Intelligent Traffic Routing Based on Real-time Congestion Analysis

Argha Sen, Monsij Biswal, and Shreyan Datta

Department of Electronics and Communication Engineering,
National Institute of Technology, Durgapur, India

***Abstract*—The problem of traffic congestion has led to several issues like an increase in carbon dioxide emission including the inability to route emergency vehicles on a priority. Hence we aim to find an optimized approach, allocating resources based on the traffic condition at that particular instant. This could be realized by analysing the traffic data relayed by device nodes present all over the city in a real-time environment. The node is a cloud-linked camera-integrated MPU equipped with GSM communication. The captured video frames are processed which is then analysed collectively to suggest the most appropriate route to the vehicles.**

***Index Terms*—Traffic modelling, vehicle detection, computer vision, Internet of Things**

I. Introduction

The consistent increase in the number of vehicles has led to serious problems like traffic congestion and violations. The increase in traffic also adds to the declining air quality in metropolitan cities. As described in [1] the total vehicle count of New Delhi is estimated to rise to 10 million by 2020, resulting in an increased congestion. Furthermore according to [2], the annual cost of congestion in the United States is likely to increase from 124 billion US$ in 2014 to 186 billion US$ in 2030.

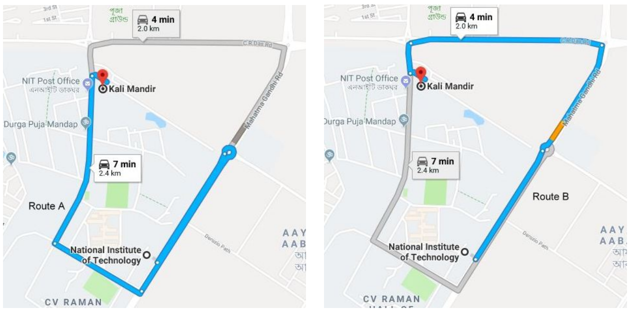
Several attempts have been made to detect and classify vehicles simultaneously.These are primarily based on magnetic sensors [3], microwave radars including laser sensors [4]. Fang et al. [5] show that vehicle detection can easily be realized using K-band unmodulated CW radar. This solution employs a K-band Gunn receiver together with a signal processing unit.Particularly, the radar is required to be at a certain height above the ground with a particular elevation angle and elevation beamwidth. The components used make it much inexpensive as compared to other approaches.

In this paper we have minimized the congestion by smartly allocating the incoming vehicles among narrow lanes thereby preventing disruption in traffic flow. Stating briefly, this paper has the following contributions:

- Prototype design and result of a camera-integrated traffic flow measurement node.
- An algorithm for resource-based allocation of the available routes to the incoming vehicles.

II. Problem Description

To reach a destination from a specific source point, multiple paths may exist. When exposed to multiple choices, it is desirable for the user to traverse the route which saves time, provided only the destination matters to the user and not anything in between.



(a) Route A (b) Route B

Fig. 1: Multiple routes to reach the same destination [Source: Google Maps]

Currently, Google Maps provide such insights to users based on the length and congestion of the routes as shown in figure 1. Therefore, we model our solution based on the number of vehicles in the i^{th} path n_i, which is directly proportional to congestion and its length L_i.

III. System Model

Our study consists of a source and destination point, which has multiple routes connecting them as shown in Figure 2. We first formulate the solution for the case of two routes and then generalize it for any number of routes. The task is to allocate the vehicles smartly so as to reduce congestion.

For this, we set up camera-integrated nodes at the two ends of the paths, to record the net traffic flow through the path at a given time. The net traffic flow is analyzed to find the threshold which determines which route is allocated to the next incoming vehicle. In our system, we have considered the routes to be of unequal length. Therefore, the resources are length of the lane L and the traffic density in that lane n/L.

IV. Architecture

We deploy nodes at the ends of the roads which measures the net flow of traffic through it for a given duration. We call these nodes, CamInt (camera-integrated) nodes. The count of the vehicles is then transmitted to the Centralized Control Unit

978-1-7281-2327-1/19/$31.00 ©2019 IEEE

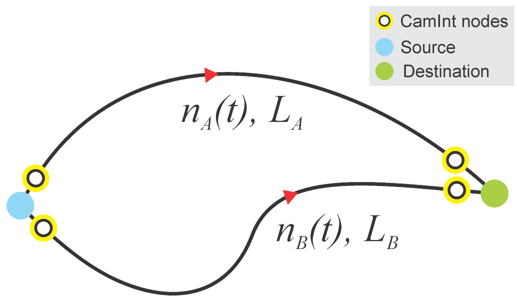


Fig. 2: Example Scenario

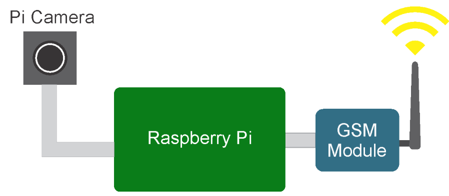


Fig. 4: Camera-integrated node - CamInt

where it is processed to determine the optimal route for the user.

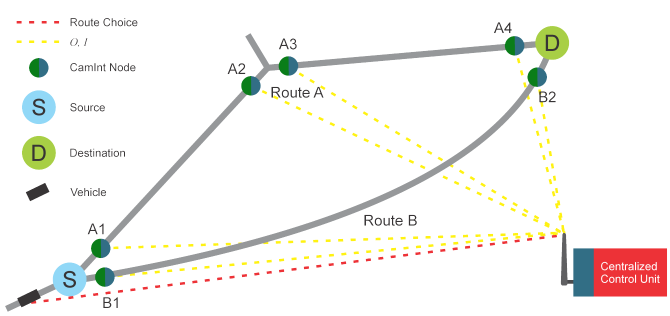


Fig. 3: Sample Network

A sample network is shown in Figure 3. Here there are two paths for an incoming vehicle to choose from, namely A and B. Route A consists of an diversion therefore it is broken into two sections. A pair of CamInt nodes are deployed at the end of each section.$(A1, A2), (A3, A4)$ are the CamInts for Route A whereas $(B1, B2)$ are employed for Route B. Each pair measures three key values - the time-stamp of vehicles entering the section I, the time-stamp of the vehicles exiting the section O and the length of the section L. The centralized control unit processes this information to evaluate the congestion in each path and provides an optimal solution to the user.

A. Camera Integrated Nodes

The CamInt or Camera-integrated nodes, as shown in Figure 4 are the sensor nodes which collects the information using a Pi camera.The processing unit (Raspberry Pi) utilizes this data to determine the count of the passing vehicles.This information along with the time-stamp and passed on to the centralized control unit through the GSM module.

V. Methodology

A. Vehicle Detection

1) Background Generation : Firstly, this algorithm takes image of the road without any vehicle as a background layer of the frame. After the background picture is available, Gaussian filters are applied to smooth the image and then converted to a grayscale image. The background b_0 is generated from the background frame b as:

$$b_0 = G(g(b)) \tag{1}$$

where $G(a)$ is the Gaussian filter and $g(a)$ represents the conversion to gray-scale, a being a two-dimensional image.

2) Background Subtraction: The moving objects in the frame are extracted as the foreground objects by subtracting the background frame from the current frame,b_0 being the background pixel values. This foreground image represents the moving objects in the current frame $f(t)$. Morphological transformation $M\{\}$ is applied to remove noises in the frame after the subtraction. The resultant image $r(t)$ is white for moving objects and black for the background.

$$r(t) = M\{g(f(t)) - b_0\} \tag{2}$$

Fig. 5: Background Subtraction

3) Binarization: Using thresholding, the image frame is converted into a binary frame by taking a certain threshold value Θ . As per equation (3), objects in the image frame with brightness above the threshold value are assigned a new brightness value of 255 whereas those having brightness below the threshold value are assigned a brightness of 0. Thus, the whole image frame consists of two brightness values [0,255]. This method is called binarization of the image frame.

$$r_0(t) = \frac{255}{2}(1 + sgn(r(t) - \Theta)) \tag{3}$$

4) Contour Detection: To detect the position of the vehicles in the image frame, contour finding technique is used. This provides us with the centroid coordinates of the moving object's contour. When the vehicle crosses a particular line

Fig. 6: Binarization

in the image frame, we increment the count of vehicles. This way we obtain the count of vehicles detected by a particular node.

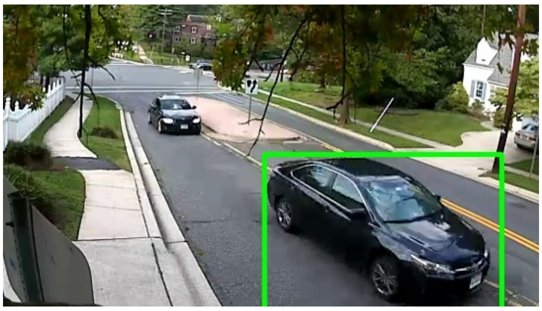


Fig. 7: Contour Detection

B. Communication

For transmitting sensor data, a digital cellular technology namely Global System for Mobile communications abbreviated as GSM is used which is embedded in the Raspberry Pi itself. The data transmitted here is the car density and not the image or video feed of the vehicles. Thus, the density being numbers are transmitted easily with a speed of 9.6 Kbit/s using GSM. GSM modem has bidirectional connection to the Raspberry Pi and this property is used for communication between user and system [6]. Here RS232 pin is used in the system. It performs serial communication between the microcontroller and the outside world. AT commands are used for interfacing GSM with Raspberry Pi.

Raspberry Pi with GSM module present in each CamInt node acts as a client and independently sends data to a single Raspberry Pi which acts as a server for the entire communication system. .

C. Traffic Congestion Estimation

We have the (CamInt) nodes at the two ends of the route sections. One detects the entry, the other exit. During any time interval, the total number of vehicles n in a section can be modelled as

$$n(t_s < t < t_e) = n_i(t < t_s) + |I(t_s < t < t_e)| - |O(t_s < t < t_e)| \quad (4)$$

where I and O are the set of in-time and out-time of vehicles respectively, $|X|$ denoting the size of the set X. Furthermore t_s corresponds to window start time, t_e to window end time and $n_i(t < t_s)$ represents all the vehicles inside the section before t_s. Typically t_s and t_e are separated by 2 minutes.

Equation (4) is an exact count of vehicles inside the section under consideration as $|O(t_s < t < t_e)|$ accounts for all the vehicles exiting the section irrespective of their time of entry.

D. Indicator Control

The threshold Φ can be easily calculated according to the idea of weighted average, the number of vehicles being the weight for each path:

$$\Phi = \frac{n_A L_A + n_B L_B}{L_A + L_B} \quad (5)$$

where n_A = total number of vehicles in path A, n_B = total number of vehicles in path B, L_A =length of path A, L_B = length of path B.

TABLE I: Indicator Control Output for different states

n_A			n_B			Indicator Control
>Φ	<Φ	=Φ	>Φ	<Φ	=Φ	
1	0	0	1	0	0	i | min(n_i/L_i) $\forall i \in \{A, B\}$
1	0	0	0	1	0	B
1	0	0	0	0	1	B
0	1	0	1	0	0	A
0	1	0	0	1	0	i | min(L_i) $\forall i \in \{A, B\}$
0	1	0	0	0	1	A
0	0	1	1	0	0	A
0	0	1	0	1	0	B
0	0	1	0	0	1	i | min(L_i) $\forall i \in \{A, B\}$

1) Generalizing the Solution: In Table I, we show the solution approach when there are two routes available. Now, we generalize the solution when we have m routes denoted by the set $X = 1, 2, 3, ..., m$. First, the threshold Φ is calculated as:

$$\Phi = \frac{\sum_{i=1}^{m} n_i L_i}{\sum_{i=1}^{m} L_i} \quad (6)$$

There are three cases, which are considered on a priority basis.

$$s_1 = i \quad | \quad \min(n_i) \quad \forall i \in \{x \quad | \quad n_x < \Phi \quad \forall x \in X\} \quad (7)$$

$$s_2 = i \quad | \quad \min(L_i) \quad \forall i \in \{x \quad | \quad n_x = \Phi \quad \forall x \in X\} \quad (8)$$

$$s_3 = i \quad | \quad \min(\frac{n_i}{L_i}) \quad \forall i \in \{x \quad | \quad n_x > \Phi \quad \forall x \in X\} \quad (9)$$

The solution is given by:

$$s_i \quad | \quad min(i) \quad \forall s_i \in \{k \quad | \quad k \neq \phi \quad \forall k \in S\} \quad (10)$$

where $S = \{s_1, s_2, s_3\}$

VI. CONCLUSION

In this paper, a new approach to efficiently route the incoming traffic at an intersection is proposed. Greedy based algorithm method ensures a minimum waiting time. Contour formation and routing are the two most important aspects in this regard. Thus continuous routing would not allow the traffic flow to get interrupted and accumulate thus resulting in better paced and managed traffic flow.

REFERENCES

[1] N. Davis, H. R. Joseph, G. Raina, and K. Jagannathan, "Congestion costs incurred on indian roads: A case study for new delhi," *arXiv preprint arXiv:1708.08984*, 2017.

[2] M. Carey and A. Srinivasan, "Externalities, average and marginal costs, and tolls on congested networks with time-varying flows," *Operations research*, vol. 41, no. 1, pp. 217–231, 1993.

[3] H. Zhu and F. Yu, "A cross-correlation technique for vehicle detections in wireless magnetic sensor network," *IEEE Sensors Journal*, vol. 16, no. 11, pp. 4484–4494, 2016.

[4] Z. Liu, D. Liu, and T. Chen, "Vehicle detection and tracking with 2d laser range finders," in *2013 6th International Congress on Image and Signal Processing (CISP)*, vol. 2. IEEE, 2013, pp. 1006–1013.

[5] J. Fang, H. Meng, H. Zhang, and X. Wang, "A low-cost vehicle detection and classification system based on unmodulated continuous-wave radar," in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. IEEE, 2007, pp. 715–720.

[6] P. Vamsikrishna, S. R. Hussain, N. Ramu, P. M. Rao, G. Rohan, and B. D. S. Teja, "Advanced raspberry pi surveillance (ars) system," in *Communication Technologies (GCCT), 2015 Global Conference on*. IEEE, 2015, pp. 860–862.