



Smart Traffic System with Green Time Optimization Using Fuzzy Logic

A. K. Kavin, R. Radha^(✉), Vishnu Prasad, and Bharathwaj Murali

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India
rradha@vit.ac.in

Abstract. Traffic Congestion has been a very big issue due to the large number of vehicles being utilized in cities. It is necessary to control the number of vehicles getting in and out of the crossing such that the city traffic congestion decreases in an efficient way. In this paper, we try to improvise the timings of green signals using fuzzy logic based on the number of vehicles, speed limit, and length of the crossing. We find out the number of vehicles using computer vision dynamically. We validate our system in two different locations in Chennai against the manual green signal duration that is set manually and we prove that our method improves the waiting time significantly.

Keywords: Deep Learning · Computer Vision · Fuzzy Logic · Internet of Things · Intelligent traffic management

1 Introduction

Traffic congestion on roads has been a huge issue in metropolitan cities ever since the popularity of motor vehicles has increased. Traffic congestion occurs when there are a large number of vehicles that are unable to drive to their respective destinations as the vehicles ahead of them have stopped. This is also called the saturation process. Situations such as accidents or sudden braking of a car in a smooth flow of heavy traffic, roadside constructions, traffic signals, and vehicles waiting to turn at an intersection can cause traffic congestion. In countries like India, there is an estimated annual loss of approximately Rs 60,000 crores due to congestion (fuel wastage, traffic maintenance, etc).

Traffic Congestion is very common due to the large number of vehicles manufactured and bought by the people. It's a challenge for controlling the vehicles in and out of the crossing in a way that overall traffic congestion decreases in an efficient way.

Existing traffic lights in India are manually operated and timers are set based on experience and history of the traffic. It needs to be replaced by smart traffic light control that is an active research and many solutions have already been proposed by researchers. We utilize sensors, computer vision and deep learning in our work. We are trying to prioritize moderate traffic conditions and provide a smart traffic lighting system in this work. The remaining parts of this article are arranged as follows. Section 2 elaborates

on the related works. Section 3 explains our idea and its implementation in detail. The experiments and the obtained results are explained in Sect. 4 and we conclude our work in Sect. 5. Future works explain how we can improve this project in the future with different ideas and approaches in Sect. 6.

2 Related Work

In [1], the authors analyze multiple factors to be considered for selecting optimal green time in managing the traffic. In [2], the authors find that start-up lost time needs to be taken into consideration which helps how quickly the vehicles move out of the intersection depending on the time in which the vehicles start to accelerate and move. Since this always takes some time to happen it is important to look into this parameter and decide the optimal value. This paper finds the optimal value by giving a brief analysis. It's important to take into account the green time optimization [3] parameter's, this paper considers the number of phases, the queue length of each phase, and the maximum queue acceptable for a given phase. Authors give an idea of the saturation headway (in seconds) [4] which is important to be known because it helps in estimating the capacity for example if the saturation headway is 4 s, authors say that 3600 s/h divided by 4 gives 900 vehicles per hour per lane.

Authors [5] have considered a new angle of approaching the green signal timing where they produced and named it effective green time based on some of the changes in the actual green lights, the changes in the actual yellow lights, and the changes in the actual red clearance lights which is subtracted by the start-up lost time and the clearance lost interval (Values are in seconds). In [6], The authors analyze the daily traffic volume at various mid-blocks in Chennai. The authors proposed a traffic light switching and traffic density calculation with the help of video processing over VANET. In [7], the authors have presented a system of taking pictures from the traffic pole with the empty road as a reference image. They also compute the volume of the traffic on the road and control the traffic in a smart way.

Authors designed a system [8] that the proposed system is implemented through IOT based protocols. With the help of an ultrasound sensor placed on the roadsides, they count the number of vehicles, and traffic lights are controlled accordingly. This article [9] has proposed an idea for smart traffic management implemented through a hybrid of a centralized system with a decentralized system focusing on future road planning and emergency reporting.

Authors improve the current traffic management system with the help of IoT-based concepts [10]. On each respective lane there are ultrasonic sensors attached. High sensor, Medium, sensor and Low sensor are the three types of sensors used. The priority order goes from high sending on first, then medium on second and finally low on third. The data is collected from the sensors and sent to the system. The traffic density is found, and the average waiting time is figured out. In this method [11], they are using RFID to compute the density of the traffic on both sides of the road. This will help in getting the direction of the vehicles and help to track them and also find the stolen vehicles. The traffic signal lights are changed based on the density of the traffic detected by the RFID placed on either side.

This paper [12] proposed to develop traffic signal timing plans. By using cluster analysis, the time was identified by analyzing the previously recorded data with data mining techniques. Authors of [13] propose implementing a method for detecting and analyzing the count of vehicles from an image input of the road. In this method, the proposed system takes a live video as input and does the necessary calculations using BeagleBoard and AVR microcontroller and the analysis used is (PCA) Principal Component Analysis. The DCSP method is used here to identify the duration of each traffic signal, which is based on the number of vehicles in each lane.

3 Proposed System

Architecture of the proposed system is depicted in Fig. 1. We dynamically calculate the duration of the green signal based on the number of vehicles, speed limit, crossing length and type of vehicles. Number of vehicles and its type are detected using CNN model YOLOv3 and other parameters can be extracted from the database based on the GPS position of the signal. Applying fuzzy logic for these input parameters help to find the optimized green time duration. Our model is meant for moderate traffic conditions of up to hundreds of vehicles in the signal.

3.1 Initialization Phase

To combat the mentioned issues, we propose a smart traffic management system that takes into consideration the various variables that affect the flow of traffic and propose a method by which can dynamically change the traffic lighting system to optimize the traffic signal in order to reduce the traffic congestion efficiently. To do this task we must first recognize the variables that can help us understand the situation on the road so that we can calculate and provide a proper green light time. The variables that we have taken into consideration are speed limit, the distance between two crossings, number of cars at a particular time, number of 2 or 3 wheelers at a particular time, and the number of large vehicles (buses and trucks) at a particular time. We will use these variables to determine the optimal green time.

During the implementation, the size of the road and speed limit are constants and can hence be manually added during the beginning of setup for the testing part, we have manually found the crossing length and speed limit of the traffic lights but for the large scale implementation, we plan to collect the speed limit from the government archives as we believe that the traffic police department would have these data at hand, for the crossing length we can use the GPS locations of two respective signals(which the vehicles will cross) to get the distance between them. These GPS locations can be obtained through government organizations or via manual labor. We use Object detection to get the number of vehicles and its type. It is enabled by computer vision technique which locates objects in the image and draw bounding boxes to perform object classification to find the label of the object. We use a near to State of the art technique which is made up of convolutional neural networks called “You only look Once” or simply known as YOLO. This standalone model can perform and produce real time results. Using the Keras deep learning library we used the third version of YOLO model for object

detection. The neural network architecture of YOLO should be robust that it can handle the day to day requirements. Robust object detection models make use of convolutional neural networks because its convolutional layers helps in retaining the image information without losing its quality. YOLOv3 uses a 53 layer convolutional neural network known as Darknet-53. This neural network is trained on the MSCOCO dataset which has 80 class labels including our required labels such as ‘car’, ‘motorbike’, ‘bus’, ‘truck’. There are several advantages in using a pre-trained model as they are easier to use and require almost no training We will be defining a model using Keras with the model architecture of DarkNet which has the required number and type of layers to match the downloaded YOLOv3 weights. By using the WeightReader function we can load and save the load the weights into our defined Keras model. Now we can make the prediction in input iamge get the count of number of cars, motorbikes, buses and trucks using our model to our main calculator, the calculator here implements the fuzzy logic method to predict the output (no of seconds of green light) and the microcontroller uses this to effectively to regulate the traffic lights and decrease congestion.

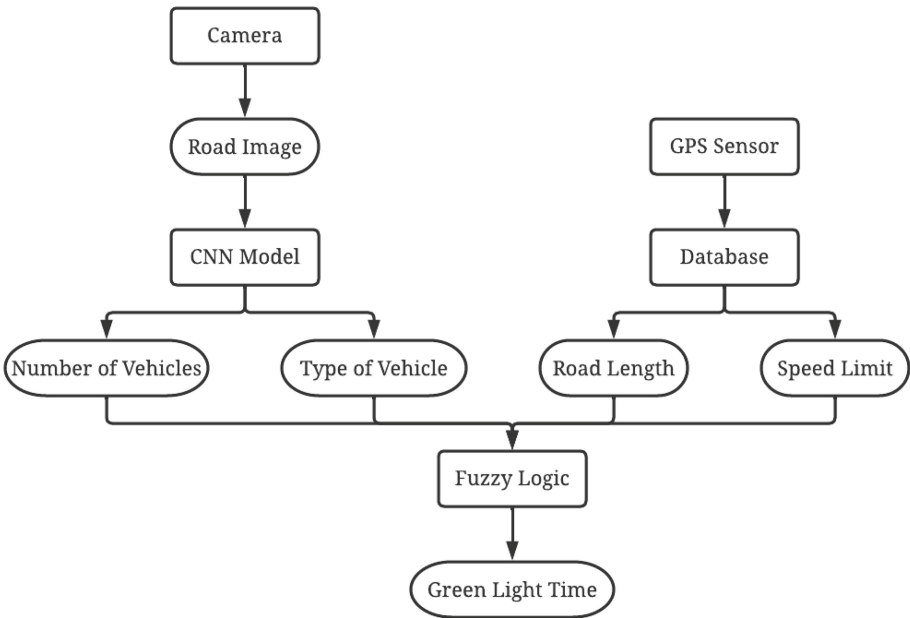


Fig. 1. System architecture

3.2 Fuzzifier

In 1975 Ebhasim mamdani proposed an fuzzy rule model know as Mamdani Fuzzy Inference System. We chose this over the Takagi-sugeno fuzzy model due to its simplicity and known to be well suited with human inputs. We represent our variables as fuzzy linguistic variables used to represent qualities spanning a particular spectrum which we

have taken into account as input variables are the number of two/three wheelers [low, medium, high] number of cars [low, medium, high], number of large vehicles [low, medium, high], the speed limit of the particular road [low, medium, high], the distance between two crossings [low, medium, high] are the input variables for the fuzzifier. In the case of the output variable, the fuzzy linguistic variable we have taken is the green signal time. We quantified the certainty of our input and output space of linguistic values (low, medium, high) and formed the membership functions. With the membership function formed we will be able to get the degree of truth or membership for every input variable.

For all input and output fuzzy variable values, the Triangular membership function has been defined. We have represented the yellow line as low value, green line as middle value, and red line as high value (Figs. 2, 3, 4, 5, 6, 7).

Membership function for number of cars

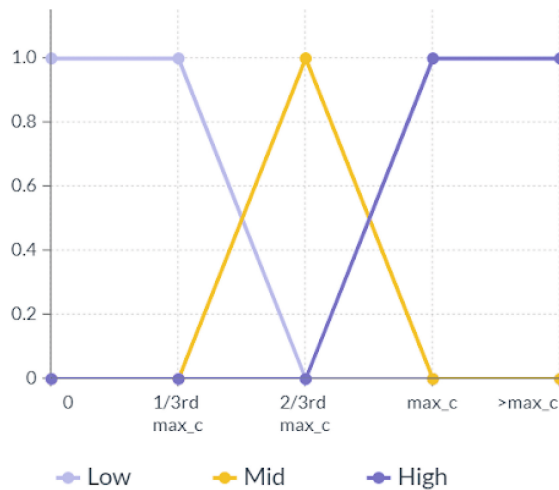


Fig. 2. Membership function for input parameter number of cars Vs the value of fuzzification. Up to $1/3\text{rd max}_c$ cars is considered (1.00 low, 0.00 mid and 0.00 high), $1/3\text{rd} - 2/3\text{rd}$ of max_c cars (1.00–0.00 low, 0.00–1.00 mid and 0.00 high) and similar values for $2/3\text{rd max}_c$ cars - max_c cars and $> \text{max}_c$ cars where max_c is the number of cars that can be passed through with maximum amount of green time

3.3 Fuzzy Rules

A predefined set of rules which are calculated based on the data we have obtained based on the existing auto-enabled traffic signals and analyzing them on the amount of time it usually takes for different amounts of vehicles to pass. Based on the rules we calculate the weights for each of the green light timing provided (20 s, 40 s, and 60 s) and obtain

Membership function for number of 2/3 wheelers

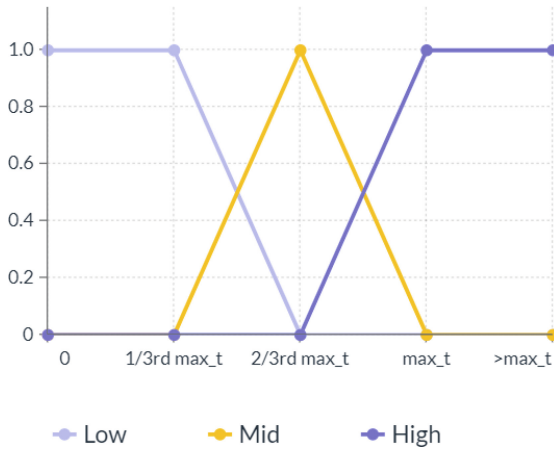


Fig. 3. Membership function for input parameter number of 2 or 3 wheelers Vs the value of fuzzification. Up to 1/3rd max_t of 2 or 3 wheelers is considered (1.00 low, 0.00 mid and 0.00 high), 1/3rd - 2/3rd of max_t of 2 or 3 wheelers (1.00–0.00 low, 0.00–1.00 mid and 0.00 high) and similar values for 2/3rd max_t of 2 or 3 wheelers - max_t of 2 or 3 wheelers cars and > max_t of 2 or 3 wheelers where max_t is the number of 2 or 3 wheelers that can be passed through with maximum amount of green time

the weighted average to calculate the final green light time. Here are a few rules we were able to build based on the reviews we have taken

- i) (high(speed limit) ∨ high(length between crossing)) ∨ (low(number of cars) ∨ low(number of bikes) ∨ low(number of large vehicles)) => low(green signal time)
- ii) (mid(speed limit) ∨ mid(length between crossing)) ∨ (mid(number of cars) ∨ mid(number of bikes) ∨ mid(number of large vehicles)) => mid(green signal time)
- iii) (low(speed limit) ∨ low(length between crossing)) ∨ (high(number of cars) ∨ high(number of bikes) ∨ high(number of large vehicles)) => high(green signal time)

3.4 Defuzzifier

Now we use the weighted average method to a crisp output value from the fuzzy values we have obtained from the fuzzy rules

$$y^* = \frac{\sum \mu(y).y}{\sum \mu(y)} \tag{1}$$

where, y is the variable with the corresponding membership function (20/40/60 s). μ(y) is the respective fuzzy value obtained from the rules y* is the final crisp output value (final green signal time in sec).

Membership function for number of Buses/Trucks

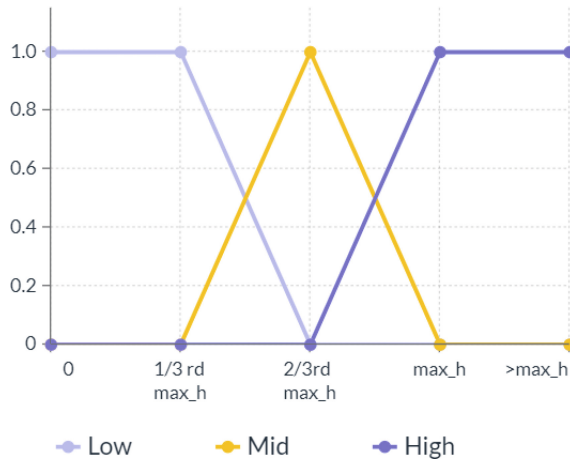


Fig. 4. Membership function for input parameter number of large vehicles V_s the value of fuzzification. Up to $1/3\text{rd max}_h$ large vehicles is considered (1.00 low, 0.00 mid and 0.00 high), $1/3\text{rd} - 2/3\text{rd}$ of max_h large vehicles (1.00–0.00 low, 0.00–1.00 mid and 0.00 high) and similar values for $2/3\text{rd max}_h$ large vehicles - max_h large vehicles and $> \text{max}_h$ large vehicles where max_h is the number of cars that can be passed through with maximum amount of green time

Membership function for speed limit

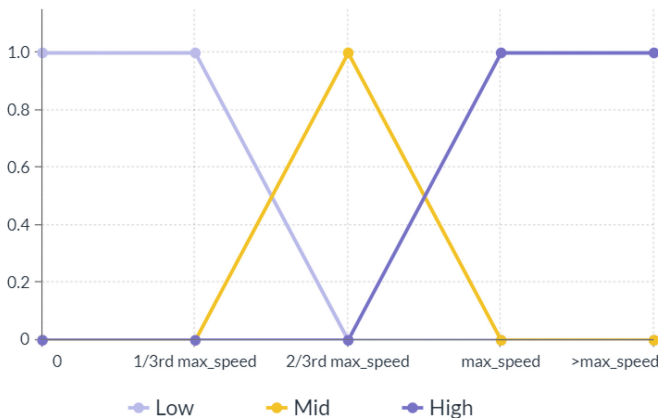


Fig. 5. Membership function for input parameter number of large speed limit V_s the value of fuzzification. Up to $1/3\text{rd max_speed}$ is considered (1.00 low, 0.00 mid and 0.00 high), $1/3\text{rd} - 2/3\text{rd}$ of max_speed (1.00–0.00 low, 0.00–1.00 mid and 0.00 high) and similar values for $2/3\text{rd max_speed} - \text{max_speed}$ and $> \text{max_speed}$ where max_speed is the maximum speed limit for which the system is applicable

Membership function for crossing length

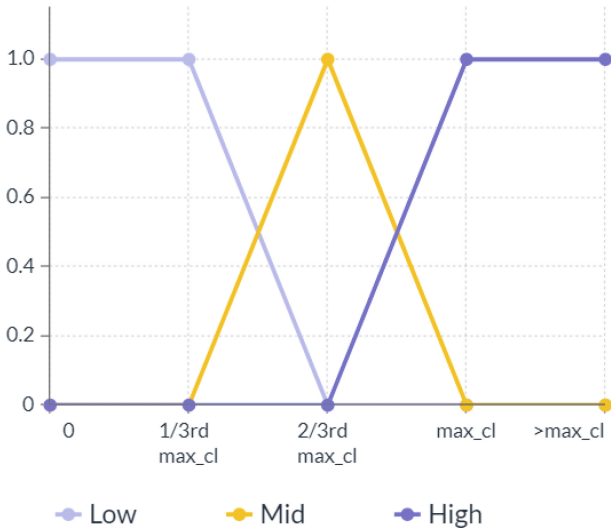


Fig. 6. Membership function for input parameter number of crossing length Vs the value of fuzzification. Up to 1/3rd max_cl is considered (1.00 low, 0.00 mid and 0.00 high), 1/3rd - 2/3rd of max_cl (1.00–0.00 low, 0.00–1.00 mid and 0.00 high) and similar values for 2/3rd max_cl – max_cl and > max_cl where max_cl is the maximum crossing length for which the system is applicable

Output function for green light time

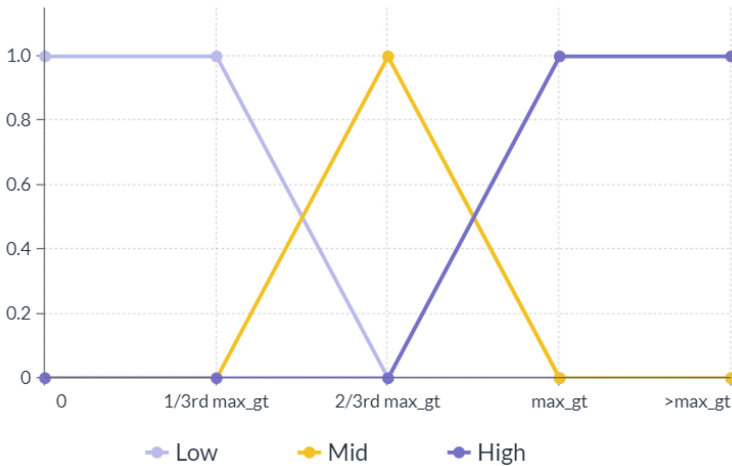


Fig. 7. Output Membership function for the output parameter (green light time) where max_gt is the maximum green time possible for the given traffic light

4 Experiments and Results

In order to understand how many seconds of green light is required for different situations so that we can design the rules for calculating the green light time, we have decided to go to various traffic signals and collect data regarding the number of vehicles of each type before the green light and after the green light in case of lesser than ideal green signal time or how much seconds were wasted in case of higher than ideal green signal and then compare it with the amount of green light time provided by our model (Table 1).

Table 1. Traffic Signal 1 green time duration

Bikes, Autos,etc.	Cars	Buses	Green signal time when set to automatic (seconds)	Actual time taken for vehicles to cross the crossing(seconds)	Vehicles didn't clear (Bikes, Cars, Buses)	Number of seconds our Proposed System gave
26	24	0	20	>20	11, 8, 0	34.3
11	18	3	20	>20	8, 4, 2	30
12	15	2	20	>20	9, 4, 1	30
23	16	2	20	>20	10, 8, 1	30
16	17	1	20	>20	7, 8, 1	30
19	24	1	20	>20	11, 8, 0	30
24	22	0	20	>20	10, 9, 0	30

The traffic signal 1 data is taken from the Tidel Park to Besant Nagar and Thiruvanmiyur signals. Here during the time in which we were taking the recordings, we noticed that the time given by the automatic signal was mostly not enough for the vehicles to cross through and hence we have taken the table details accordingly (Table 2).

The traffic signal 2 data is taken from Indira Nagar to Thiruvanmiyur. Here while we were surveying the vehicles and the time they have taken for each of them to pass, we have noted that there was a considerable amount of free time observed after the vehicles had passed that the green light was on.

Table 2. Traffic Signal 2 green time duration

Bikes, Autos, etc.	Cars	Buses	Green signal time when set to automatic (seconds)	Speed Limit (km/h)	Distance between two crossings (m)	Green Time in our proposed system (seconds)
8	5	0	40	45	10	30
5	2	0	40	45	10	30
9	4	0	40	45	10	30
3	1	0	40	45	10	30
8	3	1	40	45	10	30
4	1	1	40	45	10	30
10	5	0	40	45	10	30
2	0	0	40	45	10	30
12	4	2	40	45	10	30
4	3	0	40	45	10	30
5	4	1	40	45	10	30
1	0	0	40	45	10	30
8	2	2	40	45	10	30
4	2	0	40	45	10	30

5 Conclusion

Smart traffic systems are a step forward in making our environment clean and safe for human - life to exist. We are able to save many lives with this smart traffic system as we have trained the model in such a way that fewer false values are provided. The model is also much improved and better when including human interference. Our model was trained with parameters and we set 3 rules for the model to train with so that the model can understand the environment it is in and can proceed with the proper and necessary action. The number of cars present in the traffic, the number of cars left after the green light turns red and the amount of seconds the green light is provided for the vehicles to proceed are all important factors we have taken into consideration to achieve our goal.

6 Future Work

For future works, we feel we could make use of the synchronization process between two or more traffic lights within the same vicinity. We could include the detection of pedestrians for their safety while crossing the road as well as include an emergency option for ambulances as they need to quickly reach their destination when time is of the essence. We could improve the accuracy of the system, by adding more rules during the fuzzy logic implementation.

References

1. Hari Krishna, K., Vinay Kumar, K., Hanumantha Rao, Ch.: Signal Design Using Webster's Method (4 Legged Intersection), K L University, Green Fields, Vaddeswaram, College of Engineering, Osmania University, Hyderabad, Indian J. Sci. Res. **17**(2), 113–119 (2018)
2. Matsoukis, E., Efstathiadis, S.: An investigation of the variability of start-up lost times and departure headways at signalized intersections in urban areas. WIT Trans. Built Environ. **111**, 59–68 (2010). <https://doi.org/10.2495/UT100061>
3. Li, Y., Yu, L., Tao, S., Chen, K.: Multi-objective optimization of traffic signal timing for oversaturated intersection. Math. Probl. Eng. **2013**, 1–9 (2013). <https://doi.org/10.1155/2013/182643>
4. Signalized Intersections: Informational Guide, Published Date: 2004–08–01, Report Number : FHWA-HRT-04–091. <https://rosap.nrl.bts.gov/view/dot/39968>
5. Traffic Signal Timing Manual, U.S Department of Transport, Chapter 3. <https://ops.fhwa.dot.gov/publications/fhwahop08024/chapter3.htm>
6. Velmurugan, S., Reddy, T.S., Madhu, E., Rao, I.P.: Traffic operating characteristics and its impacts on air pollution in an urban area-a case study of Chennai, India, pp. 1799–1814 (2005)
7. Pandit, V., Doshi, J., Mehta, D., Mhatre, A., Janardhan, A.: Smart traffic control system using image processing. IJETTCS **3**(1) (2014)
8. Dubey, A., Lakhani, M., Dave, S., Patoliya, J.J.: Internet of Things based Adaptive Traffic Management System as a part of Intelligent Transportation System (ITS). IEEE (2018)
9. Javaid, S., Sufian, A., Pervaiz, S., Tanveer, M.: Smart Traffic Management System Using Internet of Things. IEEE (2018)
10. Shah, K.A., Jha, H.: Improvement of Traffic Monitoring System by Density and Flow Control for Indian Road System Using IoT
11. Chattaraj, A., Bansal, S., Chandra, A.: An Intelligent Traffic Control System Using Rfid
12. Zamani, Z., Saraee, M.M., Pourmand, M.: Application of Data Mining in Traffic Management: Case of City of Isfahan. IEEE (2010)
13. Rachmadi, M.F., Al Afif, F., Jatmiko, W., Mursanto, P., Manggala, E.A., Ma'sum, M.A.: Adaptive Traffic Signal Control System Using Camera Sensor and Embedded System. IEEE (2011)
14. Anurag, K., Ayush, S., Chetan, S.: Smart traffic lights switching and traffic density calculation using video processing, pp. 1–6 (2014). <https://doi.org/10.1109/RAECS.2014.6799542>
15. Bilal, G., Khaled, K., Khaled, C., Mohamad, K.: Smart traffic light control system, pp. 140–145 (2016). <https://doi.org/10.1109/EECEA.2016.7470780>