International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Vehicle Detection in Traffic Monitoring

Sakshi Shankad[1] , Prof.Pramod Jadhao[2]

[1]*Dept of MCA-Trinity Academy of Engineering, Pune, India*

[2]*Assitant Professor, Trinity Academy of Engineering, Pune, India*

Abstract

Traffic Monitoring is a challenging task on crowded roads. Traffic Monitoring procedures are manual, expensive, time consuming and involve human operators. Large-scale storage and analysis of video streams were not possible due to limited availability. However, it is now possible to implement object detection and tracking, behavioral analysis of traffic patterns, number plate recognition and surveillance on video streams produced by traffic monitoring. In Big data, video datasets are so large that typical database systems are not able to store and analyses the datasets. Storing and Processing big volumes of data requires Scalability, Fault Tolerance and Availability. Thus, Big Data and Cloud computing are two compatible concepts as cloud enables big data for traffic monitoring. In this paper, we proposed vehicle detection for traffic monitoring. We have applied Support Vector Machine (SVM) machine learning algorithm for detecting vehicles.

• INTRODUCTION

Traffic jams, congestion, and accidents on city roads is a common problem in most major cities across the world. Traffic monitoring systems have the capability to capture and transmit number of vehicles that pass through an intersection as a function of time intervals, and average speed of vehicles. Most cities have digital video cameras installed in hundreds of locations primarily for monitoring crime and terrorist activities. They generate video data per day at the scale of terabytes. Issues involved include efficient and secure transmission and storage, processing and feature extraction, storage and retrieval of features, and performing analytics on feature data. Analytics reveal traffic patterns keyed to geographic location and time intervals, congestion and accident reports.

Traffic Monitoring can capture, process, store, analyze, and retrieve video data at scale. Also detects and tracks individual vehicles in the video frames and computes total number of vehicles that have passed through an intersection over a time interval. It also computes the speed of individual vehicles and average speed of vehicles. In real-time additional functionality of the system includes suggesting alternative routes to commuters when congestion spotted on roadways.

Traffic Monitoring System used for video analysis process as shown in the Fig 1. After splitting the video into chunks, vehicles are detected from the chunks. Vehicle detection is not a trivial task and is performed in two stages.

- First, Haar classifiers are used for pre-detecting vehicles in the video frames. This is a pre-processing step.
- Second, Support Vector Machine (SVM) is use to accurately detect the presence of vehicles. This is also referred as post-processing step.

Purpose of this research is to build a robust and high throughput cloud computing based solution for automatic analysis of video streams coming from traffic monitoring cameras and recorded in a cloud based storage. The term video analytics refers to the processing and analysis of video streams using computing resources

Traffic video analysis aims at using a number of techniques to achieve better traffic and road safety, control congestion and provide immediate care for accident victims. The system alerts nearby hospitals and highway rescue teams when accidents occur. It also detects road congestion and broadcasts alternative route information to relevant computers. Flow Diagram of video data processing is shown in the Fig 2. Here data is divided mainly 3 parts Primary Data, Derived Data and Interpreted Data [3]. Primary Data is collection of video data with feature tracking and camera parameters. Using Primary Data we can analyze road user trajectories with position x, y, z. Derived Data is derived from Primary Data, Using this data we can estimate speed and acceleration of the vehicles. Interpreted Data is use for clustering and correlation of the Derived Data. Therefore, we follow this process for video processing

- **LITERATURE SURVEY/BACKGROUND**

Introduction to Vehicle Detection: Start by introducing the importance of vehicle detection in traffic monitoring systems. Discuss the significance of accurate and efficient vehicle detection for traffic flow management, congestion detection, safety, and urban planning.

Traditional Approaches: Provide an overview of traditional methods used for vehicle detection, such as feature-based approaches, template matching, and background subtraction. Discuss their limitations in terms of accuracy, robustness, and scalability.

Machine Learning-Based Approaches: Explore the transition towards machine learning-based approaches for vehicle detection. Discuss the advantages of using techniques such as deep learning, convolutional neural networks (CNNs), and object detection algorithms (e.g., YOLO, SSD, Faster R-CNN) for improving detection accuracy and robustness.

Dataset and Benchmarks: Discuss the importance of datasets and benchmarks in training and evaluating vehicle detection algorithms. Highlight widely used datasets in the field, such as KITTI, Cityscapes, and the Caltech Pedestrian Dataset, and explain their role in advancing research in this area.

Challenges and Limitations: Address the challenges and limitations associated with vehicle detection in traffic monitoring, such as variations in lighting conditions, occlusions, varying vehicle sizes and shapes, and real-time processing requirements. Discuss how these challenges have motivated the development of new techniques and algorithms.

Recent Advances: Highlight recent advancements in vehicle detection techniques, including multi-sensor fusion, attention mechanisms, domain adaptation, and context-aware detection. Discuss how these advancements address existing limitations and improve the performance of vehicle detection systems.

Applications and Impact: Explore the diverse applications of vehicle detection beyond traffic monitoring, such as autonomous driving, smart transportation systems, and vehicle tracking for law enforcement and security purposes. Discuss the potential societal and economic impact of advanced vehicle detection technologies.

Research Gaps and Motivation: Identify gaps in the existing literature and motivate the need for your research. Explain how your proposed approach or methodology addresses these gaps and contributes to advancing the field of vehicle detection in traffic monitoring.

- **PROPOSED WORK/SYSTEM**

Procedural work is used in OpenCV using video for Traffic Flow Vehicle Detection for Detecting vehicle image with color, speed, Direction and size of vehicle shown in Fig 5.When we use a video in OpenCV Python that is divided video in to some frames and after that TensorFlow Object Detection API apply on this frames. The TensorFlow Object Detection API is an open source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models. There are three modules:

Color Recognition Module: This module is detected vehicle using color recognition by KNN trained with color histogram for identify vehicle color in the frame.

Speed and Direction Prediction Module: This module is detected vehicle image pixel locations for prediction the speed and direction of the vehicle by pixel locations.

All steps are using with SVM (Support Vector Machine) Algorithm:

```
feature params=
{
'color model‟ : yuv',                    # hls, hsv, yuv, ycrcb
'bounding box size‟:64',                 # 64 pixels x 64 pixel image
'number of orientations': 11,            # 6 – 12
'pixels per cell': 16,                   # 8, 16
'cells per block': 2,                    # 1, 2
'do transform sqrt': True
}
src = FeatureSourcer(feature params, temo frame)
cls = BinaryClassifier(svc, scaler)
slider = Slider(source = src, classifier = cls, increment = 8)
this heatmap = HeatMap(frame = temp frame, thresh = 30, memory = 40)
window sizes = 80, 120, 150, 180
strip positions = 410, 390, 380, 380
def pipeline(this frame):
for sz, pos in zip(window_sizes, strip_positions):
        bounding boxes = slider.locate(frame = this_frame, window size = sz,
                window position = pos)
            this_heatmap.update(bounding_boxes)

heatmap, threshold_map, labeled_map = this_heatmap.get()
```

labeled_frame = this_heatmap.draw(thus_frame)

return labeled frame

This Algorithm is use for all feature parameters like color model, bounding box size, binary classifier, heat map. We can change parameter as per our analysis. In addition, we change sliding window size as per our vehicle classification and strip position is use for increasing accuracy in the frame. Heat map is use for creating bounding boxes for locating detected vehicles. End of the coding we have a labeled frame as a output.

- **RESULT AND DISCUSSIONS**

The results and discussions section of a vehicle detection system in traffic monitoring would typically involve an analysis and interpretation of the outcomes obtained from implementing the detection algorithm.

- Performance Metrics Evaluation
- Detection Accuracy
- Comparison with Existing Methods
- Robustness to Variations
- Real-time Performance
- Case Studies or Examples
- Limitations and Future Directions
- Impact and Applications

- **CONCLUSION**

An analytical solution is provided by us for road traffic video using OpenCV Python. We have demonstrated the scalability of the system. Here we use SVM classifier in entire process and generate all features. A deep learning approach would allow the classifier to learn some condition with different scaling. The advantage of SVM classifier is that it‟s quite fast to train, doesn‟t require big volume of data like a deep neural network and it‟s easy to implement. In future, the system performance will be analyzed by incorporating more nodes. Live analysis of video data is a task operating on a stream of data. OpenCV is intended for batch processing of large volumes of data. To support real time stream computing, We plan to work on enhancing high-level event recognition and prediction as well as classifying vehicles. We will also investigate and validate the relationship between collision probability and safety.

There are many open security problems like storage and high security issues with confidential data, which need to be addressed in the cloud. We also use a deep neural network for any future work on vehicle detection because it would be a lot more accurate and would not have issues like in SVM. Such an approach would also be potentially faster in real time processing.

REFERENCES

1. *Tariq Abdullah, Ashiq Anjum, M Fahim Tariq, Yusuf Baltaci , Nikos Antonopoulos"Traffic Monitoring Using Video Analytics in Cloud " 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, pp. 39 – 48, 2014.*

2. *Zhou Lin, Li Zhen, ChenYingmei, Tan Yuqin "The Video Monitoring System Based on Big Data Processing" IEEE - 2014 7th International Conference on Intelligent Computation Technology and Automation, pp. 865 – 868, 2014.*

3. *Xiaomeng Zhao, Huadong Ma, Haitao Zhang, Yi Tang, Guangping Fu "Metadata Extraction and Correction for Large-Scale Traffic Surveillance Videos" 2014 IEEE International Conference on Big Data, pp. 412 – 420, 2014.*

4. *Mazhar Hameed, Hiba Khalid, Dr. Farooque Azam "Big Data: Mathematical Topology Video Data Analytics using Superimposed Learning" IEEE- 2015 International Conference on Machine Learning, pp. 133 – 140, 2015.*

5. *Mazhar Hameed, Hiba Khalid, Dr. Farooque Azam "Big Data: Mathematical Topology Video Data Analytics using Superimposed Learning" IEEE- 2015 International Conference on Machine Learning, pp. 133 – 140, 2015.*

6. *Mazhar Hameed, Hiba Khalid, Dr. Farooque Azam "Big Data: Mathematical Topology Video Data Analytics using Superimposed Learning" IEEE- 2015 International Conference on Machine Learning, pp. 133 – 140, 2015.*