International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 3 Issue: 11
ISSN: 2454-4248
532 – 536

Comparative Study of Various Image Processing Tools with BigData Images in Parallel Environment

Akriti Maheshwari[1] Yogesh Kumar Gupta[2]
[1]Research Scholar, Department of CS, Banasthali Vidyapith, Rajasthan, India
[2]Assistant Professor, Department of CS, Banasthali Vidyapith, Rajasthan, India

Abstract. BigData refers to the collection of various kinds of data in a large amount. This data needs to be process through some tools as it can have some useful information. In order to process this data, Hadoop came in existence. It processes the data in parallel distributed environment. BigData contains various forms of data such as text, images, videos etc. In this paper we are going to discuss about image data. We need to perform image processing to process these BigData images. As BigData is very huge so the number of images to be processed is also very large. Image processing can be performed in parallel environment in order to speed up this process. There are various tools to process the images in parallel environment, such as: HIPI, OpenCV, CUDA, MIPr etc. In this paper we have performed a comparative study on these tools.

Keywords: *BigData, Hadoop, Hadoop Distributed File System, MapReduce, Image Processing.*

*****

I. Introduction

A vast amount of data has been generated every day from sensors, social sites, satellites, cell phones etc. This continuously producing data is termed as BigData. It must be handled and analyzed because it can be beneficial in taking good decisions and making strategies for organizations. BigData is completely unstructured so it can't be handled by traditional databases. Handling of BigData requires special coding skills and knowledge. The highly unstructured nature of BigData is because different people use different ways and schemas to collect the BigData. Also, the way of representing this data differs according to the applications in which it is being represented. Complexity in handling the BigData also increases with the increases in its volume [1]. Currently, BigData is used in almost every field such as: Banking, Education, Health Care, Government, Manufacturing, Retails etc.

BigData has several characteristics. These characteristics are also known as V's of BigData. Here, we are presenting 7V's of BigData, which are: Volume, Velocity, Variety, Veracity, Validity, Volatility, and Value.

1. **Volume:** It refers to the huge amount of data generated from various heterogeneous sources such as sensors, web pages, social media etc. This kind of huge and unstructured data can't be deal with traditional approaches of database such as SQL.
2. **Velocity:** It refers to the speed of data generated every day. At present time, the speed of data generation is very high. It doesn't only refer to the speed of the data generation but also to the speed of the data flow and aggregation.
3. **Variety:** It can be defined as the type of the generated data. As the data doesn't arrive only from relational databases, it can also be generated from semi structured and unstructured data sources such as HTML web pages etc.
4. **Veracity:** Veracity is the meaningfulness of the data. We cannot just assume that all this data, coming from heterogeneous sources, is useful. There is noise, biasness, and abnormality in this data.
5. **Validity:** Validity is similar to the veracity of data, but actually these two are totally different concepts of BigData. Validity of the data refers to the accuracy and correctness of data with respect to certain application. It is possible that some data is valid for an application while invalid for another.
6. **Volatility:** Volatility refers to the time for which the data is meaningful for us and must be kept for future use. Volatility of BigData can be understood with the help of retention policy which is used in structured data. In case of BigData, retention period can increase and it becomes costly to implement.

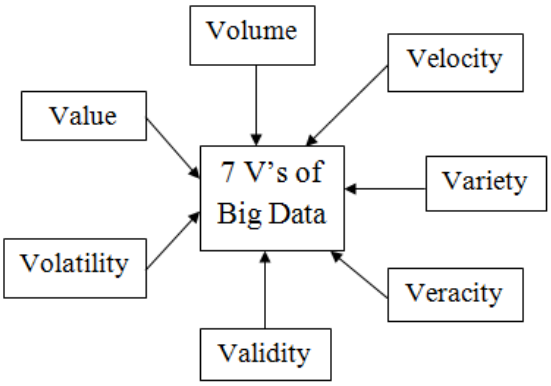

Fig. 1.7Vs of BigData

7. **Value:** Value is the most important characteristic of BigData. This V of BigData is generally discussed with respect to the outcome of BigData processing. We always try to extract maximum value from the

 ISSN: 2454-4248
532 – 536

data being analysed. Value of data also helps in making good decisions [2, 3].

BigData contains various types of data such as text files, images, audio, video, and other multimedia contents. To process these types of data, we need various tools. In this paper, we are discussing only image data. Image processing needs to be done in order to extract the information from images.

Image processing is done with the help of some mathematical operations. These operations use any kind of signal processing. An image is provided as the input for this processing. Output of image processing can be some parameters or characteristics of that image. Every image contains some sub-images which are also known as regions. Also, every image contains some objects which act as the basis for regions. Generally, the images must be digitized for image processing. In digitization process, the image is sampled and then these samples or pixels are quantized. Digital image is showed by first converting it into the analogue signal which is scanned onto an output.

Work Flow of Image Processing

Digital Image Processing includes the following procedures:

1. **Image Acquisition-** First of all we need to collect the images from any source. These images can be medical images, special images, or any normal image.
2. **Image Pre-processing-** Image pre-processing is performed in order to perform some correction in the image such as geometric corrections. Image Enhancement is performed here, which includes contrast enhancement, noise removal etc.

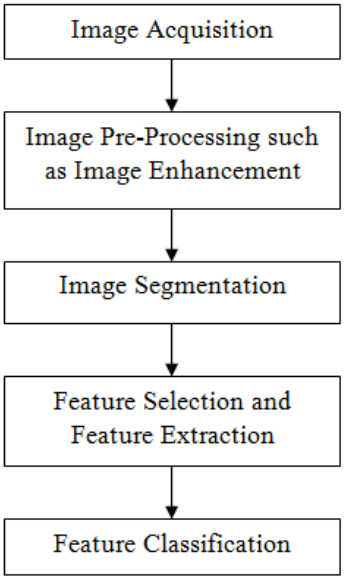

Fig. 2. Work Flow of Image Processing

3. **Image Segmentation-**In this we classify the images into various groups. It includes various kind of clustering methods such as: k-means clustering, subtractive clustering etc.
4. **Feature Selection & Extraction-**These techniques are applied to get features that will be useful in classifying and recognition of images. Features show the behaviour of an image.
5. **Classification-** This step is generally implemented in geographical images to identify land, water, soil etc.
6. **Image Output-** Finally we get processed image or features as the output of image processing [4, 5].

II. Literature Review

We have studied papers on BigData, Hadoop, Image Processing, HIPI, OpenCV, CUDA, and MIPr since 2008 to 2016.

Wu, X., et.al., [1] explains the rise of BigData with the help of some applications and examples. He presented some challenges in this field. **Khan. M., et.al. [2]** Discussed 7Vs of BigData. He also discussed various aspects of BigData with the help of various applications. **Mukherjee S., Shaw R. [3]** discussed software tools to process this huge amount of data. He explained Hadoop with respect to BigData. **Basavaprasad B., Ravi M. [4]** has discussed image processing in this paper. **Tong J., et.al.[5]**has discussed about the importance and various techniques of image processing. **Sweeney C., et.al [6]** discussed about Hadoop Image Processing Interface. He discussed the working of HIPI in 3 stages. **Barapatre H.K., et.al [7]** explained the full process to set up Hadoop on windows. Also, he has presented his experiment to prove that when this module processed large number of small files, it gave better processing time. **Dhinakaran, K., et.al. [8]** Explained MIPr framework. Author has presented the representation of images with the help of input/output tools. **Arsh S., et.al [9]** discussed some tools for image processing in distributed environment. He discussed about HIPI and OpenCV. **Deepthi R.S., Sankaraiah S. [10]** discussed about OpenCV and its application in image processing. **Patel H.M., et.al [11]** discussed MPI and CUDA as other tools for image processing in distributed environment. **Yang Z., et.al. [12]** Discussed CUDA tool with image processing in distributed environment.

III. Image Processing Tools in Parallel Environment

At present, there is huge amount of unstructured data that is processed or need to be processed every day. Large parts of this data are the images that are being uploaded every singles day. This huge amount of image data comes from various social sites, satellites, medicals etc. This type of data is highly unstructured. Traditional databases can't be used to

 ISSN: 2454-4248
532 – 536

manage and process them. This type of data requires high computation and processing power and more number of resources. So following are some tools to process these images in parallel manner:

3.1 Hadoop Image Processing Interface (HIPI)-

Hadoop Image Processing Interface (HIPI) is a framework to perform image processing on large-scale. This framework is based upon Hadoop. It is designed to work with Hadoop MapReduce which is a parallel processing module. With the help of MapReduce, these large amounts of images can be processed in parallel distributed manner. HIPI runs with Hadoop and all its vast image data is stored on the distributed file system of Hadoop. This data is provided to the MapReduce for efficient processing. Hadoop Image Processing Interface hides all the technical details of Hadoop ecosystem. Generally HDFS is used to store files on various nodes in Hadoop but, it is difficult for this distributed file system to store such a huge amount of image data. In order to store these image files in the distributed file system, some operations are performed by the downloader module of HIPI. Downloader module of HIPI is a node which is responsible for downloading the images. There are three stages in HIPI to process an image, which are:

1. Culling
2. Mapping
3. Reducing

First of all, the input is provided to the HIPI in the form of HIPI Image Bundle (HIB). It is actually a file stored on HDFS. Various tools are used to create these HIBs. These tools include a MapReduce program which helps in downloading the images from the image sources.

In Culling phase, a HIB is provided to the framework where the images are filtered on the basis of the specification provided in the program by user. In Map phase, all the images that survived through the Cull phase are provided in order to increase the locality of data. After this, all the images are presented to the mapper individually. Key-value pairs are generated in this phase. The output of mapper phase works as the input for the reducer. To reduce the usage of network bandwidth and pre aggregated key-value pair the third phase, i.e., Shuffle step is done [6, 7]. At last, all the reduce tasks, defined by the user, are executed in parallel manner to get some outputs. These outputs are then aggregated to get the final result from HIPI framework.

3.2 MapReduce Image Processing Framework (MIPr)-

MapReduce Image Processing Framework (MIPr) is based on MapReduce and its implementation is done on Hadoop. Some image processing API are included in MIPr for them who are not familiar with Hadoop. Many forms for image representation and input/output tools are provided in Hadoop by MIPr. MapReduce performs the task in two phases: Map phase and Reduce phase. There can be a number of map functions. Each map function will process single image. These functions can process different or related images. At last, the reduce function will combine the output of all the map functions. MIPr provides an interface in which we can create some function. With these functions, we can process a single image or a group of image at the same time.

Image representation in MIPr framework in done on the basis of two famous image processing libraries: Java 2D and OpenIMAJ. All the images, which are in the given format, can be used as the values in MapReduce program. InputFormat reads the images from HDFS. To process these images, a key-value pair is generated and then Map function is performed. OutputFormat writes back the image to HDFS after processing. The information about the image is stored in the metadata fields.[8]

3.3 Open Source Computer Vision Library (OpenCV)-

Open Source Computer Vision Library (OpenCV) was developed by a number of programmers so that Image Processing can be incorporated in a variety of computer languages. It can run on Windows, Linux, Android, and Mac operating system. The interfaces of OpenCV are in C, C++, and Python. OpenCV is a library so it has defined functions. OenCV have four modules:

cv – These are main OpenCV functions.
cvaux – Auxiliary OpenCV functions.
cxcore – Data structures and linear algebra support.
highgui – GUI functions.

To read the images from distributed file system of Hadoop, two types of InputFormat have been developed for OpenCV. OpenCVFileInputFormat reads single image at once, whereas OpenCVCombineFileInputFormat can read multiple images at the same time from HDFS.[9, 10]

3.4 Compute Unified Device Architecture (CUDA)-

Compute Unified Device Architecture (CUDA) is a technology for general purpose computing on Graphics Processing Unit (GPU). CUDA can implement some classical algorithms of image processing such as histogram equalization, removing noise, edge detection etc. CUDA have a development environment like C, and also, it uses C compiler to compile its programs. CUDA provides a heterogeneous interface for programming by running the code on two separate platforms: a device and a host. Host contains CPU, memory etc. and the device contains video card having a GPU (CUDA-enabled).

 ISSN: 2454-4248
532 – 536

CUDA is good in processing big images such as airplane and satellite images, having high resolution. Traditional methods of image processing can't process these types of big images as per requirement. Whereas CUDA provides highly parallel processing, suitable for these kind of images. Also, it is cheaper in hardware implementation. CUDA works by implementing threads to perform processing. We can assign one thread to one pixel. Each thread will be responsible for calculating the final result of the image. It is massively parallel processing device architecture and can process 32 threads in parallel. It contains a number of blocks. These are the blocks of threads where each block can contain threads in the multiple of 32. [11, 12]

IV. Comparative Study of Image Processing Tools

Table 1. Comparison between various Image Processing Tools

Parameters	HIPI	MIPr	OpenCV	CUDA
Image Storage	Images are stored in HDFS as a HIPI Image Bundle	Each image is stored in HDFS as a separate file	Images are stored separately in HDFS	Images are stored on the host machine of the system
Image Size	It can process only one large file of image at a time	It can process the images in both large and small files	It has different format to process both large and small image files	It works best in processing large image files.
Image Representation	It does not have convenient image representation	It is done on the basis of popular image processing libraries such as: Java 2D and OpenIMAJ	It uses its own library functions for image representation	It uses its own functions for image representation
Type of Image	It can read-write only color images	It can read-write color, grey scale, and black-white images	It can read-write color, grey scale, and black-white images	It can read-write color, grey scale, and black-white images
Programming Language	Java	Java	C, C++, Python	C, C++
Image Processing Capabilities	It does not need any additional API. HIPI has in-built image processing capability	It offers image processing APIs for those who are not familiar with Hadoop	It includes elementary IP tasks. We can process some IP algorithms with its help	Some classical IP algorithms can be implemented with its help
Interoperability	Less interoperable because images have to be bundled due to which other system can't read them	High interoperable because images are stored separately and can be used by other system	High interoperable because images are stored separately and can be used by other system	High interoperable because images are stored separately and can be used by other system

4. 1 Discussion

Image in HIPI are stored in the form of HIB because HIPI can't process these images one by one while in other tools images are stored separately. HIPI and CUDA works best for large files whereas MIPr and OpenCV have different file formats to process the image files of different sizes. FileInputFormat is used when large files are processed and CombineFileInputFormat is used to combine small files into one large file. HIPI is poor in functionalities than other tools. It is made for image processing only while other tools

 ISSN: 2454-4248
532 – 536

have some other functionality as well. HIPI and MIPr supports are made in Java language and their programming is also done in Java, while OpenCV and CUDA are developed in C and C++ respectively. OpenCV have interfaces in C, C++, and Python whereas CUDA provide a C like interface for programming.

V. Conclusion

As we know that BigData is very huge in size and it contains various types of data which is important for us in better decision making. In this paper we have discussed about image data only. To process some a huge number of images in fast speed, we need some tool to process them in distributed environment. For this reason, we have discussed some of these tools and performed a comparative study on these tools. In this paper we have discussed only four image processing tools: HIPI, MIPr, OpenCV, and CUDA. We have done their comparison on the basis of certain parameters. We have found that most suitable tool for image processing in distributed environment is Hadoop Image Processing Interface.

References

[1] Wu, X., Zhu, X., Wu, G. Q., Ding, W.: Data Mining with Big Data, Digital Object Identifier 10.1109/TKDE.2013.109, 1041-4347/13/$31.00 © 2013 IEEE, USA (2013)

[2] Khan, M., Uddin M.F., Gupta N.: Seven V's of Big Data Understanding Big Data to extract Value, Conference of the American Society for Engineering Education (ASEE Zone 1), 978-1-4799-5233-5/14/2014 IEEE (2014)

[3] Mukherjee S., Shaw R.: Big Data – Concepts, Applications, Challenges and Future Scope, International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 2, February 2016, Kolkata, India (2016)

[4] Basavaprasad B., Ravi M.: A Study on the Importance of Image Processing and It's Applications, National Conference on Recent Innovations in Engineering and Technology-2014, International Journal of Research in Engineering and Technology, Volume: 03 Issue: 01 | Jan-2014, Karnataka, India (2014)

[5] Tong J., Cheng-Dong WU, Dong-Yue C.: Research and Implementation of a Digital Image Processing Education Platform, Supported by Fundamental Research Funds for the Central Universities of China, Grant No. N090304001, 978-1-4244-8165-1/11/$26.00 ©2011 IEEE, Shenyang, China (2011)

[6] Sweeney C., Liu L., Arietta S., Lawrence J.: HIPI: A Hadoop Image Processing Interface for Image-based MapReduce Tasks, Virginia (2015)

[7] Barapatre H.K., Nirgun V., Jagtap H., Ginde S.: Image Processing Using MapReduce with Performance Analysis, International Journal of Emerging Technology and Innovative Engineering Volume I, Issue 4, April 2015, Mumbai, India (2015)

[8] Dhinakaran, K., Prasanthi K., Kumar P.: Distributed Image Processing Using HIPI, IJCTA, 9(12), 2016, pp. 5583-5589, © International Science Press, Chennai, India (2016)

[9] Arsh S., Bhatt A., Kumar P.: Distributed Image Processing Using Hadoop and HIPI, 2016 Intl. Conference on Advances in Computing, Communications and Informatics (ICACCI), Sept. 21-24, 2016, 978-1-5090-2029-4/16, IEEE, Nagpur, India (2016)

[10] Deepthi R.S., Sankaraiah S., Implementation of Mobile Platform Using QT and OpenCV for Image Processing Applications, 2011 IEEE Conference on Open Systems (ICOS2011), September 25 - 28, 2011, Langkawi, Malaysia, 978-1-61284-931-7/11/$26.00 ©2011 IEEE, Karnataka, India (2011)

[11] Patel H.M., Panchal K., Chauhan P., Potdar M.B.: Large Scale Image Processing Using Distributed and Parallel Architecture, International Journal of Computer Science and Information Technologies, Vol. 6 (6) , 2015, 5531-5535, Gujarat, India (2015)

[12] Yang Z., Zhu Y., Pu Y.: Parallel Image Processing Based on CUDA, 2008 International Conference on Computer Science and Software Engineering, 978-0-7695-3336-0/08 $25.00 © 2008 IEEE, DOI 10.1109/CSSE.2008.1448, China (2008)