INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS.....

Object Detection Using Hadoop

Aruna U Gawde[1], Miraj Shah[2], Imaad Ukaye[3], Mihir Nanavati[4]

[1]Professor, [234]Student of Computer Department
[1234]Dwarkadas J Sanghvi,
Vile Parle (w), Pin no.400056

[1] aruna.gawade@djsce.ac.in, [2] mirajshah05@hotmail.com,
[3] imaad.ukaye@gmail.com, [4] nanavati93@gmail.com

Abstract: *The promise of data-driven decision-making is now being recognized broadly, and there is growing enthusiasm for the notion of ``Big Data.'' With the advent of Big Data problems start right away during data acquisition, and how to store what we keep reliably with the right metadata. Much data today is not natively in structured format. With the increasing popularity of cloud computing, Hadoop has become a widely used open source framework for large scale data processing. In this paper, we propose to use the Hadoop Map Reduce frame-work to that allows processing of extremely large video files or image file on data nodes. This project aims at processing these input images in real-time using low-end computers. The input image received from the user will direct the node-head to search for a particular object within the stored files and return the output with the files in which the object will be found. The input storage may be extended to terabytes of storage files without degrading the systems performance.*

Keywords: *Distributed processing, video processing, Image processing, video conversion in Hadoop.*

1. INTRODUCTION

Although the computing power of machines is keeping increase in a very high speed. Almost every 3 years, CPU's computing power increase twice. However size of the files keeps increasing also in an amazing rate. To store such colossal amount of data instead of using a simple Client Server architecture it'll be better to use an architecture wherein the data exhibits the property of logical independence. A system where in the data must be distributed on a large number of workstations so that it may reduce the burden of analysis on a single machine. Video processing is very well suited to distributed system implementation. Processing in the Hadoop is inherently distributed. Hadoop supports parallel running of applications on large clusters of commodity hardware. [6] "Hadoop Library is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly available service on top of a cluster of computers, each of which may be prone to failures." – Apache.

Our idea is very simple. Assume there is a very large video data base. Giving a set of video frames or image, we hope to find it from that database, and tell the position of that input file. The idea is simple but it is very useful in different aspects. If we change the algorithm to an object detection algorithm, it can be used in a surveillance video application. If we put face detection algorithm, it become a video diary. The key point of this project is building application with high scalability. When database is increased, the application can still handle it.

Project Requirements:

Desktop PC Memory 1-4 GB Storage 160 GB (depending on application)	Data Node
Desktop PC Memory 4 GB or more Storage 160 GB (depending on application)	Name Node

Table 1: Hardware Requirements

Table 1 show the minimum requirements to set up a hadoop cluster requiring at least 1 namenode and is scalable to 'n' data nodes. We can reconfigure it for 2 namenodes for better reliability.

1.1 Any Supporting Operating System

Ubuntu (64-bit): We have used Ubuntu as it is the most common platform to set up cloud applications. There are more support and discussion on the internet. Ubuntu and hadoop is the perfect match. Ubuntu is the ideal foundation for Hadoop clusters because it is based on open industry standards, optimized for the cloud, and able to scale with no increments in licensing costs.

1.2 Hadoop:

The Hadoop platform was designed to solve problems where you have a lot of data — perhaps a mixture of complex and structured data — and it doesn't fit nicely

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS.....

into tables. It combines the advantages of distributed processing and centralized server processing at low cost. Hadoop uses Hadoop Distributed File System (HDFS) providing high throughput access to application data and is suitable for applications that have large data sets. In our project it enables terabytes of storage and retrieval of information, analysis of data, distributing the data and task, creating mappers for each data node and carrying out the job of collecting the result and final analysis of result-set using a reducer.

2. PROPOSED SYSTEM

2.1 System Architecture

Nowadays, the most common post-event investigations on surveillance video are done by hardworking security officers who manually browse through hours of video footage.

With a good video analytics platform, we can further leverage the structured insights Hadoop provides by using an advanced query language like SQL.

A simple example of this would be the question: "When did you see this red car yesterday?" A more sophisticated question would be, "How fast did this red car drive in the parking lot yesterday between 9AM to 12PM?" These sorts of questions can be written in a simple SQL SELECT query. We propose a system in which we will try to moderate the requirement of the computational power and resources by distributing the knowledge discovery task among multiple low-cost nodes governed by a master node. Each of these nodes will perform the required task and return the results to master node. [7]

Master node will then return the compiled report to the user. Then there is an algorithm where we will analyze a given input image and will find a video containing similar image or a scene. It is known that finding an image in a video in not a new problem, but we aim to find it in minimum time which will help us to produce near real time analysis. We will make use of at least 2 slave nodes and a master node thus increase the computation power by factor of 3.Similarly by employing more slaves(data nodes) we can increase the processing power by factor of n as well reduce the strain on the resources by factor of n.

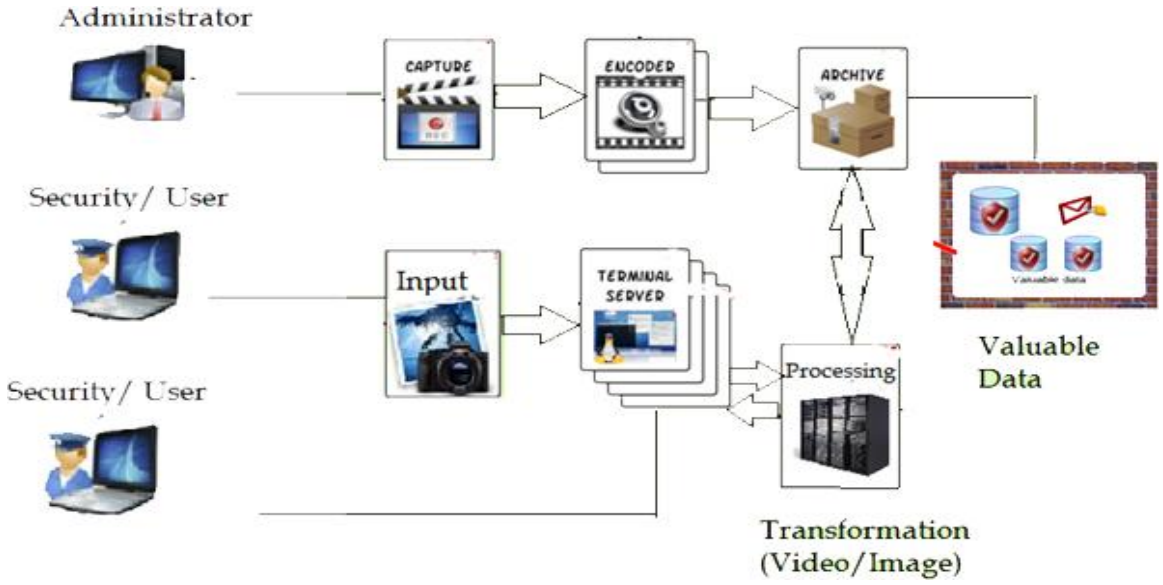

Figure 1: Video Processing in Hadoop

a. Additional Components and Libraries

Hadoop Avro

"Avro provides a convenient way to represent complex data structures within a Hadoop MapReduce job"- Apache. Avro [2] can be used basically for both input and output. Avro's format stores data structure definitions with the data, in an easy-to-process form. To present data to applications in a more generic way, rather than requiring code generation these implementations could use these definitions at runtime. In our project it helps us fetch the original data back after the processing and is used to send as an output to the user.

Sequence Files

The reason for sequence file is the "small files problem". Normally hadoop is used to retrieve very large file of terabytes of storage so the block size for each file is kept large. By default hadoop offers 64MB of block size however, storing every small file in such a large block leads to wastage of space and time. This may lead to slower read operation. Say, we have an image frame of 1-10 MB and each frame is stored in a separate block of 64MB leading to a minimum wastage of 85% of the total storage space. Imagine terabytes of storage requires peta bytes of space. This is not feasible.

In Sequence files, the motivation is to store the file in key-value pairs while combining the entire data-set. In order to distinguish data-set of each pair, keys are used. Keys could be file name, in our case they tell us about the properties of the image.

Image Filtering:
Video Jittering: PDE Model [5] based method has been a successful tool for image restoration. Partial differential equations (PDEs) are equations that involve rates of change with respect to continuous variables.

Error Concealment: Packet Video Error Concealment with Auto Regressive Model [4] solves the problem of gaps in images. Each pixel within the corrupted block is replenished as the weighted summation of pixels within a square centered at the pixel indicated by the derived motion vector in a regression manner.

Video In painting: Detecting a moving foreground from background by repainting the sequence from previous frame and detecting the changes.

Packet Video error Concealment: Auto regressive Model is used where two block-dependent AR coefficient derivation algorithms under spatial and temporal continuity constraints are processed.

b. Procedure

Video Storage
Each input file, which may be MVI files, is first split into smaller manageable frames called File Split. For analysis to take place we need separate frames. Processing of video is complex and unmanageable. The detection of a moving object, low frame rate, blurring of an image and many more problems to solve. This is an expensive process but the cost incurred is just once. Once the video is divided into File Split we will store them in HDFS as key-value pair which is the archive stage as seen in Figure 1. Key will denote the sequence of the frame, the name of the Video file and the sequence number of the frame in the video file. The storage of file split is done namely on this key-value pair. So, logically equivalent files will be stored with alongside each other. Once this is done we have basically converted our problem domain from the video analysis to image analysis which is much simpler then the video analysis.

HDFS will then distribute this set of File Split in cluster. As the data is stored in HDFS format all the data is replicated as per settings appropriate configuration file. In order to efficiently store the key-value pair HBase is used providing real-time access to big data. Hadoop initially lacked in providing facilities to small files due to problem of large block size leading to wastage of memory and memory accesses. So we will use Sequence file for efficient use of memory. We will discuss the use and problems of sequence file later.

Search/Query
The user would like to identify a particular object from the stored files. So, a simple query with the input file will be passed to the name node. However, due to various image degradation problems and for better results we apply the filters [4] on the input object and enhance the image and separate it onto segments. These segments help in analyzing the image and separating the foreground from background.

Mapper
Now in mapping phase the job is created and further split into separate tasks which can be accomplished by each data nodes independently. It will take the input image which will be basically a jpeg image. Then, will convert this jpeg image in a frameset format which is compatible with stored video frame format. Further, will convert the input image into key-value pairs where key depends on the properties of the image rather than filename. Afterwards, we select the nodes on the cluster where we will distribute the input image in our cluster.

Now each of the individual Data node will perform the frame comparison using HIPI (Hadoop Image Processing Interface)[1][3] where we will compare all the input frame with all stored frame .Keys of all the frames which give comparison status of greater than 80 % will be returned to the Name Node in Reduce phase.

Reducer
Reducer checks the status of all data nodes and collects all the result. In case of any failure it will again redistribute the failed task and complete the mapped phase. The final results will be filtered again according to input query and result is obtained. List of all the frames thus obtained will be compiled and displayed to user. Additionally all the results will be cached in order for faster retrieval incase of similar query.

3. PROJECT FEATURES

Our project will consist of following features:

- Adding and removing of Data nodes in cluster. Node can be easily added and removed from the cluster by execution of the few simple commands.
- Running queries for mining of information in unstructured data (videos in our case) and accessing of results.
- Monitoring of complete cluster performance and maintenance.
- Automatic parallelization and distribution
- When we run our application, the Job Tracker will automatically handle all the messy things, distribute tasks, failure handling, report progress.

Some other features are:

a) Fault-tolerance
The TaskTracker nodes are monitored. If they do not submit heartbeat signals in a period of time,

they are deemed to have failed and the work is scheduled on a different TaskTracker. If a task failed for 4 times (in default), the whole job will failed. A TaskTracker will notify the JobTracker when a task fails. The JobTracker decides what to do then, it may resubmit the job elsewhere, it may mark that specific record as something to avoid, and it may even blacklist the TaskTracker as unreliable.

b) Locality optimization
Here JobTracker will assign the splits which have the same keys to nearest clusters; this may reduce the job distribution time in Map procedure and collection the output key-value pairs in Reduce procedure.

c) Backup Tasks
In MapReduce, some operations may be come straggler and increase the total running time, one of the most important reason is that some of the Mapper/Reducer will fight for the local resources such as CPU, Memory, local disk, and network bandwidth, etc. these may increase the delay latency. A simple solution in Hadoop is to have more than 1 copies of the Map/Reduce job running in n different machines if there has some empty resources/slots. If a copy finished the job, the others will be killed. The overhead of this procedure is small but the total time reduced significantly in large scale clusters operations.

4. CONCLUSION

The recent surge in the size of data and the rate at which it is generated has led to the use of distributed computing system such as Apache hadoop. Hadoop at moment is primarily focused on binary data i.e textual data and hence there is a need to enhance the use of such tool efficiently in image processing domain. The system proposed above is designed to work statically but Hadoop's real-time processing models can be used to make it real-time. The current applications of the system include feeds from closed circuit television (CCTV) surveillance and can be taken forward to integrating video broadcasting websites such as youtube to search objects in video streams.

REFERENCES

[1] HIPI: A Hadoop Image Processing Interface for Image-based Map Reduce Tasks (University of Virginia) http://cs.ucsb.edu /~cmsweeney/papers/undergrad_thesis.pdf

[2] Apache Avro™ 1.7.5 Hadoop Map Reduce guide

[3] http://avro.apache.org/docs/current Hadoop Image Processing Interface Java doc http://hipi.cs.virginia.edu/documentation

[4] Packet Video Error Concealment With Auto Regressive Model http://ieeexplore.ieee.org /xpl/?tp=&arnumber=5734825

[5] An Improved Fourth-order PDE for Noise Removal with Dissipation Reduction – Ji Jing http://ieeexplore.ieee.org/xpl/articleDetails.jsp ?tp=&arnumber=4347073

[6] Hadoop Tutorial- A. Hammad http://gridkaschool.scc.kit.edu/2011/downloads/Hadoop_tutorial-1-Introduction.pdf

[7] Hadoop: The Definitive Guide –O' Reilly, Yahoo Press