Traffic Density Estimation for Traffic Management Applications Using Neural Networks

Manipriya Sankaranarayanan, Indian Institute of Information Technology, Sri City, India*
https://orcid.org/0000-0002-0973-2131

C. Mala, National Institute of Technology, Tiruchirappalli, India
https://orcid.org/0000-0001-8286-2568

Snigdha Jain, National Institute of Technology, Tiruchirappalli, India

ABSTRACT

Traffic density is one of the elemental variables used in molding road traffic kinetics. Current density estimation techniques include loop detectors and sensors which are dependent on the crowd-sourcing of traffic data, which suffers from limited coverage and high cost. This article proposes a unique method to estimate traffic density based on neural network and mathematical modelling which uses surveillance feed from cameras. The proposed method can save both transportation costs and journey time, thus helping in better traffic management. The result analysis shows that the proposed method works well for varying traffic flow conditions and dynamic conditions.

KEYWORDS

Neural Network, Traffic Densit, Traffic Flow Conditions y, Vehicle Detection

1. INTRODUCTION

Due to the step-up in the number of vehicles everyday, traffic over-crowding and jams have become quite common. Recently with the technological advancement in Intelligent Transport System (ITS), the transportation system authority can acquire information used in traffic engineering such as number of vehicles and speed of the vehicles (Takayuki, 2017). Traffic jams not only affect the routine lives of human but also lead to a rise in the cost of transportation. This makes developing an automated management system for traffic unavoidable. Traffic density estimation is the most important task that should be done for ITS. Traffic density is one of the elemental dealing's variables used in molding road traffic kinetics. It quantifies the number of vehicles is on route. It is the first harmonic building block for many traffic management applications like advising best itinerary to user based on stream

DOI: 10.4018/IJIIT.335494
*Corresponding Author

traffic conditions, live updates about congestion and traffic jams etc. Thus, the main aim of this work is on finding the density of traffic in a day-to-day scenario (Zhiming, 2018). To obtain information on the routine of vehicles and their speed through CCTV, the primary thing to be done is detection of vehicles. There are different methods that can be used for detection of vehicles. The application of image processing and computer vision techniques in detection of vehicles improves improves these strategies of traffic information assortment and road traffic observance (Girshick, 2014, Chen, 2007).

Traffic density can be estimated in three ways and are as follows:

1. **Sensor based method:** With the technological advances in microchip, RFID (Radio Frequency Identification) and inexpensive intelligent beacon sensing technologies, sensing systems for ITS have been extensively developed (Karandeep, 2017):
    a. **Inductive Loop Detector:** A prominent sensing technology currently being used for vehicle detection is induction loop. An induction or inductive loop is an electromagnetic communication or detection organization. It makes use of an oscillating magnet or moving current to stream electric current in a wire situated nearby. These loops can be placed below the bed of the road to find vehicles as they cross the loop's magnetic flux. The simple loop detectors just count the number of vehicles throughout a unit of your time that skip the closed circuit, whereas additional refined sensors estimate the appraisal of speed, category of vehicles and also the space between them. Loops may be placed in a single lane or across multiple lanes, and that they work with vehicles moving at variable speed (Karandeep, 2017).
    b. **Mobile Sensor:** Inductive sensors discussed above have limitations related to their installation, data collection etc. Using vehicles as sensors can remove these limitations as sensors are put in and maintained easily on vehicles than on the road and thus the traffic information is collected from everyplace (Zhiming, 2018).
2. **GPS based Methods:** Smartphones are now equipped with many sophisticated built-in sensors such as Global Positioning System (GPS) receivers, accelerometers, gyroscopes, cameras, and microphones. These sensors can be exploited to sense the traffic data. Without any new equipment, a person with a smartphone can turn any vehicle into a mobile traffic sensor (Takayuki, 2017).
3. **Computer Vision based method:** Here, camera feed from surveillance cameras placed at road junctions and Road Side Units (RSUs) are analysed by computer algorithms using either image processing or artificial intelligence-based techniques with respect to various traffic parameters like speed, density:
    a. **Image Processing Techniques:** The image processing technique used for vehicle tracking and counting is background subtraction. It is a motion-based image segmentation method (Karandeep, 2017).
    b. **Deep learning techniques:** After the advent of deep learning, accuracy for many computer vision tasks has increased by a large margin, especially for detection and classification. R-CNNs and You Only Look Once (YOLO) are some important state-of-the-art detectors. These are real time object detection systems which can detect the bounding box coordinates of all the vehicles in a given image. Trade-off between the speed and accuracy can also be achieved by changing the size of the model. The detected image patches can be further classified into various classes like cars, trucks, motorcycles, etc using deep neural network classifiers like VGG16, ResNet, DenseNet, etc (Evan, 2018).

In this work, the detection of vehicle density is the main aim. To obtain this objective there are modules to be incorporated. The combination of the techniques used in these modules ensures the improved quality of traffic density estimation. The proposed techniques in this paper aids in improving the estimation of traffic density that in turn ensures the applications utilizing it becomes effective with improved quality.

2. RELATED WORKS

This section discusses about various image analysis techniques in estimation of traffic density.

(Karandeep, 2017) proposed estimation of traffic density using data provided by loop detectors. It uses lane-alteration effect for modelling of traffic and uses Markov chain into the state area model to explain the lane-alteration. It uses Kalman filtering method for estimation of traffic. However, this method requires high infrastructure support and regular object tracking.

(Zhiming, 2018) proposed the traffic estimation in case of low frame rate. Here different convolution neural networks are used to segment the image and learn the features specific to a particular image. Traffic images are classified into empty image, fluid image, heavy image and jammed image. The learned features from the traffic are transferred to the new traffic scenes. It takes use of repetitive features of traffic frames. This method does not require regular object tracking but the computation involved to learn features about vehicles becomes a bottleneck here.

(Takayuki, 2017) proposed a speed estimation approach from time-based sequences of vehicle. This method does not need vehicle tracking using any tagged information. It measures traffic velocities with less expensive detectors like web cameras. These velocities are used to measure density using vehicle counts. This method observes traffic density at specific time intervals. If the time intervals for observation increases the accuracy of this method decreases.

(Evan, 2018) combined AutoClass with Hidden Markov Models (HMMs) to estimate density of road traffic in a Region Of Interest (ROI) on a particular lane in a captured traffic video. This involves extracting low-level features and using AutoClass on these low-level features to get a unsupervised clusters for the different states of traffic density. For each density state four models of HMM are created. However, this method is highly manual since it requires labeling of the sequences of traffic.

(Girshick, 2014) proposes an algorithm for object detection that improvises mean Average Precision (mAP) by more than 30\% compared to the best result on VOC dataset from 2012. This approach is scalable and uses Convolutional Neural Networks (CNNs) to partition the objects in a bottom to top approach. It combines deep learning along with computer vision but training the neural network becomes very difficult in case of real time.

(He, 2016) eases the training of neural network by using a reformulated learning framework. The network layers are rearranged and mapped with the input layers with respect to residual functions. There has been comprehensive proof showing that these networks have easier optimization at high accuracy. This method requires a high computation time.

(Xie, 2017) presents a method for image classification using a modularized architecture. The network is built by repetition of a building block that aggregates with an equivalent topology a set of transformations. This leads to a solid, multi-branch design that has solely a couple of hyper-parameters to line.

(Girshick, 2015) presents a Region Proposal Network (RPN). This network has the same features with the detection network. This results in extremely low-cost region proposals. The RPN network generates high-quality region proposals due to training. These region proposals are used by Fast R-CNN for detection. This method requires training the network over many images to give high accuracy.

(Papandreou, 2017) proposes a method for multi-person detection and 2-D key point localization (human pose estimation) that achieves state-of-the-art results on the challenging COCO key points task. It is a top-down approach consisting of two stages. In the first stage, the location and scale of boxes which are likely to contain people are predicted using the Faster RCNN detector with an Inception-ResNet architecture. In the second stage, the key points of the person potentially contained in each proposed bounding box are estimated. However, the accuracy of this method can be reduced by heavier occlusion and lower frame rate.

(Chen, 2007) proposes a method based on analysis of blobs for vehicle counting. The algorithm has five steps: background subtraction, detection, analysis, tracking and vehicle counting. A vehicle is modelled as a rectangular patch and classified via blob analysis. By analysing the blob of vehicles, the meaningful features are extracted. Tracking moving targets is achieved by comparing the extracted features and measuring the minimal distance between consecutive frame. This method can be improved by adding vehicle classification to improve the statistics

(Chen, 2007) proposes a method to count vehicles in night time environment using headlight information. The basic idea is to use variation ratio in color space to detect the ground- illumination resulted from the head-lighting of vehicle. Then, headlight classification provides the headlight information for determining the moving-object region and compensating pixels. Experimental results show that the proposed algorithm can detect vehicles and reduce both effects of ground-illumination and shadow. In the normal condition (non-crowding), the average accuracy can be raised near to 90\%. Even though this method incorporates classification, it is restricted to moderate traffic flow conditions.

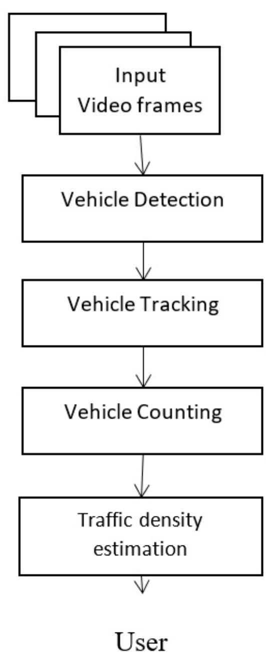
Considering the above factors, this paper proposes a novel method using neural network to estimate traffic density by analysing traffic surveillance video for single lane, multi-lane, junction traffic and various climatic conditions. The different distributions analysed are Kernel Density Estimation, Poisson Distribution and Gibbs Density Function. The analysis works well for all traffic flow conditions and climatic conditions.

3. PROPOSED METHODOLOGY

Estimation of traffic density is required for traffic controlling, routing, and scheduling of vehicular network traffic. The proposed system for estimation of traffic density is shown in Figure 1. This proposed system consists of four modules:

1. Vehicle detection
2. Vehicle tracking
3. Vehicle counting
4. Traffic Density Estimation

Figure 1. Proposed system for traffic density estimation



The input to the proposed system is the traffic surveillance video. The entire video is broken down into frames which is passed into the vehicle detection module.

3.1 Vehicle Detection

Vehicle detection is performed using Mask R-CNN, which is a conceptually simple and flexible framework for object-aware instance segmentation. Mask R-CNN adopts a two-stage procedure - Region Proposal Network (RPN) and Region of Interest (ROI) Pooling as shown in Figure 2 (Ren, 2019).

RPN takes the video frames as input and outputs the following for each region:

1. An "objectless" score for that region, which is how likely it is, for a video frame to contain a vehicle object.
2. Four coordinates representing the four corners of the input frame. Each frame of the video is given as an input to the Mask R-CNN detector. The detector does the following:
   a. Detects each vehicle in the image.
   b. Classifies it into classes – cars, trucks, motor-cycles etc.
   c. Gives bounding box for each detected vehicle.

The convolutional feature map is the output of one filter applied to the previous layer. A given filter is drawn across the entire previous layer, moved one pixel at a time. Each position results in an activation of the neuron and the output is collected in the feature map. To generate region proposals, a small network is applied over the convolutional feature map output by the last shared convolutional layer. This small network takes as input a n×n spatial window of the input convolutional feature map. Each sliding window is mapped to a lower-dimensional feature. This feature is fed into two neighbouring fully connected layers—a box-regression layer (reg) and a box-classification layer (cls) as shown in Figure 3 (Ren, 2015).

At each sliding-window location, multiple region proposals are predicted where the number of maximum possible proposals for each location is denoted as k. So, the reg layer has 4k outputs encoding the coordinates of k boxes, and the cls layer outputs 2k scores that estimate probability of object or not object for each proposal. The k proposals are parameterized relative to k reference

Figure 2. Architecture of mask R-CNN

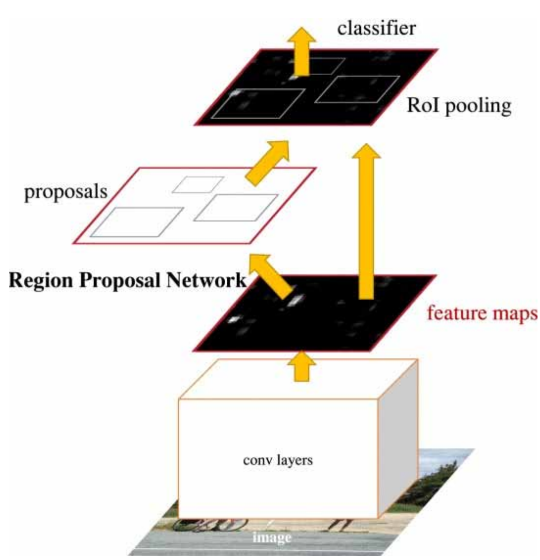
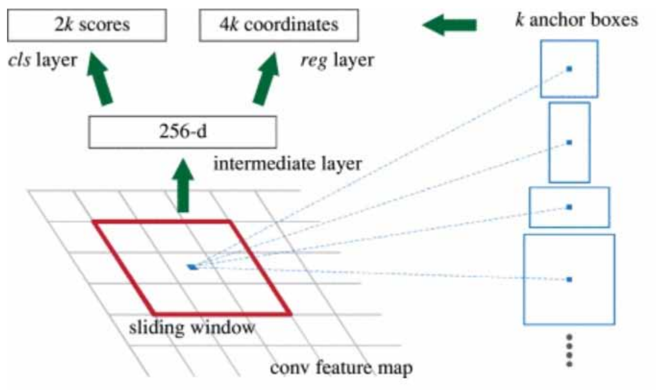


Figure 3. Architecture of region proposal network



boxes, which are called anchor boxes. An anchor is centred at the sliding window in question and is associated with a scale and aspect ratio.

3.2 Vehicle Tracking

Tracking-by-detection is a common approach to multi-object tracking. With ever increasing performances of object detectors, the basis for a tracker becomes much more reliable. In combination with commonly higher frame rates, this poses a shift in the challenges for a successful tracker. This shift enables the deployment of much simpler tracking algorithms which can compete with more sophisticated approaches at a fraction of the computational cost. The algorithm used for vehicle tracking is the Intersection over Union (IoU) algorithm (Bochinski, 2017). This algorithm evaluates how similar the predicted bounding box is to the ground truth bounding box. The IoU measure is defined as (Bochinski, 2017) in Equation 1:

$$IoU(A,B) = \frac{Area(A) \cup Area(B)}{Area(A) \cap Area(B)} \tag{1}$$

The Intersection over Union algorithm to find the overlap between two frames is shown in Algorithm 1.

Algorithm 1. Intersection over Union algorithm

```
Input: Co-ordinates of the vehicle detection block.
Output: Vehicle detection block co-ordinates.
  1. Take the bounding box co-ordinates of the vehicle as input
  from the previous module.
  2. Compare the IoU metric of the same area within the co-
  ordinates with respect to the previous frame.
  3. If the IoU metric exceeds a certain threshold then the
  vehicle is still in the video frame.
  4. Else, start a new vehicle track with the coordinates of the
  new bounding box.
  5. Pass the co-ordinates of the vehicle detection block to the
  vehicle counting module.
```

The overall complexity of the method is very low compared to other state-of-the-art trackers as it uses information on overlapping pixels of the consecutive video frames for vehicle detection. Hence it can be seen as a simple filtering procedure at detection level. This means if the tracker is used on-line in conjunction with a state-of-the-art detector, the computational cost compared to the detectors becomes negligible. Therefore, tracks can be obtained at virtually no additional computational cost from the detection.

3.3 Vehicle Counting

Vehicle counting involves drawing a detection line in the video frame. Detection line is a custom drawn viewpoint on the video frame which reaches the two ends of a single lane road along width of the road. A vehicle driving across the road will always pass through the detection line in the perspective of the image plane. Once, the vehicle crosses the detection line the count of the vehicle is increased by one. The vehicle count algorithm is given in Algorithm 2.

Algorithm 2. Vehicle count algorithm

```
Input: Vehicle detection box.
Output: Vehicle count and co-ordinates of each vehicle.
  1. Take the co-ordinates of the vehicle detection box from the
  vehicle tracking module.
  2. Set vehicle status as occupied.
  3. If vehicle’s bounding box does not cross the detection line
  goto step 1.
  4. Increase vehicle count by 1.
  5. Set vehicle status as released.
```

3.4 Traffic Density Estimation

Traffic flow and traffic density theories are the tools that help mathematicians and engineers to understand and express the properties of traffic flow. At any given instant, there might be thousands of vehicles on roadways throughout the year. There may be loss of an infinite number of hours on road as a result of traffic jams every year. Hence, to efficiently control traffic density analysis of data over a long period is done using mathematical models and traffic flow is predicted which in turn helps in routing decisions. Traffic density estimation is done with respect to different mathematical models such as Poisson Distribution, Kernel Density Estimation (KDE) and Gibbs Density Estimation.

3.4.1 Poisson Distribution

It is assumed that the traffic distribution is dilute with no disturbances and accidents. Then the number of arrivals in Poisson Distribution can be given by the following Equation 2 as specified in (Kayijuka, 2017):

$$p\left(x\right) = \frac{\mu^{x} e^{-\mu}}{x!} \tag{2}$$

where x is the number of vehicles passing the road during a predefined time interval. μ is the mean of the number of vehicles passing over the specific time interval.

3.4.2 Kernel Density Estimation

Kernel Density Estimation (KDE) is a non-parametric way to estimate the probability density function of a random variable. The aim of KDE is to find Probability Density Function (PDF) for a given dataset. Kernel Density Estimation smoothes the data estimation values around the PDF. The kernel density function (p(x)) is given by Equation 3 (Evan, 2018):

$$p(x) = \frac{1}{nh} \sum\nolimits_{i=1}^{n} \frac{K(x - x_i)}{h} \tag{3}$$

where h is a bandwidth, n is the number of data points, x_i stands for the density of vehicles at every data point i and K is the smoothening factor. When the dataset is finite KDE gives highly accurate results with a smooth curve.

3.4.3 Gibbs Density Estimation

Gibbs measure, named after Josiah Willard Gibbs, is a probability measure frequently seen in many problems of probability theory and statistical mechanics (Wei lu, 2009). Gibbs density estimation is more useful in case of an imbalanced data distribution. Since, the vehicle distribution on road is usually irregular, estimating density with respect to the Gibbs density function provides higher accuracy.

4. SIMULATION AND PERFORMANCE ANALYSIS

The proposed methodology is simulated and tested using deep learning and mathematical modelling.

4.1 Simulation Environment

The simulation environment for the proposed model is given in Table 1.

The screenshot of few of the datasets are shown in Figure 4. Various modules in the proposed model such as Vehicle Detection, Vehicle Tracking, Vehicle Counting and Traffic Density Estimation are tested for the benchmark dataset using deep learning and mathematical models.

LISA dataset consists of three colour video sequences captured at different times of the day and illumination settings with variations: morning, evening, sunny, cloudy, etc. Also, the videos are taken in different driving environments: highway and urban. KITTI is a Computer Vision benchmark suite used for object detection and object orientation estimation. It is collected using an autonomous fully calibrated driving platform. It consists of 7481 training images and 7518 test images, comprising a total of 80,256 labelled objects.

Table 1. Simulation environment for the proposed model

CPU	Intel i5
RAM	6GB
OS	Windows 7
Java	1.8.16
GPU	Nvidia Geforce MX150
Dataset	Microsoft COCO, LISA, KITTI, MIT traffic dataset

Figure 4. Screenshot of datasets

4.2 Vehicle Detection

Since deep learning-based detectors are used, training data is required to train the networks. For vehicle detection, the combination of Mask R-CNN and ResNet101 is first trained on Microsoft Common Objects in COntext (COCO) dataset. COCO is a large-scale object detection, segmentation, and captioning dataset on images. COCO has several features such as object segmentation, recognition in context, superpixel stuff segmentation, 330K images, 1.5 million object instances, 80 object categories, 91 stuff categories, 5 captions per image and 250,000 people with key points. After initial training of the network on COCO, additional training is done on Laboratory for Intelligent and Safe Automobiles (Andreas et al 2012) dataset and KITTI dataset (Jannik et al 2013).

Any object detection task is evaluated based on some standard evaluation metrics such as Average Precision (AP) and Average Recall (AR). Average Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. Recall is the ratio of correctly predicted positive observations to all observations in actual class (Renuka, 2016). Let tp be the number of true positives, tn be the number of true negatives, fp be the number of false positives and fn be the number of false negatives.

The Equation 4 and Equation 5 are given for AP and AR as specified in [20]:

$$Average\ Precision = \frac{t_p}{t_p + f_p} \tag{4}$$

$$Average\ Recall = \frac{t_p}{t_p + f_n} \tag{5}$$

Average precision and average recall are calculated for the combination of Faster-RCNN and ResNet101 method with respect to the combination of Mask-RCNN and ResNet101 method. Average precision and average recall are higher for the combination of Mask-RCNN and ResNet101 method. The comparison is shown in Table 2.

4.3 Vehicle Counting

The proposed model is tested on camera feed from surveillance cameras placed at various junctions and RSUs. Density estimation was collected, tested and analysed under four different scenarios:

1. Single lane (urban scenario)
2. Multilane (urban scenario)
3. Intersection (urban scenario)
4. Different climatic conditions (highway scenario)

Video surveillance is taken from multiple junctions in MIT Traffic dataset. Traffic video sequences are of 90 minutes each. The size of the scene is 720 by 480 with twenty clips which are merged and split into eight video sequences. Thus, these eight clips are used to simulate the camera feeds from 8 locations.

Scenario 1: Single Lane and urban scenario

Figure 5 is used for vehicle counting in single lane and urban scenario. This scenario represents vehicle counting in cities where the vehicular traffic is high, vehicles move in one direction only at low speed.

Scenario 2: Intersection and Urban Scenario

Table 2. Average precision and recall for vehicle detection

	Average Precision	Average Recall
Faster RCNN+ResNet101	82%	65%
Mask RCNN+ResNet101	84%	69%

Figure 5. Vehicle counting in single lane and urban scenario

Figure 6 is used for vehicle counting in an intersection and urban scenario. This scenario represents vehicle counting in cities where the vehicular traffic is high, vehicle move in more than one direction at low speed.

Scenario 3: Rainy Climate and Highway Scenario

Figure 7 is used for vehicle counting in rainy climate and highway scenario. This scenario represents vehicle counting in highways where the vehicular traffic is moderate, vehicle move in one direction only at high speed.

In all three cases the vehicles are first identified using Mask R-CNN. The mean and covariance of the total vehicle count over a specific time interval, the data points of the vehicle and the respective

Figure 6. Vehicle counting in an intersection and urban scenario

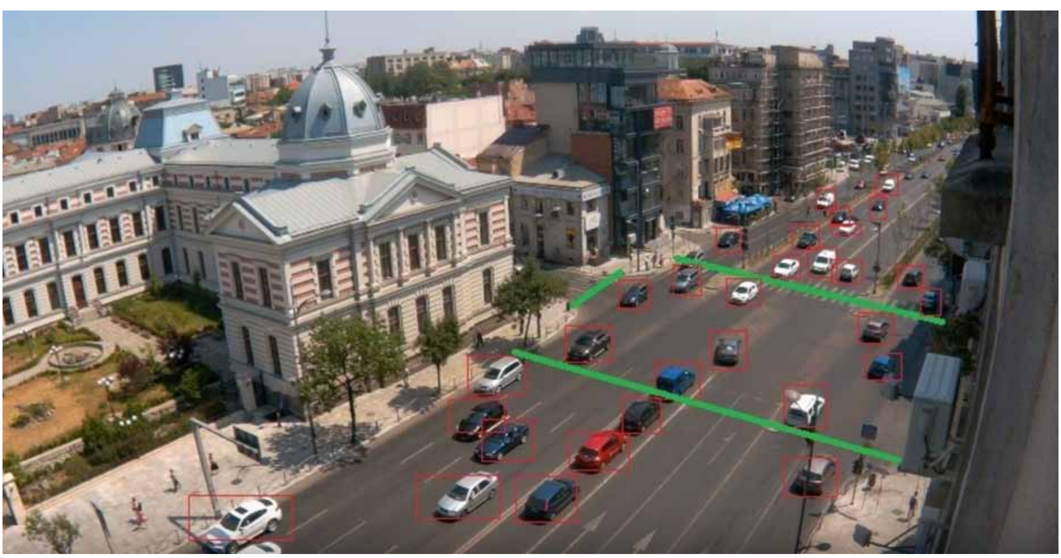


Figure 7. Vehicle counting in rainy climate and highway scenario

times are stored in a CSV file. Then the traffic density is estimated by using these values as input to the mathematical models.

4.4 Traffic Density Estimation

In mathematics and civil engineering, traffic flow is the study of interactions between traveller and infrastructure, with the aim of understanding and developing an optimal transport network with efficient movement of traffic and minimal traffic congestion problems.

4.4.1 Poisson Distribution

Traffic density is estimated using Poisson distribution for single lane. This distribution has been used since it gives a good distribution when the number of vehicles passing through the road is large over a fixed interval of time. This has been shown in Figure 8.

From the graph it is inferred that the distribution of the vehicles is reported accurately when the time period of the observation is high. The count of the cars crossing the detection line is obtained from the vehicle counting model. This is averaged over the entire time duration to get the mean car count.

4.4.2 Kernel Density Estimation

The mean car count of the vehicles is plotted with respect to time using the Kernel Density Estimation for single lane. Kernel density estimation is more useful in case of a finite data sample as it helps to give a smooth curve representing the data distribution This has been shown in Figure 9. The graph is smoothed to give a continuous function.

From the graph it can be inferred that this method gives a smooth distribution curve over a small period of time The video was tested for 14 seconds and it shows accurate results when the traffic density increased from 5 seconds to 9 seconds interval.

Figure 8. Traffic density vs. time using poisson distribution for single lane

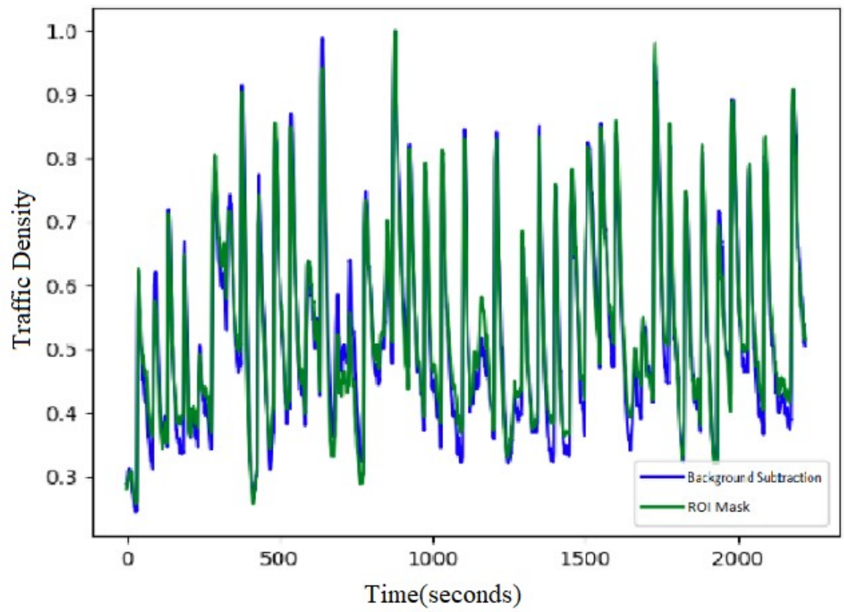
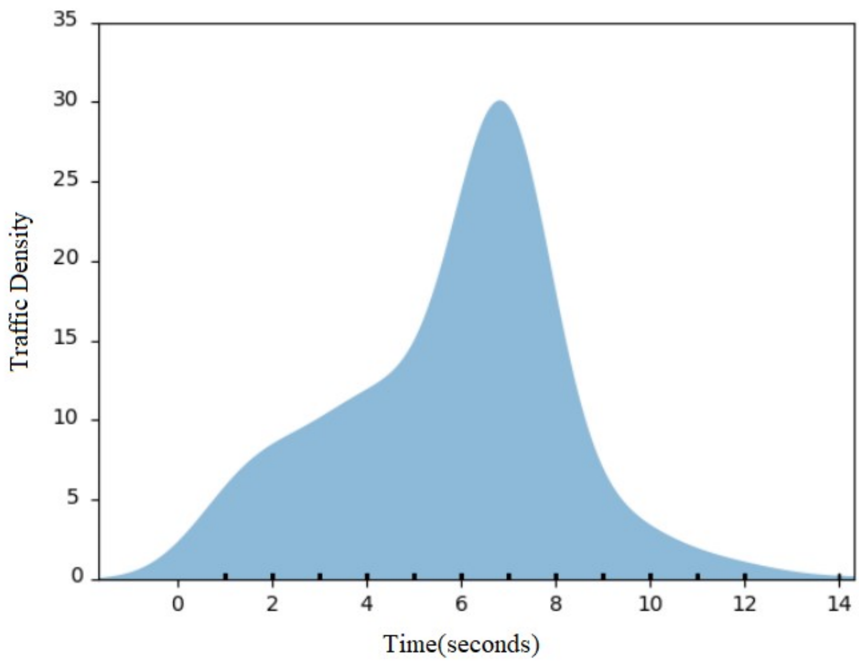


Figure 9. Traffic density vs. time using KDE for single lane



4.4.3 Gibbs Density Estimation

The traffic density is also estimated using Gibbs density function. This function is useful in case of imbalanced vehicle distribution on road. The mean car count obtained from Gibbs density estimation for single lane is plotted with respect to the time as shown in Figure 10.

From the graph it can be seen that the traffic density is estimated correctly in case of irregular traffic over a finite period of time.

Scenario 4: A comparison of traffic with respect to different time spans of the day, each span of 3 hrs is performed. Three roads with heavy traffic from MIT traffic dataset (MIT Traffic dataset, 2018) are chosen for analysis as shown in Figure 11.

Figure 10. Traffic density vs. time using Gibbs density function for single lane

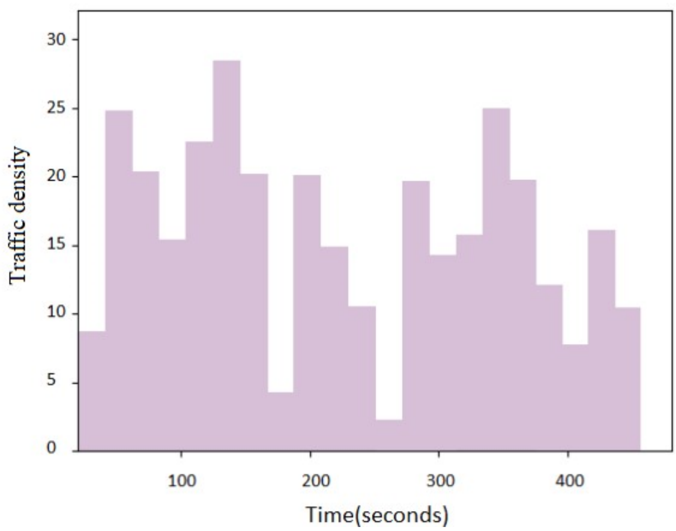


Figure 11. Sample video frame from MIT traffic dataset

The analysis is done for two consecutive days and averaged for 3-hour duration as shown in Figure 12.

On X-axis, time period is plotted in hours and on Y-axis, the frequency is plotted. For example, between 6 A.M. to 9 A.M., traffic density was updated for 3 times on an average. From the graph it can be seen that the traffic is highest between 6pm to 9pm followed by 9am to 12pm.

4.5 Application in Routing

Density estimation of traffic can be applied in routing where users can find the best possible path and estimated time to reach a destination from a given source. A small application has been developed on the same which depicts the travel time from source to destination via different routes and in case of different traffic distributions.

The input is taken from a benchmark dataset- Solomon Benchmark dataset (Soloman, 2019) which contains textual representation of traffic map of the road. The distribution is compared for increasing number of vehicles on road. The most optimal distance from source to destination is compared with respect to two different mathematical models: Gaussian Distribution and Poisson Distribution. The travel time is compared with respect to increasing number of nodes in topology from 50 to 950. The graph comparing the distance from source to destination for these mathematical models is given in Figure 13.

From the graph in Figure 13, it can be inferred that the distance from source to destination varies according to the position of the nodes. The optimal distance from source to destination are also compared with respect to three other types of distributions-Random Distribution, Clustered Distribution and combination of Random and Clustered distribution. The travel time is compared with respect to 15 different topologies each consisting of 100 nodes. The graph comparing the distance from source to destination for these three distributions is given in Figure 14.

From the graph in Figure 13, it can be inferred that the distance is highest for the combination of random and clustered nodes and least for clustered distribution.

Figure 12. Traffic density with respect to different spans of the day

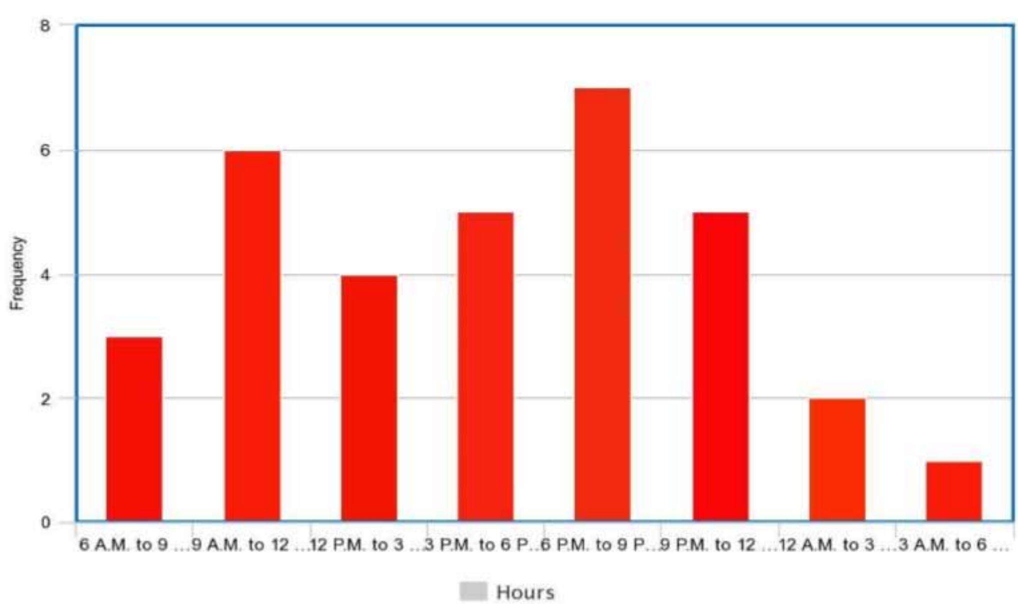


Figure 13. Comparing the distance from source to destination for two mathematical models

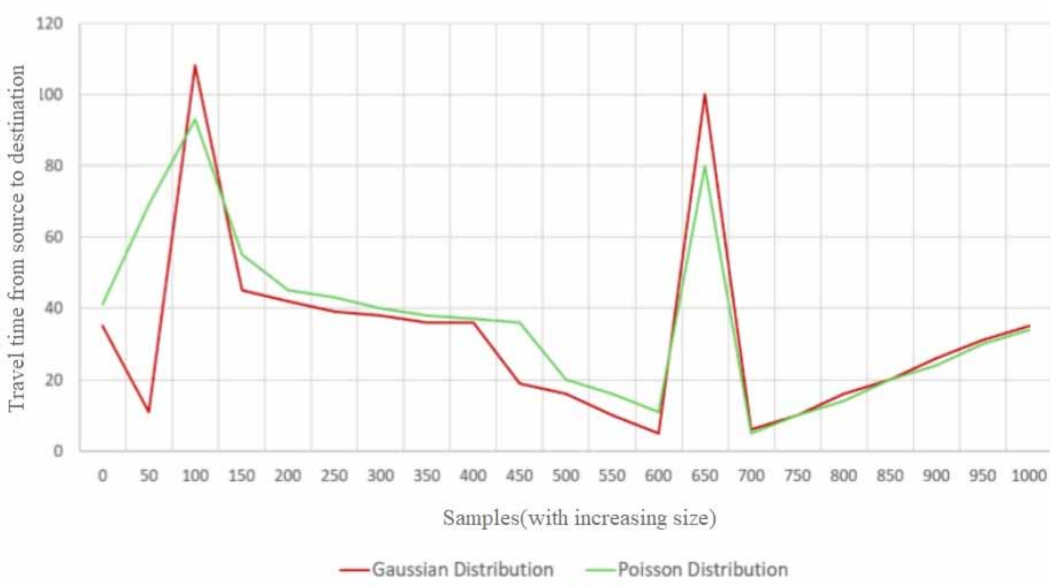
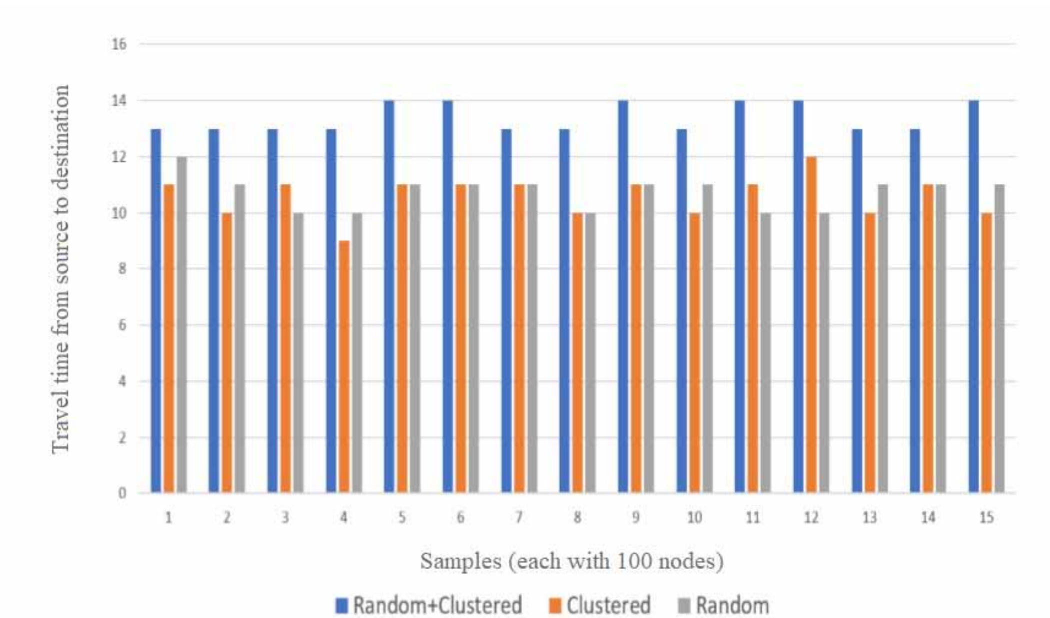


Figure 14. Graph comparing the distance from source to destination for three different distributions



Clearly, combination of Mask R-CNN and ResNet used in the proposed methodology has achieved higher average precision and average recall on the test data than the combination of Faster R-CNN and ResNet. Also, the mathematical models estimate traffic density accurately both in case of irregular distribution and finite datasets.

5. CONCLUSION

This paper proposed a method for estimation of traffic density as it forms a building block for controlling of traffic, routing and scheduling vehicular traffic. Density estimation is performed on mathematical models using neural networks. It has been further tested on benchmark datasets such as Microsoft COCO, LISA, KITTI and MIT traffic. The results show that the proposed method estimates traffic density accurately even with finite dataset.

REFERENCES

Bochinski, E., Eiselein, V., & Sikora, T. (2017, August). High-Speed tracking-bydetection without using image information. In *Advanced Video and Signal Based Surveillance (AVSS), 2017 14th IEEE International Conference on* (pp. 1-6). IEEE.

Chen, T. H., Chen, J. L., & Chen, C. H. (2007, November). Vehicle detection and counting by using headlight information in the dark environment. *In Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on* (Vol. 2, pp. 519-522). IEEE. doi:10.1109/IIH-MSP.2007.321

Chen, T. H., Lin, Y. F., & Chen, T. Y. (2007, September). Intelligent vehicle counting method based on blob analysis in traffic surveillance. In *Innovative Computing, Information and Control, 2007. ICICIC'07. Second International Conference on*(pp. 238-238). IEEE.

Fritsch, J., Kuehnl, T., & Geiger, A. (2013). A New Performance Measure and Evaluation Benchmark for Road Detection Algorithms. *International Conference on Intelligent Transportation Systems (ITSC).* IEEE. doi:10.1109/ITSC.2013.6728473

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587). IEEE. doi:10.1109/CVPR.2014.81

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017, October). Mask r-cnn. In Computer Vision (ICCV), 2017 IEEE International Conference on (pp. 2980-2988). IEEE. doi:10.1109/ICCV.2017.322

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). IEEE.

Idrissa, K. (2017). Mathematical Study for Traffic Flow and Traffic Density in Kigali Roads. *International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering Vol*, *11*(3).

Jensen, M. B., Philipsen, M. P., & Møgelmose, A. (2015). Vision for Looking at Traffic Lights: Issues, Survey, and Perspectives. In: IEEE Transactions on Intelligent Transportation Systems. IEEE.

Joshi, R. (2016). *Accuracy, Precision, Recall and F1 Score: Interpretation of Performance Measures*.

Katsuki, T., Morimura, T., & Inoue, M. (2017). Traffic Velocity Estimation From Vehicle Count Sequences. *IEEE Transactions on Intelligent Transportation Systems*, *18*(7), 1700–1712. doi:10.1109/TITS.2016.2628384

Liu, W. (2009). *Estimate density with Gibbs Sampling*. Utah University.

Luo, Z., Jodoin, P.-M., Su, S.-Z., Li, S.-Z., & Larochelle, H. (2018). Traffic Analytics With Low-Frame-Rate Videos. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(4), 878–891. doi:10.1109/TCSVT.2016.2632439

Møgelmose, A., Trivedi, M. M., & Moeslund, T. B. (2012). Vision based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey. *IEEE Transactions on Intelligent Transportation Systems*, *13*(4), 1484–1497. doi:10.1109/TITS.2012.2209421

Philipsen, M. P., Jensen, M. B., & Møgelmose, A. (2015). Thomas B Moeslund, and Mohan M Trivedi. "Learning Based Traffic Light Detection: Evaluation on Challenging Dataset". In: *18th IEEE Intelligent Transportation Systems Conference*. IEEE.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems (pp. 91-99).

Singh, K., & Li, B. (2017). Estimation of Traffic Densities for Multilane Roadways Using a Markov Model Approach. *IEEE Transactions on Industrial Electronics*, 59–61.

Tan, E., & Chen, J. (2017). Vehicular Traffic Density Estimation via statistical methods with automated state learning. 2017 IEEE Conference on Advanced Video and Signal Based Surveillance. IEEE.

Wang, X., Ma, X., & Grimson, W. E. L. (2009, March). Unsupervised Activity Perception in Crowded and Complicated Scenes Using Hierarchical Bayesian Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(3), 539–555. doi:10.1109/TPAMI.2008.87 PMID:19147880

Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017, July). Aggregated residual transformations for deep neural networks. *In Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on* (pp. 5987-5995). IEEE. doi:10.1109/CVPR.2017.634

Manipriya Sankaranarayanan is Assistant Professor in the Department of Computer science, Indian Institute of Information Technology Sri City Andhra Pradesh. Her research area includes Video Image Processing, Intelligent Transportation Systems and Vehicular Adhoc Networks.

C. Mala is a Professor in the Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli, Tamil Nadu, India – 620 015. Her research area of interest includes Data Structures & Algorithms, Computer Networks, Parallel Algorithms, Computer Architecture, Sensor Networks, Soft Computing Techniques, Image Processing, Intelligent Transportation Systems and Vehicular Adhoc Networks.