VIT University, Vellore, TamilNadu, (India)

ABSTRACT

In real world scenarios storing the required data and retrieving is becoming a hectic task by the growing size of data. Hadoop is an open source software framework that offers solution for this problem and here efficient retrieval of images by similarity search is done, the input being the query image, using Hadoop framework and HIPI interface. Here Hadoop HDFS is used to store the images which are used as input to the popular MapReduce programming model for processing the images. As image content cannot be given directly as an input to MapReduce, HIPI API is used for storing images in hib (Hadoop image bundle) format and processed through MapReduce. Similar images are retrieved by k-nearest neighbour approach. The image based similarity searching can be used significantly in the medical field for diagnosis of diseases in order to provide effective treatment.

Keywords: Big data, HIPI, Hadoop, Image Retrieval

I.INTRODUCTION

Data has become ubiquitous in the past few years. The data has grown greater than ever than before and scale at which data grown is constantly increasing. According to the Digital Universe study of International Data Corporation (IDC), 2011 the data which is produced all over the world is beyond 1.8 Zettabytes, which signifies an exponential growth of factor nine from the past five years. This huge data has been the reason for enormous research on how to handle the data effectively: the systems which exist are not capable of storing, processing and retrieving such huge data in the software or hardware perspective. In this context, the 'Bigdata' is the term that is used to signify the explosion of data. Big data is nothing but the rapid growth of data generated from different kinds of data sources. This growth can be in terms of volume or speed of the dataflow in and out of data management systems. It can be the huge data which is in different formats of structured and unstructured data. Especially multimedia data plays a major role in Bigdata. The main reason of huge growth of multimedia data is through social networking sites. In 2012, Facebook has reported that nearly 300 million images are uploaded every day, which is of 7 petabytes of data every month. However, in the context of Bigdata the scalability is still an issue. Infrastructures such as clusters, Grids, Clouds(Nimbus [5], Amazon Elastic Compute Cloud [6]), multi-core servers with huge memory is a major step forward, but still have their limitations when accommodating Big Data.

CBIR Technology

Content Based Image Retrieval is a technique that uses the image visual contents for retrieving the images on large image datasets and it has been a good research area from the last decade. A Content Based Image Retrieval system is a field where lot of companies are working and developing which are known as CBIR systems. The bottlenecks for CBIR systems are high computation tasks which are caused by complexity in computing and large amount of data for storing, indexing etc. Therefore the intention for every new CBIR system proposed is to overcome the common limitations of Hadoop to explore the solution for the problem. For parallel processing Google introduced MapReduce model which is used in this paper for retrieving the results faster.

Hadoop MapReduce Framework

The Hadoop Mapreduce framework is proposed to implement the analysis on images while extracting the features of the image which are the main representatives of the image as a whole and while retrieving images which are similar to the features which are measured at the time of feature extraction by Map reduce on whole dataset and on queried image. MapReduce [3] is the most efficient and popular tool that enables the processing of huge amounts of information using commodity machines. In content based image retrieval field, several million images can be handled by the systems [11], [12], billions of descriptors [13], [14], or address web-scale problems [15], [16] and the Hadoop based ImageTerrier platform[17].To support this to have fast access to huge database requires a good computing model which is Hadoop framework model for effective distributed computing model. There are several MapReduce implementations like Apache's Hadoop [7], Twister [8], Disco [9] on top of which specialized higher level frameworks were developed. Computer vision-wise, the Mahout [10] project uses Hadoop to provide an extensive list of machine learning algorithms. A typical CBIR system can be likely decomposed in three steps: firstly, the characteristic features for each image in the database are extracted and are used to index images, secondly, the features vector of a query image is computed; and thirdly, the features vector of the query image is compared to those of each image in the database. The Map Reduce programming model has been implemented by the open-source community through the Hadoop project [4].

HIPI

HIPI is a library for the Apache Hadoop programming framework that provides an API for performing image processing tasks in a distributed computing environment. [19] The input format used in HIPI is a HipiImageBundle(HIB). A HIB is set of images which are combined to format single large file along with some metadata describing the layout of images. The Hadoop Distributed File System (HDFS) [18] was built to provide storage for huge files with streaming data access patterns for running on clusters of commodity hardware. A HIB is created by the images already present on HDFS or a remote source. To improve the efficiency, HIPI has a culling stage to conventionally Mapreduce workflow and discards images that do not meet the criteria based on metadata of the image. Culling stage avoids the decompressing of image full pixel buffer into the main memory. The CullMapper can be specified by the user according to their requirements. The images are represented as FloatImage object associated with ImageHeader object. Even though HIPI does not modify any Map reduce properties of job scheduling, the parameters which the image processing is done varies

according to user which are setup by the HipiJobobject during the setup. While implementing the key steps of Hadoop MapReduce workflow care should be taken for efficient workflow.

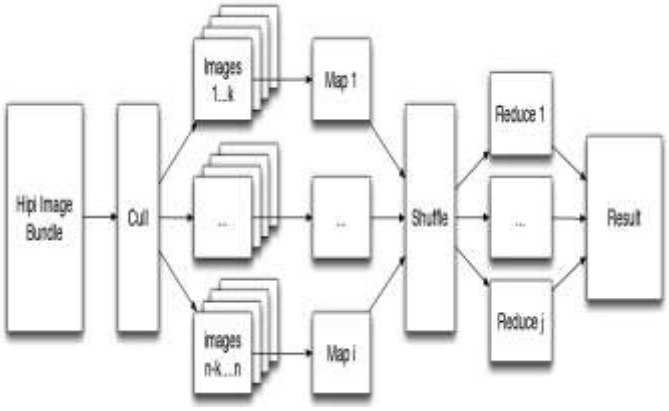

Figure 1 HIPI Image processing[19]

II. METHODS

Every CBIR system will have three stages for obtaining the final outcome. First stage is that the images should be stored and the features of the images are to be extracted (offline processing). Second stage is the queried image features are to be extracted and similarity search algorithm is implemented on the queried image and all other images in the repository. Third stage is retrieving the images in the order by which it matches the properties of queried image.

In this section the three modules of the proposed CBIR system is clearly explained.

Module-I

Hadoop framework is used to handle the huge data .So the image dataset is stored in the Hadoop HDFS in the form of Hadoop image bundle using HIPI. This image bundle is divided into many bundles according to the size of the dataset as we

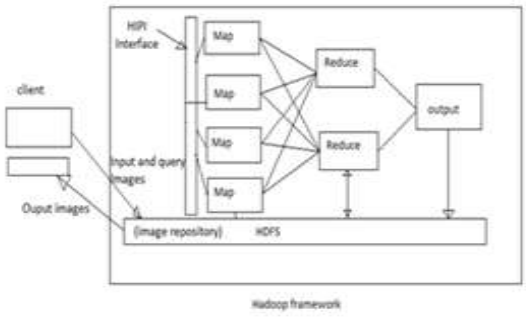

Figure 2 Architecture of proposed CBIR system

know the data stored in the HDFS is stored in the form of blocks of each 64Mb.Now the image bundles are given as input to the MapReduce processing i.e for feature extraction. The output of the MapReduce is also stored in the HDFS. By this the offline processing (stage-I) of CBIR is accomplished.

Module-II

Now the image which is queried from the client is converted in to .hib format and same is given to the mapreduce for feature extraction i.e. online processing. After obtaining both offline and online feature extraction now the image similarity search between the queried image and rest of the images in the dataset is being done using hexcode generated using SHA-1 for all images. So in the text file the input text file of feature extraction is compared as strings.

Module-III

In this stage the images which are similar as per the features extracted earlier are retrieved accordingly by the descending order of similarity. The images are present in the hib format to process but the output must be image in jpg or jpeg format, so the final images which are similar are converted in to the bytearray format such that when this is sent back to the client such that the output is visible in jpg or jpeg format to the end users.

By this analysis the similar images are computed in the background with mapreduce tasks and final results are obtained in the client machine in few seconds. In the single machine with large data set it takes minutes for offline processing and then it takes time comparatively more than distributed environment to do the processing of similarity search and retrieve the results. The scenario is being implemented in the distributed environment by creating a cluster of nodes and utilizing those resources as slave data nodes. This will be the optimized output scenario of the proposed system.

III.RESULTS AND DISCUSSION

After successful installation the namenode, datanode, tasktracker, jobtracker and secondary namenode works fine. The HIPI interface is downloaded for easy image processing while using mapreduce tasks. By using HIPI, the image dataset is converted into the Hadoop image bundle (.hib) format which is stored in the HDFS itself. After obtaining the images now mapreduce tasks are written for feature extraction by processing the images, this is also made simple by the HIPI API. The mapreduce program is written for extraction of colour, texture and shape of the images and stored in a binary format. By this the stage-I is completed successfully. This data is used for similarity search algorithm at last for retrieving the images in the descending order of similarity.

The image which is needed to query is also converted to .hib format. Then input images and query image is given as inputs to the feature extraction module which is a mapreduce program. In this module each image hashcode using SHA-1 algorithm is extracted in hexadecimal format let's say it is a hexcode of image. In the above figure input.hib is the Hadoop image bundle file of the all input images and input.hib.dat is metadata of the hib file. Similarly the input1.hib file is the Hadoop image bundle file of the query image.

The images are stored in the HDFS in .hib format is shown in Fig 3 below. Now both the files in hib format are given individually as input to feature extraction and the outputs are output1 as a text file for input images and output2 again as a text file for query image. For each image one line of key and value pair is generated as output. In this project input file consists of 1000 images so 1000 lines of output is generated.

International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 12, December 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

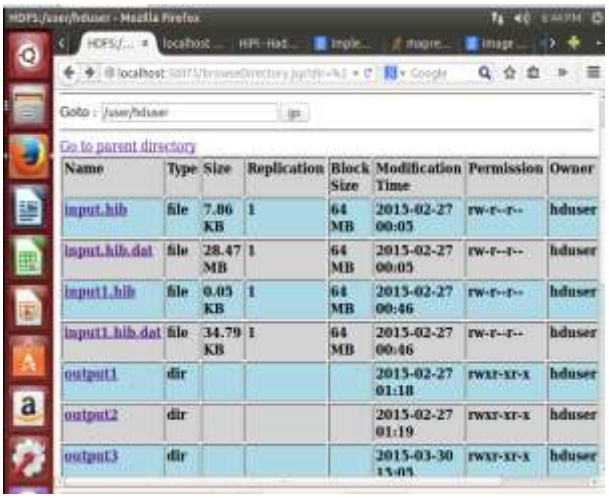

Figure 3 Images (.hib format)

As there is only one image to query, one line of output is generated. The output files output1 and output2 are as below (Fig 4 and Fig 5).

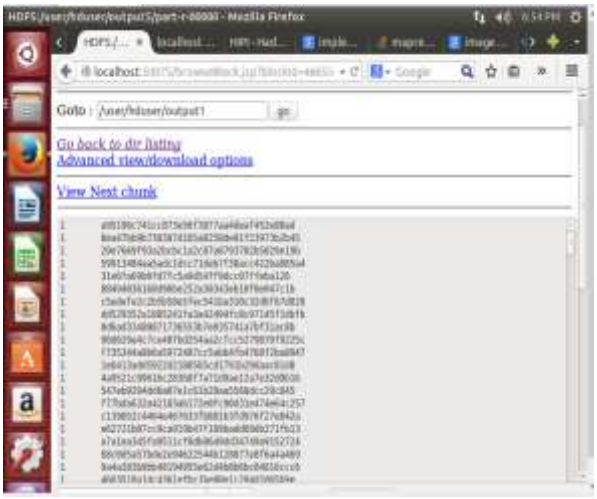

Figure 4 Output of input images

Both the text files are given as input for comparison, to check whether the queried image exists in the input image dataset or not. Every line of the text file of input image is compared with the string in the text file of queried image. If in case match found the queried image string is given as output in a text file**.** As the output of comparison is not null, the image which is duplicate is to be retrieved. So it is being retrieved from the input1.hib and the output is a jpg file and text file to convey the image is successfully generated.

The image present in jpg format in the file system is in byte array format this is not visible as the actual image when it is opened directly this image is sent to the local file system and opened with the image viewer then the final duplicate image is visible.

International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 12, December 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

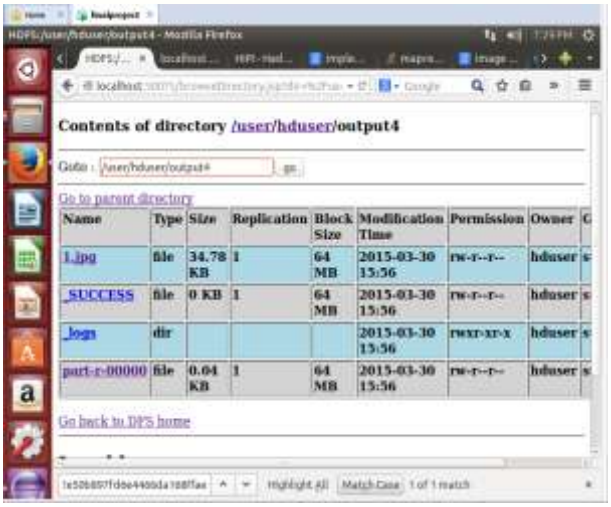

Figure 5 Final Output

This is the command used >>$ Hadoop fs –get /user/hduser/output8/1.jpg /home/hduser/Desktop/project. By this command the final output image is sent to the project folder on Desktop. The final output is the duplicated image clearly visible in jpg format or whichever format the input images are.

IV.CONCLUSION

In this paper, Hadoop distributed computing environment to content based image retrieval is used. For that, a method is proposed to characterize the numerical content of input images: the method is to extract the hex code from the images and do the similarity search on a string comparison basis, and utilize MapReduce computing model to improve the performance of image retrieval among massive image data. Furthermore, image retrieval based on MapReduce distributed computing model are more efficient when target image data is large. The method just consists of small data of hexcode generated which is unique and applying similarity search on unique data to get accurate results. This way the outcome of this work can be used in scenarios in medical field where to find out the exact cases of patient problem which is undertaken before and in few crime scenarios like thumb prints.

REFERENCES

[1] Diana Moise**,** Denis shestakov, Gylfi Gudmundsson and Laurent Amsaleg. Terabyte-scale image similarity search: experience and best practice. *IEEE* .978-1-4799-1293-3/13/2013

[2] Chunhao GU and Yang GAO.A Content-based Image Retrieval System Based on Hadoop and Lucene. 978-0-7695-4864-7/12/2012 .*IEEE* DOI 10.1109/CGC.2012.33

[3] J. Dean and S. Ghemawat. MapReduce: simplified data processing on large clusters. *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[4] The Apache Hadoop Project. *http://www.hadoop.org*.

[5] K. Keahey and T. Freeman. Science Clouds: Early experiences in Cloud computing for scientific applications. *Cloud Computing and Its Applications CCA*, 2008.

[6] D. Robinson. Amazon Web Services Made Simple: Learn how Amazon EC2, S3, SimpleDB and SQS Web Services enables you to reach business goals faster. *Emereo Publishing*, 2008.

[7] The Hadoop MapReduce Framework. *http://hadoop.apache. org/mapreduce/*.

[8] Twister - Iterative MapReduce. *http://www. iterativeMapReduce.org/*.

[9] P.Mundkur, V. Tuulos, and J. Flatow. Disco: a computing platform for large-scale data analytics. *Proc. of the 10th ACM SIGPLAN workshop on Erlang*, 2011, pp. 84–89.

[10] S.Owen, R. Anil, T. Dunning, and E. Friedman. Mahout in Action. *Manning Publications*, 2011.

[11] H.Lejsek, F. H. Amundsson, B. T. J´onsson, and L. Amsaleg. NV-Tree: An efficient disk-based index for approximate search in very large high-dimensional collections. *IEEE Trans. Pattern Analysis and Machine Intelligence PAMI*, vol. 31, pp. 869–883, 2009.

[12] H.J´egou, F. Perronnin, M. Douze, J. S´anchez, P. P´erez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Trans. Pattern Analysis and Machine Intelligence PAMI*, vol. 34, no. 9, pp. 1704–1716, 2012.

[13] H. Lejsek, B. T. J´onsson, and L. Amsaleg. NV-Tree: nearest neighbors at the billion scale. *ACM Proc. International Conference on Multimedia Retrieval ICMR*, 2011, pp. 54:1– 54:8.

[14] H.Jegou, R. Tavenard, M. Douze, and L. Amsaleg. Searching in one billion vectors: Re-rank with source coding.*IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 2011, pp. 861–864.

[15] M.Douze, H. J´egou, H. Sandhawalia, L. Amsaleg, and C. Schmid.Evaluation of GIST descriptors for web-scale image search. *ACM Proc. International Conference on Image and Video Retrieval CIVR*, 2009, pp. 19:1–19:8.

[16] M.Batko, F. Falchi, C. Lucchese, D. Novak, R. Perego, F. Rabitti, J. Sedmidubsk´y, and P. Zezula, Building a web-scale image similarity search system. *Multimedia Tools Appl.*, vol. 47, no. 3, pp. 599–629, 2010.

[17] J. S. Hare, S. Samangooei, D. P. Dupplaw, and P. H. Lewis. ImageTerrier: an extensible platform for scalable highperformance image retrieval. *ACM Proc. International Conference on Multimedia Retrieval ICMR*, 2012, pp. 40:1– 40:8.

[18] K. Shvachko, H. Kuang, S. Radia, and R. Chansler. The Hadoop Distributed File System. *IEEE Proc. Mass Storage Systems and Technologies MSST*, 2010, pp. 1–10.

[19] Sweeney, Chris, et al. "HIPI: a Hadoop image processing interface for image-based mapreduce tasks." *Chris. University of Virginia* (2011).

[20] How to get started with HIPI interface, *http://hipi.cs.virginia.edu/*

[21] Wichian Premchaiswadi, Anucha Tungkatsathan and Sarayut Intarasema.Improving Performance of Content-Based Image Retrieval Schemes using Hadoop MapReduce.*IEEE*.978-1-4799-0838-7/13/,2013