

Fraud Detection in Road Toll Systems using Apache Storm

Pratik Khairnar¹, Aditya Budhkar², Swapnil Gawali³, Prof. Ms. Shikha Agarwal⁴
^{1,2,3,4} AISSMS Institute of Information Technology, Pune (MH) India

Abstract- In general, network load happens due to various reasons but main reason is communications of frames therefore distributed systems are not capable of processing image data streams. Due this reason processing of image data stream is done in one node. As to improve quality and quantity of multimedia data we need to operate image data stream in distributed manner. In this paper, we shall present toll detection in a distributed environment which processes image data stream in real-time on Apache Storm framework. Apache storm achieves by speeding up by distributing frames onto several nodes called bolts and each processes different regions of image frames.

Keywords- 1) Apache Storm, 2) distributed system, 3) image processing, 4) framework

I. INTRODUCTION

Now a days, the data in network has been increasing and also the volume of the data is also increasing continuously. To deal with big data, more powerful resources are required to access big data and perform many calculations across multiple machines concurrently. Therefore it is necessary to process large-scaled stream image data in a distributed environment.

In this paper, we propose a system in which all vehicles are detected or counted which are passed through tolls for fraud detection for checking total amount collected in tolls and number for vehicles passed. We will be using Apache Storm for the implementation which is a distributed stream processing framework in real-time. Apache Storm runs topologies cluster which have components such as spouts and bolts. The spout is a streamer task which makes sequence of tuples which is a data model of Storm and bolts are tasks for processing job.

Because of this it will speed up the processing by operating in parallel on the each node by distributing frames onto several computation and also it can reduce the overall computation load by dividing the frame into several region to detect on the each frame.

II. IDENTIFY, RESEARCH AND COLLECT IDEA

In India mostly, tolls are collected manually i.e. by registering car number whenever car passes through tolls. As various tags are not used such as RFID, it is very difficult to keep track of numbers of vehicles passed through the tolls. As numbers of vehicles are not tracked properly there is a high chance of fraud i.e. collected amount of money is shown less than the actual collected money. So our goal is to detect or keep track of vehicles passed through toll plaza for avoiding the fraud.

III. RELATED WORKS

Mostly, big data systems like Hadoop have been used on various fields. Hadoop is popular framework on the big data systems. Hadoop is quit slower compared to Apache Storm. so users have to process data more quickly. Therefore Apache Storm platforms appear to provide services which can process the big data in real-time. Apache Storm is a distributed real time stream data processing system, which has been used in Twitter for various critical computations. Hadoop ecosystem does not cover real-time stream processing. For this reason, Apache Storm appeared and has been a solution for real-time processing.

Apache Storm has various features. Firstly, it guarantees fault-tolerant and high availability. If errors occurs on the process works, it will reassign it immediately. Therefore, the works can be operated continuously. Secondly, its latency of process is short, because it does not save data and processes in real-time. Third, it is scalable i.e. it can add additional nodes on the Storm cluster easily. Finally, it guarantees reliable. Every data can be processed without any loss. It has one master node which is called Nimbus and one or more worker nodes which are called Supervisor. In addition, Storm relies on Zookeeper for managing nodes in Storm cluster. Zookeeper gives all information of supervisor's state to Nimbus. Then, Nimbus will assign works to supervisors and each supervisor processes assigned tasks.

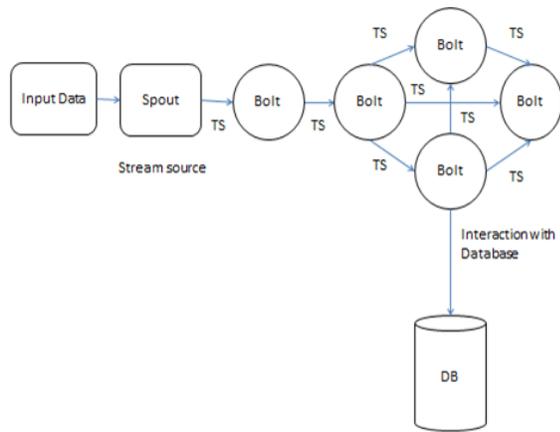


Fig.Core concept of Apache Storm

Vehicle detection:

Vehicle detection techniques are for tracking vehicles and keeping count of vehicles that passes through toll plaza. This technique is quit useful for avoiding fraud in amount toll tax collected and also providing security in apache storm by access right.

Numbers of CCTV cameras are installed in toll plaza. The frame rate of video is higher than before. The 30fps's CCTV translates 30 frames during 1 secondly.



Fig. Sample image for input

Each vehicle will be treated as an object by dividing image to frames and then frames to regions.

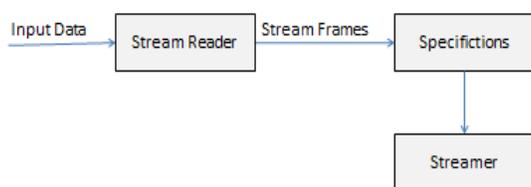


Fig.Data-flow diagram

The workflows are as follows:

- (1) Capture images: Capture servers captures an image frame from the provided videos including streaming video channel and VOD data.
- (2) Input the captured image to message queue: The captured image is sent to the message queue. Queue maintains feeds of messages including images until spout consumes.
- (3) Spout: Spout consumes messages from the queue, and passing them to Bolts that will be in charge of the processing.
- (4) Bolt: Bolts are small pieces of programs which process tuple stream. Complex stream transformations require multiple steps and thus multiple bolts. In this system, object detection can be decomposed into smaller subtasks(bolts) and executed in parallel on nodes. A face detecting bolt, an eye detecting bolt, and a nose detecting bolt are processed simultaneously. Object detecting is the most time-consuming part of the whole system and is implemented by Storm bolt.

Output results: Results from all object detecting bolts will be collected and all parallel processing output will be stored in databases.

IV. CONCLUSION

In this paper, we have presented the vehicle detection methodology in the distributed real time environment using Apache Storm framework. The image processing in the distributed environment is not appropriate due to the network load problem caused by communications between nodes for translating frames. Therefore, we cannot use the distributed processing since only one node cannot cope with the processing high resolution and high fps image data stream in real-time. Therefore we use Apache Storm for real time distributed environment.

REFERENCES

- [1] Im,Dong-Hyuck, Cheol-Hye Cho, and IIGu Jung. "Detecting a large number of objects in real-time using apache storm."Information and Communication Technology Convergence (ICTC), 2014 International Conference on.IEEE, 2014.
- [2] Dean, Jeffrey, and Sanjay Ghemawat. "MapReduce: simplified data processing on large clusters." Communications of the ACM 51.1 (2008): 107-113.

- [3] Ting Liu, Charles Rosenberg, Henry A. Rowley, "Clustering Billions of Images with Large Scale Nearest Neighbor Search," IEEE Workshop on Applications of Computer Vision, pp. 28, 2007.

- [4] Kothiya, Shraddha V., and Kinjal B. Mistree. "A review on real time object tracking in video sequences." Electrical, Electronics, Signals, Communication and Optimization (EESCO), 2015 International Conference on.IEEE, 2015.

- [5] Paul Viola, Michael J. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," IEEE Conference on Vision and Pattern Recognition, issue. 1, pp. 511-518, 2001.

- [6] Brian O'Neill, P. Taylor Goetz, Storm Blueprints: Patterns for Distributed Real-time Computation. Packt Publishing, 2014.