International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 03 Issue: 06 | June -2019 ISSN: 2590-1892

"Human Iris Recognition for Biometric Identification Using Apache Spark with RDD. "

Guruprasad jathar, Trupti Lawande, Amol Bansode, Shivlila Kantekure

ABSTRACT

Iris recognition is an automated method of biometric au-thentication that uses mathematical pattern recognition techniques on images of the iris of a human eye, whose complex random pat-terns are unique and can be seen from some distance[10], many millions of persons in several countries around the world have been enrolled in iris recognition systems, for convenience purposes such as passport free automated border crossings, and some national ID systems like Aadhar Card System based on this technology are being deployed. The main advantage of iris recognition, besides its speed of matching results and its extreme contradition to false matches, is the stability of the iris as an internal, protected, exter-nally visible part of the eye[11].

Now in current world most of the iris comparison sys-tems uses sequential and parallel execution but when iris dataset is large i.e. Big Data then in that case it has certain deficiencies like speed, complexity of dividing data, handling large data and robustness. So we can implement Iris recognition process using an open source technology known as Hadoop. Hadoop technol-ogy is based on most popular programming model used to handle big data i.e. MapReduce framework[3]. Hadoop provides specific library for handling large number of images: Hadoop Image Pro-cessing Interface and it can be used to implement the proposed system. Hadoop Distributed File System (HDFS) is used to han-dle large data sets, by breaking it into blocks and replicating blocks on various machines in cluster. Template comparison is done in-dependently on different blocks of data by various machines in parallel[3]. Map/Reduce programming model is used for process-ing large data sets. Map/Reduce process the data in key-value for-mat. Iris database is stored in a key-value text format. Mappers pro-cess the input and produce an intermediate output. Reducer takes intermediate output and produces final result. This project work shows how, the most time-consuming operations (matching pro-cess) of a modern iris recognition algorithm are parallelized. In par-ticular, template matching is parallelized on a Hadoop based system with a demonstrated speedup gain[3].

We are using Hadoop image processing interface to im-prove the speed of processing large number of small sized images. HIPI is a library for Hadoop's MapReduce framework that provides an API for performing image processing tasks in a distributed com-puting environment.

Keywords

Hadoop, Apache Spark, Biometric Identification, Iris recognition, HIPI etc.

1. INTRODUCTION

A biometric is a biological measurement of any human physiological or behavior characteristics that can be used to verify the identity of individual.Biometric authentication (or identifica-tion) systems, which use physical characteristics to check a persons identity, ensure much greater security than password and number systems. A biometric system provides automatic recognition of an individual based on some sort of unique feature or characteristic possessed by the individual. Biometric systems have been devel-oped based on fingerprints, facial features, voice, hand geometry, handwriting, the retina, and the one presented in this paper.Iris is the main important part of the human eye; it consists of circular muscle and the other longitudinal control in the amount of light passing the retina through the human eye.A biometric is character-ized by use of a feature that is decidedly unique so that the chance of any two human having the same features will be minimal[2].

2. IRIS RECOGNITION

Iris recognition is a method of biometric identification, based on extraction of features of the iris of an individual's eyes. Each individual has a unique iris. The variation even exists be-tween identical twins and between the left and right eye of the same person[1].Formation of the unique patterns of the iris is random and not related to any genetic factors.The iris is part of human eye and is a thin circular type, which lies between cornea and lens of the hu-man eye. The iris is close to its centre by a circular aperture known as the pupil. The function iris is to control amount of light entering through the pupil, and it is done by sphincter and dilator muscles, which adjust size of pupil. The average diameter of iris is 12 mm, and the pupil size can vary from 10% to 80% of the iris diameter. Iris formation process begins during the third month of embryonic life. A unique pattern of the surface of the iris is formed during the year of life, and pigmentation of the stoma takes place for the few years[1].

2.1 Iris Recognition Process

An Iris recognition process consists following main sub processes

2.1.1 Image Acquisition: . In this process eye image is captured.The image can be captured from live video camera or can be used already stored in memory or from a Dataset.

2.1.2 Image Preprocessing: . Preprocessing consists of imagefiltering and enhancement, iris image localization and normaliza-tion. Captured Image is converted into gray scale image if it is col-ored one. Canny edge detection algorithm is applied to detect the edge map of the image. For detecting inner and outer boundaries for pupil and iris, Hough Transform technique is used.

2.1.3 Feature Extraction: . Iris structure has complicated orcomplex and plentiful textures which can be extracted as features for coding. The Extracted Feature vector is compared with the al-ready stored iris templates. Integer feature vector method can be used to compare with iris templates.

3. EXISTING SYSTEM:

3.1 Parallel and Distributed Processing on Hadoop

As the structure of the system, It consists of two components which are the Hadoop Distributed File System (HDFS) and MapReduce, performing distributed pro- cessing by single-master and multi-pleslave servers. The MapReduce have two elements , namely JobTracker and TaskTracker, and two elements of HDFS, namely DataNode and NameNode[2].

3.1.1 JobTracker: . JobTracker manages cluster resources andjob scheduling to and monitoring on separate components.

3.1.2 TaskTracker: . TaskTracker is a slave node daemon in thecluster that accepts tasks and returns the results after executing tasks received by JobTracker.

3.1.3 NameNode: . An HDFS cluster consists of a single Na-meNode, a master server that manages the file system namespace and regulates access to files by clients. NameNode executes file sys-tem name space operations, such as opening, closing, and renam-ing files and directories. It also determines the mapping of blocks to Data Nodes[2].

3.1.4 DataNode: . The cluster also has a number of DataNodes,usually one per node in the cluster. DataNodes manage the storage that is attached to the nodes on which they run. DataNodes also perform block creation, deletion, and replication in response to di-rection from NameNode[2]

3.1.5 SecondaryNameNode: . SecondaryNameNode is a helperto the primary NameNode. Secondary is responsible for supporting periodic checkpoints of the HDFS metadata.

3.2 Hadoop Distributed File System (HDFS)

HDFS is designed to reliably store very large less aeross machines in a large cluster. It is inspired by the Google File System. HDFS is composed of NameNode and DataNode. HDFS stores all files as a sequence of blocks (currently 64 MB by default) with all blocks in a file the same size except for the last block. Blocks belonging to a file are replicated for fault tolerance[2].

3.3 MapReduce

MapReduce (implemented on Hadoop) is a framework which uses parallel distributed processing for large volumes of data. In programming using MapReduce, it is possible to perform parallel distributed processing by writing programs involving the following three steps: Map, Shuffle, and Reduce. Because Map Reduce automatically performs inter-process communications be-tween Map and Reduce processes, and maintain load balancing of the processes[2].

4. PROPOSED SYSTEM:

4.1 Apache Spark:

In proposed system we are using apache spark with RDD for achieving parallelization with the help of HIPI framework for iris recognition of human iris[16].The HIPI (hadoop image pro-cessing interface) framework facilitates efficient and high through-put image processing with map reduce style.It provides solution for how to store a large collection of images on HDFS (hadoop distributed file system) and make them available for efficient dis-tibuted processing[2].

One of the main limitations of Hadoop-MapReduce is that it persists the full dataset to HDFS after running each job[2]. This is very expensive, because it incurs both three times (for repli-cation) the size of the dataset in disk and a similar amount of net-work.Spark uses the pipeline approach to perform operations.When an output of one operation needs to perform another operation Spark passes the data directly without writing to persistent storage.

We are using Apache Spark because the main advantage of Apache Spark was to introduce an in-memory caching abstrac-tion.This makes Spark ideal for workloads where multiple oper-ations access the same input data[2]. Users can instruct Spark to cache input data sets in memory, so they don't need to be read from disk for each operation. This will simply reduce the disk access operation for each user for same data[2].

The primary advantage Spark has here is that it can launch tasks much faster. MapReduce starts a new JVM for each task, which can take seconds with loading JARs, JITing, parsing configuration XML, etc. Spark keeps an executor JVM running on each node, so launching a task is simply a matter of making an RPC to it and passing a Runnable to a thread pool, which takes in the single digits of milliseconds[2].

5. SYSTEM ARCHITECTURE:

Image Acquisition is process of retrieving image from any source for processing. Sobel edge detection means finding edges in image. It is used to find the boundary of objects. Sobel edge detection technique was applied on iris images to determine edges present in the input image. In thresholding we seprate the dark and light regions. Hough transform is process of detecting straight lines in an image. In Pupil detection, pupil is detected from the input image. Gabor filter is the algorithm used for the iris recog-nition. In the gabor filter there are five phases. Image Acquisition is the first step, in this step image is taken and send it to the next phase i.e. preprossing. In the preprocessing phase, the unwanted part from the image is removed. In Feature Extraction phase, useful features from the image is extracted then it is matched with the available dataset by using the single node clustring or multinode clustring.

5.1 Single node clustering:

In single node clustering the system is working only on single node and all the processing of data is done at this node. The input image is processed for patterns present in iris and then it will be sent for further processing of comparing that image with the database. After checking it will show the result that image is present or not in the database.

5.2 Multi node clustering:

The processing of input image and comparison of that image with database is same as single node. The difference id that in multi node there are severaal worker nodes which improves the processing speed of data comapared to the single node. More than one worker nodes are present, hence the processing is done much faster.

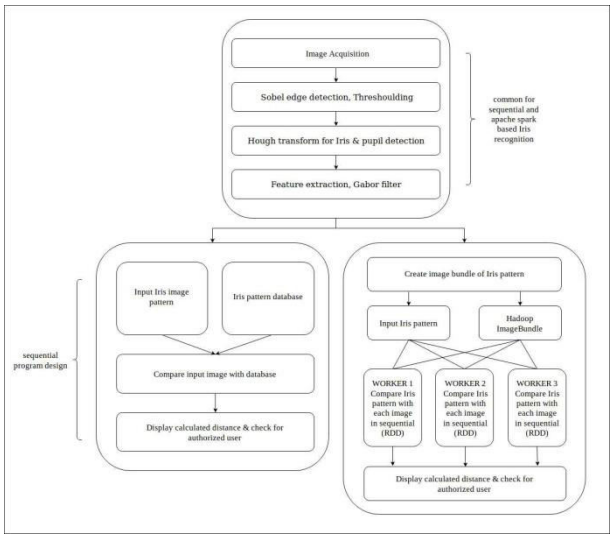

Fig. 1. System Architecture

6. PERFORMANCE:

The above graph represents the difference between the apache spark and hadoop. The apache spark is faster than hadoop in almost every execution of same datasets. same size of datasets are given to apache spark and hadoop First we gave the 421 MB of dataset to both the apache spark and hadoop. The hadoop system takes more time for processing as compared to spark. In this 421 MB of dataset, the spark system takes 18 seconds to successfully process the dataset while hadoop system takes 138.92 seconds to process the same dataset In second iteration, we doubled the size of the datasets which is 842 MB. The spark system successfully process this dataset in 18 seconds and hadoop system takes about 199 seconds to process the whole dataset As we increase the size of datasets the hadoop system requires more time as compare to the spark system. In first iteration the spark system is almost 7 to 8 times faster than the hadoop. In the last iteration we gave the datasets of size 2.53 GB to both the systems, the spark system takes 36 seconds to process all the data while hadoop takes 701 seconds to process the dataset. Hence from above results we have concluded that the spark is about 20 times faster than hadoop.

The configuration of our system to run this job is, processor - Core i7 8th Gen - (16 GB/512 GB)

7. RESULT:

Below diagram shows the result of iris comparison. In this, when we give the input image as a human face then it recognize eyes from face and from the eyes it detects the pupil, And it is matched with the dataset. When we give eye as input to our

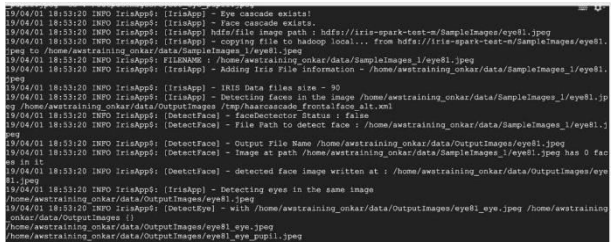

Fig. 2. Results.

program then it detects pupil form that eye and match this pupil with the given dataset. And if we give iris image as a input to our program then it directly matched with the available dataset. If the image present in our dataset it shows result true and if it is not present in our dataset it shows false result to us.

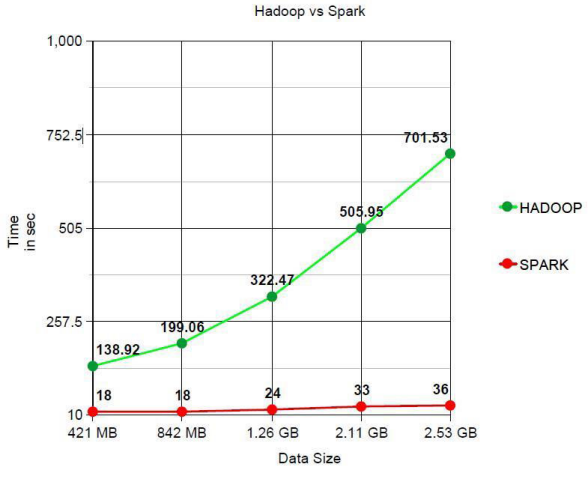

Fig. 3. Hadoop vs Spark.

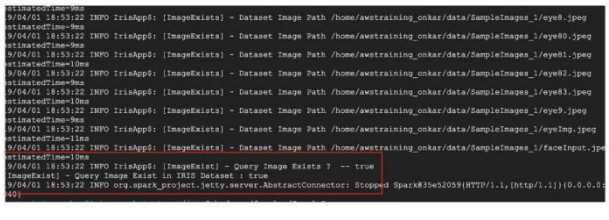

Fig. 4. Results.

8. REFERENCES

[1] John Daugman. How iris recognition works. In *The essentialguide to image processing*, pages 715–739. Elsevier, 2009.

[2] Agus Harjoko, Sri Hartati, and Henry Dwiyasa. A method for iris recognition based on 1d coiflet wavelet. *world academy ofscience, engineering and technology*, 56(24):126–129, 2009.

[3] NS Raghava et al. Iris recognition on hadoop: A biometrics system implementation on cloud computing. In *2011 IEEE In-ternational Conference on Cloud Computing and Intelligence Systems*, pages 482–485. IEEE, 2011.

