HIPI: Hadoop Image Processing Interface

Shri Vivekanand Reshmi[1] Shri Vijay Kumar Badiger[2] Smt Sneha Kadadevar[3] Shri Kiran Javali[4]
[1,2,3,4]Lecturer
[1,2,3,4]BVVS Independent PU College Bagalkot, India

Abstract— The amount of images being uploaded to the internet is rapidly increasing, In most extensively used social media platform –Facebook users uploading over 2.5 billion new photos every month, however, applications that make use of this data are sternly lacking. Current computer vision applications use a small number of input images because of the difficulty is in attaining computational resources and storage options for huge amounts of data. As such, development of vision applications that use a large set of images has been inadequate. The Hadoop Mapreduce platform provides a system for large and computationally intensive distributed processing, though use of Hadoop system is rigorously limited by the technical complexities of developing useful applications. To ease this complexity, various image processing frameworks have been introduced which summaries the MapReduce model while converging on image processing task. Hadoop Image Processing Interface is one of those frameworks with many features and that objects to create an interface for computer vision with MapReduce technology.

Keywords: HIPI: Hadoop Image Processing Interface, Big Data Image Processing

I. INTRODUCTION

In the real time consequence, the volume of data used linearly increases with time. Usually the growing data are ordinarily unstructured. Every day the users of social media like Facebook and Twitter are growing in an incredible range. The use of data presented in image format in fields of satellite imaging, medical imagery, astronomical data analysis, computer vision etc. has been increased over the years. And as a result of it, necessities to process those images have also been increased. Various algorithms, tools and techniques have been developed to anatomize and process those images. Hadoop and HIPI technology that aids organization to accomplish image data efficiently and commendably.

II. HADOOP

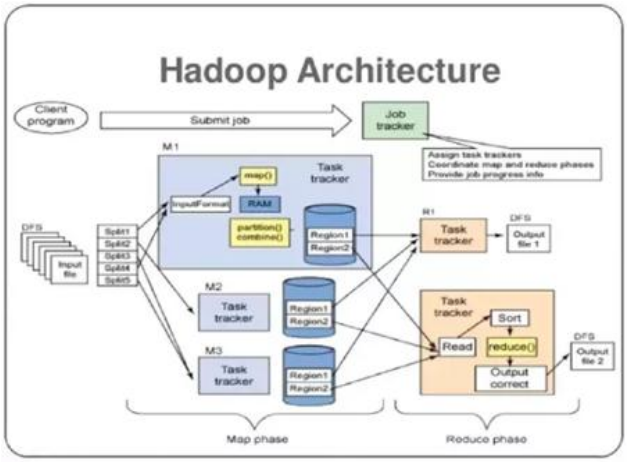

Fig. 1: Hadoop Architecture

Hadoop is an Open Source implementation of a significant batch processing system. Hadoop is an Apache project being built and used by a global communal of contributors, written in the Java programming language.[5] It provides a distributed file system and a framework for the analysis and transformation of very large data sets using the Map Reduce paradigm. it allows developers to deploy custom written programs coded in Java or any other language to process data in a parallel fashion across hundreds or thousands of commodity servers. An important characteristic of Hadoop is the partitioning of data and computation across thousands of hosts, and executing application computations in parallel close to their data. MapReduce is core of Hadoop [1] ecosystem.

A. Map-Reduce is a programming model and an allied implementation for processing and generating large data sets [4]. MapReduce basically consist of two phases: Map and Reduce, and an added hand-off process called Shuffle and Sort. Mapper and Reducer are the interfaces implemented aiming to provide the map and reduce methods.

III. BIG DATA IMAGE PROCESSING:

Predictable image processing systems are not accomplished of processing big image data [6]. Hadoop is effective at processing textual data, but when it comes to processing images, it becomes moderately difficult meanwhile the data of the image to be managed is taken as String format. Other than this, there is a main known problem named as Small File Problem.

IV. HIPI: HADOOP IMAGE PROCESSING INTERFACE

HIPI is an image processing library designed to be used with the Apache Hadoop MapReduce parallel programming framework. HIPI eases competent and high-throughput image processing with MapReduce style parallel programs normally executed on a cluster. It provides a key for how to store a huge group of images on the Hadoop Distributed File System (HDFS) and make them obtainable for efficient distributed processing.

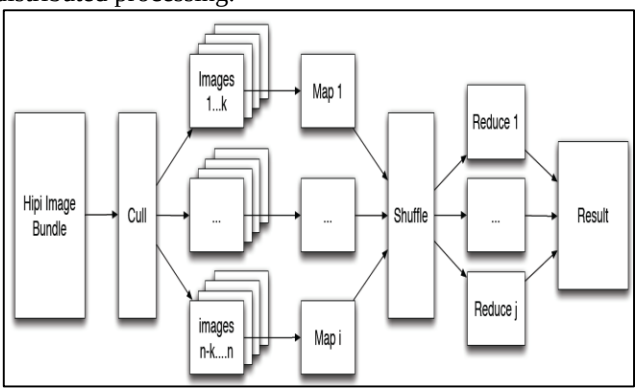

Fig. 2: The Workflow of Hadoop Image Processing Interface

This is intended to be used with the Apache Hadoop Map Reduce. HIPI is used for enhanced performance image processing. It is operative with Map Reduce style parallel programs executed on a cluster environment. Huge collection of images on the Hadoop Distributed File System (HDFS) and it is available for distributed processing.[2] The input object to a HIPI program is a HipiImageBundle (HIB).

1) Hipi Image Bundle: It is open source framework. It maintained by a group of dedicated researchers and developers.
2) Cull: The initial stage of a Hadoop image processing interface program is a culling step that allows filtering the images in a Hadoop based on a user denied condition are used in spatial resolution and criteria related to the big data.
3) Images: The primary presentation for a collection of images on the Hadoop distributed file system. Map reduce is optimized to support efficient processing of large file system. HIB is actually compared two file stored on the system.
4) Shuffle: shuffle can start before and after map phase has finished to save same time. Reduce status is greater than 0 per cent but less than 3% when map status is not yet 100%.
5) Mapper: Take the data and convert into another set of data where Individual elements are broken down into tuples.
6) Reducer: Take the data from mapper and combine those data tuples into smaller set of tuples.

V. HIPI TOOLS

The HIPI distribution contains several programs placed in the tools subdirectory that are useful in their own right and will assistance you acquire how to create your own HIPI programs.

A. hibImport:

This is a simple program and a good place to start exploring HIPI. hibImport creates a HipiImageBundle (HIB) from a folder of images on your local file system. A HIB is the important input file to the HIPI framework and signifies a collection of images stored on the Hadoop Distributed File System (HDFS).

B. hibInfo:

The hibInfo tool allows probing basic information about HIBs such as image count, spatial dimensions of individual images; image meta data stored at the time of HIB creation, and image EXIF data. It also allows mining individual images as a stand-alone JPEG or PNG.

C. hibDump:

Similar to hibInfo, but this is a MapReduce/program that extracts basic information about the images in a HIB. It does this using multiple parallel map tasks (one mapper for each image in the HIB) and writes this information to a text file on the HDFS in a single reduce task.

D. hibDownload:

This is a MapReduce/HIPI program that creates a HIB from a set of images found on the Internet. This program highlights some of the more elusive parts of HIPI and the Hadoop framework and will be a prised tool for creating inputs. It is also intended to work flawlessly with the Yahoo/Flickr 100M Creative Commons research dataset.

org.hipi.image	A group of classes for representing and manipulating images.
org.hipi.image.io	A group of classes for reading (decoding) and writing (encoding) images in various storage formats such as JPEG and PNG.
org.hipi.imagebundle	Classes for storing and manipulating collections of images.
org.hipi.imagebundle.mapreduce	Classes for processing HIBs including producing input splits and reading and decoding image records in the context of a MapReduce program.
org.hipi.mapreduce	Extensions to the basic MapReduce framework specific to HIPI, including a Culler class that enables efficient culling of images during HIB processing.
org.hipi.opencv	Helper classes and utility functions which facilitate interactions between HIPI and OpenCV.
org.hipi.util	Helper functions for working with byte arrays during object serialization.

Table 1: HIPI Packages

E. hibToJpeg:

This is a MapReduce/HIPI program that extracts the images within a HIB as discrete JPEG files written to the HDFS. This program explains many important features of the HIPI API (e.g., how to process the images in a HIB based on the MapReduce programming model). It is also a valuable tool to prove that a HIB has been properly created.

F. Covar:

This is a MapReduce/HIPI program that gears the experiment described in the paper The Principal Components of Natural Images, written by Hancock et al. in 1992. This program computes the principal components of natural image patches (eigenvectors of the covariance matrix computed over a large set of small image patches). This is a good starting point for

learning how to build more complex HIPI programs and also illustrates HIPI's ability to interface with OpenCV.

VI. THE MAIN GOALS OF THE HADOOP IMAGE PROCESSING FRAMEWORK ARE

- Provide an open source framework over Hadoop MapReduce for developing large-scale image applications.
- Provide the ability to flexibly store images in various Hadoop file formats.
- Present users with an intuitive application programming interface for image-based operations which is highly parallelized and balanced, but which hides the technical details of Hadoop MapReduce.
- Allow interoperability between various image processing libraries.

VII. CONCLUSION

Hadoop image processing library support to store and retrieve the huge size of image bundle with Hadoop distributed file system. Huge volume of visual data is acquired by biomedical imaging, remote sensing, astronomy, Internet and their need for efficient and effective processing profess the use of Hadoop image processing. Hadoop image processing is use to medical diagnosis such as brain tumour detection, iris detection, face recognition etc. Hadoop image processing frame work eases efficient and high-throughput image processing with MapReduce frame work. Hadoop base, cloud computing, image segmentation, thresholding and image enhancement technique are used in Hadoop image processing.

REFERENCES

[1] Apache Hadoop. http://hadoop.apache.org/
[2] http://ijariie.com/adminuploadpdf/feature_extraction_in_hadoop_image_processing_interface_ijariie8487.
[3] http://hipi.cs.virginia.edu/examples.html
[4] J. Venner, Pro Hadoop. Apress, June 22, 2009.
[5] https://www.academia.edu/6920246/IJIRET-Vivekanand-S-Reshmi-Significance-of-HADOOP-Distributed-File-System

