B.Sc. in Computer Science and Engineering Thesis

Towards Devising An Automated System for Determining The Length of A Traffic Jam in Dhaka City

Submitted by

Md. Saidul Hoque Anik
1105017

Supervised by

A. B. M. Alim Al Islam

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh

February 2017

CANDIDATES' DECLARATION

This is to certify that the work presented in this thesis, titled, "Towards Devising An Automated System for Determining The Length of A Traffic Jam in Dhaka City", is the outcome of the investigation and research carried out by us under the supervision of A. B. M. Alim Al Islam.

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Md. Saidul Hoque Anik
1105017

CERTIFICATION

This thesis titled, **"Towards Devising An Automated System for Determining The Length of A Traffic Jam in Dhaka City"**, submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in February 2017.

Group Members:

Md. Saidul Hoque Anik

Supervisor:

A. B. M. Alim Al Islam
Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

ACKNOWLEDGEMENT

First of all, I would like to give praise and thanks to Almighty Allah for giving me this unique opportunity to study in a wonderful environment and to have this unparalleled taste of doing research work with such talented people. I am also thankful to my family members for providing me constant support regardless of my unorganized daily routines. I am thankful to my thesis supervisor A. B. M. Alim Al Islam for being so kind to me when I was constantly struggling to make progress. I am thankful to Kazi Hasan Zubaer for reaching out to me lending his electrical equipments and building the circuits for me. I am also truly thankful to Tusher Chakraborty and Taslim Arefin Khan for helping me build the circuits and for providing me valuable electrical components and the authentication for the web server.

Dhaka
February 2017

Md. Saidul Hoque Anik

Contents

List of Figures

List of Tables

ABSTRACT

Different methods are used to sense the road condition to generate traffic signals for an intersection. However, traditional approaches fail to meet the challenges that the roads of Dhaka city pose. Our proposed model uses a group of sensors connected to a central computer to sense the state of vehicles on the road. In order to predict the traffic condition and calculate the length of jam, we had to solve a few additional problems. We had to find a way to make the sonar sensor readings noise-free. We also had to consider irregularly shaped vehicles and their unpredictable movements. Finally, we had to make sure that we may produce the correct result from the system despite having pedestrians and food carts that are very common on the road.

Chapter 1

Introduction

Traffic controlling is usually done by police sergeants in Dhaka city. A few attempts were taken to automate the process using traffic signaling device, but the result was not satisfactory as the stop signals would switch between green and red at a fixed time interval. As the road condition and traffic density change throughout the day, the signal generation demands a more dynamic approach.

1.1 Automating the Signaling Process

In order to automate the traffic signaling process, sensing the road condition is crucial. The automated system must know which part of the road has more traffic density and which part is light on traffic.

By gathering the informations from the road, the system may be able to generate the suitable amount of time for showing green signal on one side so that the traffic jam is dissolved. At the same time, the system must consider the fact that blocking one part of the road for a large amount of time may create a bigger traffic jam while clearing jam on the other part.

1.2 Different Types of Sensing Devices

Different types sensing devices are available for sensing the condition of a road. Each has it's pros and cons. Some device can capture the image of the road, some produce thermal images, others sense the road through the change in physical property of the road. Some device are more expensive than others, eg., thermal camera is more expensive than regular cameras. Some device can only work well in a well-lit environment, such as closed circuit camera. Others can sense the road regardless of the lighting condition on the road, eg., infrared camera.

Selecting the right device is an important part of the automation process, because the output of the sensing device governs the work-flow of interpretation of data by the system.

1.3 Challenges in the Roads of Dhaka

Sensing the roads in Dhaka city is particularly challenging in a sense that the vehicles do not move through the road in well organized fashion. There are various types of vehicles, some of which are motorized, others are not. Most importantly, the vehicles tend to switch lanes in an irregular manner. Therefore, sensing the vehicle density is not merely a simple counting task using image analyzer.

1.4 Our Approach to Sensing the Road Condition

Considering the unique characteristics of the roads of Dhaka city, we have considered ultrasonic sensors to sense the condition of vehicles. Instead of counting the number of vehicles, our approach involves deploying multiple sensors on the side of a road to calculate the length of a traffic jam. Each of the sensors tell whether the vehicles in front are moving or are at standstill. From the sensor data, we can estimate the size of jam that may have been produced in a certain road, so that traffic signal may be set accordingly.

This approach imitates how a human makes decision by looking at a road, ie., a traffic police decides by looking at existence of jam. Here are some of the significant points of the study.

- The system uses only ultrasonic sensors. So the sensing can smoothly be done both during the day and night. It is also immune to foggy weather where visibility is significantly reduced.
- It works with heterogeneous vehicles too. Also it works in conditions where vehicles switch lane in an irregular fashion.
- Computing is done centrally for signal generation. So per unit computational cost is reduced.
- Deployment of the device is carefully measured so that occlusion from pedestrians and other non-relevant objects is minimized.
- Significant improvement has been made on the algorithm for sensing the roads using sonar sensor in order to reduce the false positive and true negative errors in sensor data.

- Further experiment is done on the extracted data in order to find the optimal sampling rate that minimizes power consumption.

Chapter 2

Related Works

Researchers have explored many techniques for sensing the traffic condition of the roads in a city. The developed systems from some of the papers and articles have even been implemented in some parts of the cities.

The data collection methodology varies from country to country. Some countries have the ability to invest on advanced technologies for traffic controlling. Others rely on improved algorithmic prediction from simpler technical equipments.

2.1 Traffic Signaling

Automated traffic signaling is very common in many countries. Several recent papers discuss this process using various approaches. For example, [2] proposes a solution that was actually implemented in Lanzhou City. The proposed system collected data via GPS[1], analyzed them though mathematical model, and predicted traffic flow based on the result.

The authors of [3], on the other hand, propose a linear time-expanded model for both traffic assignment and signal parameter. The method uses linear model to compute travel time function, but requires more variables to describe traffic flow on a single link.

Among the other papers, [4] predicts traffic flow by clustering using BP (Back-Propagation) neural network, [5] creates a traffic network model based on network delay and packet loss rate in wireless signals from vehicle, and [6] produces smart traffic signals by applying Grey-Markov Model[2] on collected data.

[1]Global Positioning System

[2]A new forecasting model based on Grey Model and Markov Chain Model

2.2 Data Collection

One of the important parts of generating smart traffic signal is to collect traffic data in a reliable way. Vehicles can be sensed using various transducers [3] or through analysis of images.

2.2.1 Image Analysis

One way of sensing the road is to take pictures of the road and analyze the images to get an idea on how many vehicles are there on the road. This straight forward method requires image analysis tools on the server-side and a clear view of the road from the road-intersection. Picking up heat signatures is also an interesting way of tracking the vehicles on the road.

Real-time Photograph

Some Researchers have developed traffic signaling systems that rely on live closed circuit camera footage. One paper [7] senses vehicle by detecting edge on live images. Another article [8] also provides a method for analyzing traffic video footage. We can also find articles that actually count the number of vehicles for traffic prediction, eg., this paper [9]. Some other articles such as [10, 11] provide traffic models for special cases like detecting emergency vehicles[4] or city buses.

Thermal Image

The authors of [12] propose a traffic data collection method by calculating windshield temperature along with tires thermal energy reflection areas. Some other papers [13, 14] also discuss methods for traffic surveillance using thermal imaging.

2.2.2 Transducer

Among other popular transducers, many of the researchers around the world have used sonar sensors[5] in order to sense traffic density. One of the papers [1] proposes a solution using sonar sensor. It requires that a module is to be installed in every vehicle passing through the road junction. Another paper [15] provides an analysis of vehicle detection with WSN[6]-based ultrasonic sensors. This paper [16] extends the scope to multiple drive lanes.

[3] A device that interprets changes in a physical quantity of the real world to produce equivalent electric signals

[4] Eg., ambulance

[5] A sensor that records the elapsed time to receive reflected ultrasonic sound for measuring the distance to an object

[6] Wireless Sensor Network

Chapter 3

Motivation Behind Study

Several methods are discussed in Chapter 2 for sensing the road condition. However, roads of Dhaka city throw a set of challenges that are needed to be considered carefully. Choosing the right device from a wide range of detecting devices is also crucial for computing correct jam length of a road.

3.1 Challenges

The challenges found in Dhaka city road are different from ideal scenario. From the researcher's point of view, formulating a solution to perceive the condition of such roads is something that has not been considered that much. The following list shows the characteristics of the typical roads of Dhaka city holds.

- The types of vehicles on the road are heterogeneous.
- Vehicles do not maintain a fixed lane while driving.
- Food-carts and hawkers may occupy part of a road.
- Vehicles such as rickshaws have an uneven shape.
- Roadside trees and banners may occlude part of the road, thus, disrupting the visibility from intersection.

3.2 Criticism on Some of the Detection Methods

Considering the challenges noted above, it's difficult to sense the road using a closed-circuit camera. Because someone may put a banner or a sign-board on the side of a road which will obstruct the view of the camera.

We could not use thermal imaging either. Because wind-shields and tires of a car may be obstructed by a rickshaw, or the view may be occluded if vehicles are in a disorganized manner, which is very common in Dhaka city.

Infrared sensor will not provide a very accurate reading considering the limited range of the sensor and the temperature fluctuation in the city. Induction loop could be used in a limited area, but it is prone to wear and tear.

3.3 Choosing a Sensor

The discussion on previous section leaves us with only one suitable sensor, the ultrasonic sensor. One of the many advantages it has is that it works at night with equal efficiency. However, we cannot directly use the methodologies discussed by articles in Section 2.2.2 because of the challenges we have in Dhaka city. It is not realistic to install sonar sensor in every rickshaw of the city. We cannot ensure that vehicles on the road will maintain static lane. We cannot also ensure that there will not be any overlapping between the vehicles. So we need to come up with a new solution that faces the challenges discussed in Section 3.1.

Chapter 4

Problem formulation

Now that we have decided to use sonar sensor to detect traffic condition on the road, let's take a look at the things that we shall need to consider.

4.1 Consideration on Using Sonar Sensor

Sonar sensor is a popular choice among the researchers for calculating distance. It is light in weight and easy to use. However, when we are using it to measure the length of a traffic jam in a busy road, we must consider the followings carefully.

- The sensor is usually very accurate in calculating distance. However, occasionally it produces noise.
- The calculated distance points from sonar must be able to predict the road condition.
- When predicting traffic jam, the parameters must be tuned so that the ratio of false positive and true negative is kept in balance.
- The solution will be used in Dhaka city. The vehicles are heterogeneous and tend to change lanes unexpectedly and frequently.
- Pedestrians, bicyclists and food-carts are common in Dhaka. They may disrupt the readings of the sensor.

We need to devise a way to sense the traffic jam intensity using sonar sensor overcoming the challenges discussed above, so that the traffic signaling duration can be properly calculated for intersections.

A traditional method for removing noise from sonar sensor is to take several consecutive readings and choosing the median point as the final output. But this method is not fool-proof. The finally selected point may still contain noise, if the median point is noise itself.

4.2 Interpreting Sensor Data

The sonar sensor will only send the calculated distance to the object in front in a particular interval. In order to sense whether there is traffic jam or not, we must find a way to tell what the data-points indicate.

The process of interpreting collected data can be ‘taught’ to an artificially intelligent machine. We need to explore the wide range of models of machine learners in order to find the right model that would tell us what the road condition is from collected data-points.

4.3 Consideration on Energy Consumption

We also want our solution to be as energy-efficient as possible. Therefore, we want to make sure the deployed devices use least amount of work to produce the most optimal solution. This includes finding the optimal sampling rate and the number of consecutive readings taken by the sonar sensor.

We must keep in mind that reducing the sample rate increases energy efficiency, but may decrease prediction performance. Increasing the rate does the opposite. Therefore, we must find the right balance between these two.

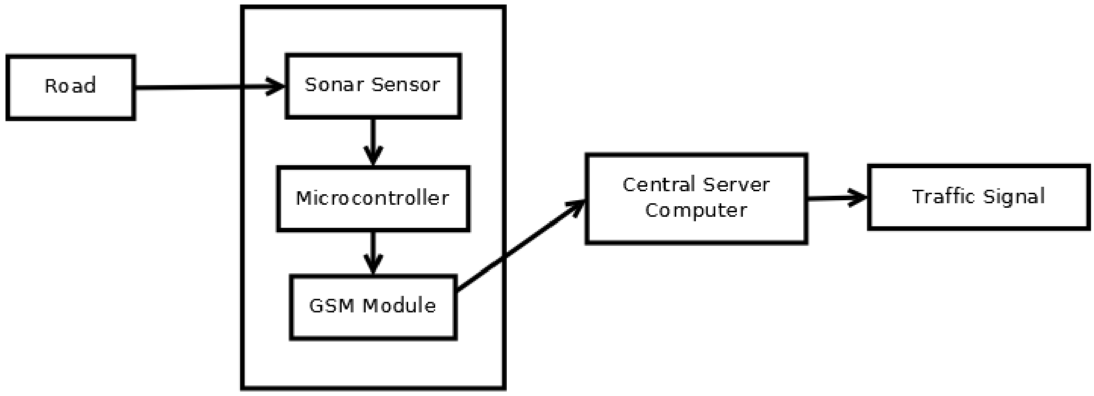

Figure 5.1: Flowchart of the methodology

Chapter 5

Proposed Solution

In order to predict the road condition, we deploy several sensor modules on the sides of every road. Through each module, we receive the distance data in a regular interval. Then we use group of distance data to predict if the vehicles are mobile in front of that particular module or not. If the vehicles are stagnant, we can get an idea on how long the traffic jam is from the data of other modules combined. From the length of jam, traffic signaling duration can be calculated afterward. The flow of data processing is pictured in Figure 5.1.

5.1 Overcoming Challenges

We have pointed out several challenges in Chapter 4. This section discusses how we have dealt with the challenges during the research period.

Noisy data We have devised a method which can predict approximate distance from several data-points with one or two noise points.

Jam prediction Jam prediction is done using a machine learner. It uses deep learning to predict the road condition based on previous observations of input data.

Parameter tuning The machine learning tool is tuned using labeled data from previous records of readings on the same location.

Lane changing vehicles Our solution approach only considers if there is jam or not. As it does not consider vehicle identification or count, lane changing is not a problem.

Pedestrian The modules are placed in such a height and angle that pedestrians are overlooked.

5.2 Solution Development

In order to build a robust solution that ensures a continuous flow of informations from the busy roads to our central server, we need to craft our module carefully. The modules are deployed on the side of the roads. They contain the sonar sensor along with a transmitter that transmits information through wireless network to the central server.

5.2.1 Module Placement

Placing the module correctly solves a set of challenges stated earlier. We need to deploy minimum number of modules to ensure cost efficiency. We also need to eliminated data corruption from pedestrians. So we need to choose not only on which side of we have to deploy the device, but also at which height and what angle we need to set the module.

Choosing a Side

We need to find the length of the jam. So we need to know on which side the jam tail ends, ie., does the last vehicle choose to go to the rightmost lane, or the leftmost lane? If the jam ends on the right side, we should place our module on the right side, so that we do not miss some portion of the jam.

A group of satellite images from Google Maps reveal the answer. The tail actually depends on the property of the junction! Most of the vehicle choose the lane that is closer to its destination road.

Finding the Correct Angle and Distance

When deploying the sonar sensor on the road, we must consider the pedestrians and food carts. So we need to find the correct angle and height of deployment. Considering the scenario in

Figure 5.2: Satellite images show on which lane the jam ends

Figure 5.3, we have derived the following formulas.

$$V_c = \frac{V_x}{n}$$

$$\theta = tan^{-1}\frac{V_c}{y - V_y}$$

$$h_1 = \frac{x(y - V_y)}{V_c}$$

$$H = h_1 + y$$

$$R = \sqrt{h_1^2 + x^2} + \sqrt{(y - V_y)^2 + V_c^2}$$

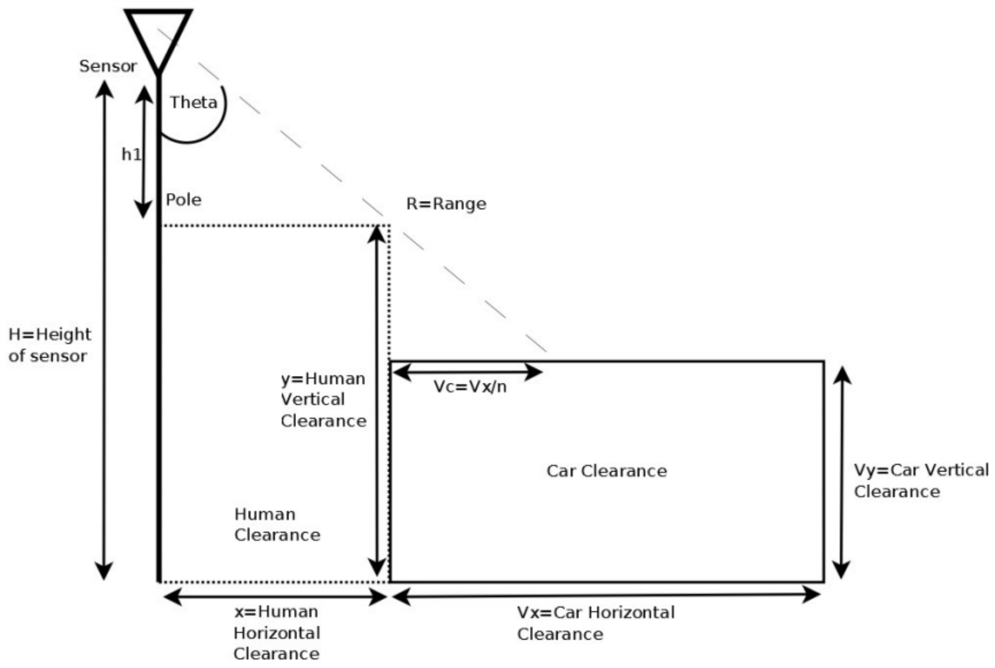

Figure 5.3: Scenario of the deployment

Here,

V_y = Car vertical clearance,

V_x = Car horizontal clearance,

$V_c = \frac{V_c}{n}$ = Area covered by sensor. For half area, n = 2,

x = Human horizontal clearance,

y = Human horizontal clearance,

H = Sensor height,

R = Range, &

θ = Sensor direction.

We can get H, θ, & R from the formulas stated above if n, x, y, V_x, & V_y are provided.

5.2.2 Noise-free Data Retrieval from Sonar Sensor

There are two types of noise that a calculated distance from sonar sensor may contain.

1. **Salt and Pepper:** Sudden zero value as distance
2. **Random:** Random fluctuation in reading

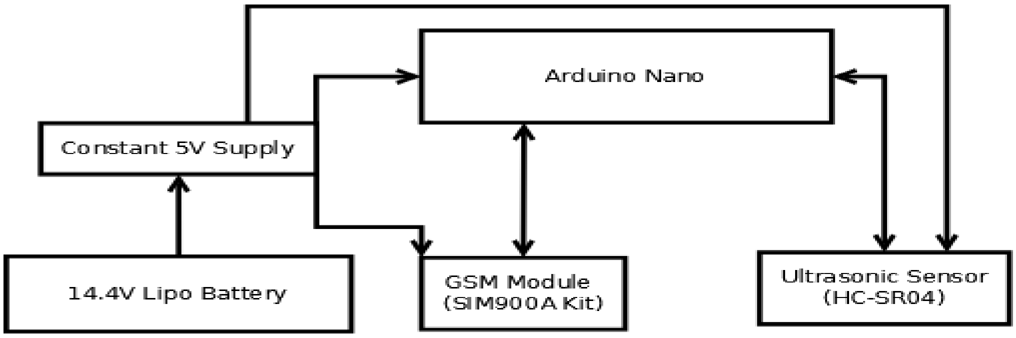

Figure 5.4: High-level overview of the on-board circuit

As the calculated distance is the only input for the system, it is crucial that the input should be as noise free as possible.

In order to approximate noise free data, 4 consecutive distance reading were taken from sonar sensor. Then the approximate distance d' is given by the following formula.

$$d' = (\text{Average}_{distances}) + (\text{Standard Deviation}_{distances})$$

5.2.3 Circuit Design

Primarily it was decided to use ATmega32A as it was a cheap micro-controller. However, due to the lack of user-friendly APIs, a better alternative named Arduino Nano was selected. Arduino Nano was not very expensive, but it provided an easy interfacing between the GSM device and Sonar sensor.

In order to make the device portable, a Lipo battery was included as a power source. To ensure wireless communication with the central server, a GMS[1] module was used. It uses a standard SIM card and EDGE[2] data service to connect to the Internet. Figure 5.4 shows the connection between the components in circuit.

5.2.4 Programming

In order to ensure the correct flow of data, the components of the system were programmed to handle inputs and produce outputs. In deployed modules, Arduino was the programmable module. The server end receives the information and saves it to database for further processing.

[1]Global System for Mobile Communication
[2]Enhanced Data rates for GSM Evolution, 2G data connection technology

Arduino

Arduino was programmed in order to correctly send the sonar sensor data. It was programmed to do the following tasks in a loop.

1. Assign sonar sensor connecting pins.
2. Initialize GSM module.
3. Take several readings from sonar sensor.
4. Combine the readings in a single packet.
5. Send the packet to sever through GSM module.
6. Repeat the process from Step 3.

The original code can be found in Appendix A.1.

Server End

The server request was handled by a PHP engine. The PHP script writes in a mySQL database on the same server. The process that the PHP script follows is given below.

1. Receive data packets.
2. Set current region to Dhaka to get the correct server time.
3. Assign present date-time-stamp to the packet.
4. Insert it in database.

The code written in the script can be found in Appendix A.2

mySQL database may need an initialization before it can be used. The initialization code is written in Appendix A.3.

5.2.5 Artificial Intelligence

In order to interpret road condition, we need to find pattern in the collected data. This is something that is seamlessly done by humans. However, to automate this process, we need a machine learner that will be able to learn the pattern with some human guidance at first, and then will be able to tell if the captured data-points represent traffic jam on its own.

Machine Learning Model

Various types of learners are available that can recognize pattern from input sequence. Each of them has a different way of interpreting data, so we must choose a model that fits our scenario.

We need to decide whether there is jam (marked by 1) or not (marked by 0). This is a classification problem, ie., we need to classify input data-points as either 1 or 0.

Several models are there to attack a classification problem. Naive Bayes is used for identifying what the topic of a group of text is. However, our problem requires that we classify a set of real world data points based on previous experience. For this purpose, we have chosen a neural network model. This is one of the models that is used to automate driver-less cars.

There are different variations of neural network. We are choosing deep neural network model so that we can generate a better estimation of the data-points we have collected.

Data-mining Tools

Our problem requires a straight forward deep learner. We do not need to modify the internal algorithms of it. So we can simply use a data mining tool that has deep learning model in it's library.

Weka is a popular choice among data-miners. However, there is a better data-miner called RapidMiner with a more user-friendly interface and a large community base. So we have used RapidMinder for our data-mining operations. It also allows us to tune different parameters of deep learning models.

Chapter 6

Experimental Evaluation

After deployment of the modules, we need to find out how our system performs. Collecting data from the road, sending to server, processing the data-points for sending it to machine learner, and post-processing data-points - each of the stages has it's benchmark.

We need to ensure that each stage has performed up to the mark and produced satisfactory results. This procedure is carried out by experimental evaluation.

6.1 Experimental Settings

Our proposed system consists of several components. Each of the components has a few parameters that affects the performance of the output. So we have carefully chosen the values in order to ensure optimal performance.

6.1.1 Sonar Sensor Tuning

There are two parameters that we can tune in a sonar sensor. These two play a vital role in producing accurate results. They are given below.

- Sampling Rate of Sonar Sensor: 2 seconds
- Number of Consecutive Sonar Sensor reading taken : 4

The reason why 4 readings are chosen is described in Section 6.2.1. The sonar sensor placement is shown in Figure 6.1

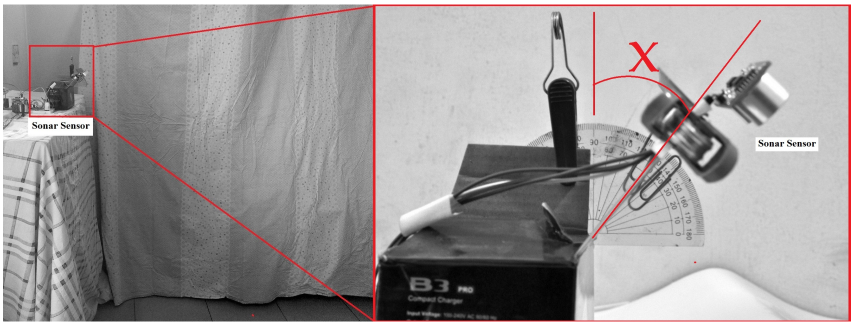

Figure 6.1: Sonar sensor angle

Figure 6.2: Module deployment

6.1.2 Module Placement

The module was placed as shown in Figure 6.2. The height and direction is given below.

- Height: $50\ cm$
- Angle X : $45\ degree$

6.1.3 Post-processing the Data-points

Raw data from sonar sensor was recorded as measured distance and collection time. The data collection process was also video recorded. After that, each distance (x) was assigned a jam coefficient (y) having value of either 0 or 1. Here, 0 denotes that the road was free from traffic, and 1 denotes that either the road was busy, or it was experiencing jam.

Various types of datasets were made from (x,y). The first type of dataset was different in temporal input points, ie., predicting a single jam coefficient (y) based on previous several distance observation (x). One to six observations were considered.

Another type of dataset was made by skipping the alternative values. For example, x1 and x3 was used to predict y3, x2 and x4 was used to predict y4 and so on. Two and three consecutive skipped values are considered.

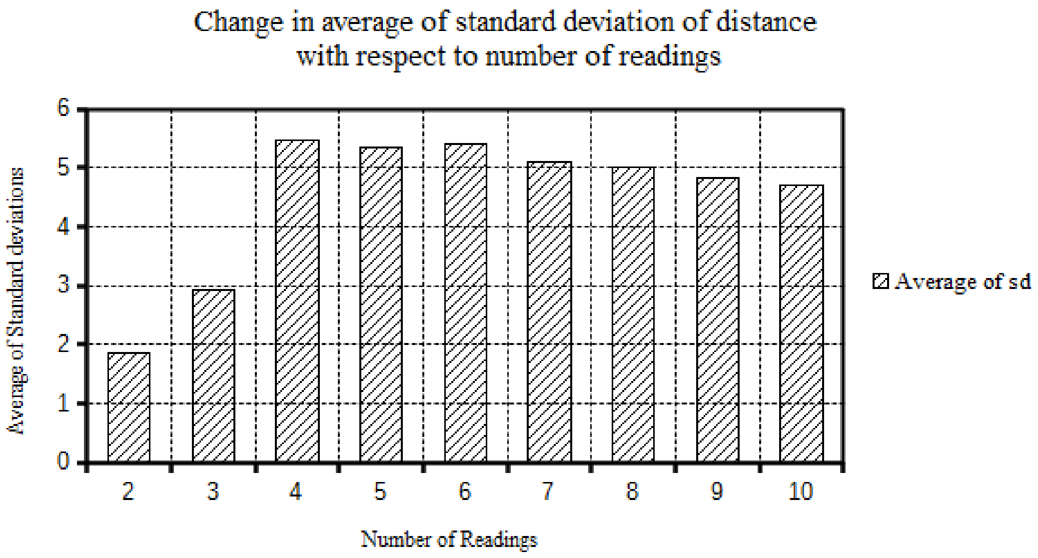

Figure 6.3: Choosing the number of consecutive readings

6.1.4 Machine Learning Model

As our's is a classification problem, deep neural network learning model was used. The parameters were the following.

- Activation Function: Rectifier function
- Validation: K-Fold cross validation (K=4)
- Sampling Type: Linear

6.2 Experimental Results

6.2.1 Finding Optimal Number of Consecutive Reading from Sonar Sensor

Sonar sensor may produce occasional noise points. Therefore, we need to take consecutive readings to make sure that we do not miss any sample. In order to make the device efficient, we need to find the least amount of consecutive readings we should take that covers possible erroneous points. So an experiment was done to calculate the distance of a fixed object of which the distance was already known for one up to ten consecutive readings. The result is shown in Figure 6.3.

Table 6.1: Prediction using a single point

	True 0	True 1	Class Precision
Predicted 0	48	25	66.75%
Predicted 1	94	68	41.98%
Class Recall	33.80%	73.12%	

Table 6.2: Prediction using 2 points

	True 0	True 1	Class Precision
Predicted 0	39	18	68.42%
Predicted 1	102	75	42.37%
Class Recall	27.66%	80.65%	

Table 6.3: Prediction using 3 points

	True 0	True 1	Class Precision
Predicted 0	83	132	61.40%
Predicted 1	8	10	55.56%
Class Recall	94.29%	10.75%	

Table 6.4: Prediction using 4 points

	True 0	True 1	Class Precision
Predicted 0	32	20	61.54%
Predicted 1	108	72	40.00%
Class Recall	22.86%	78.26%	

6.2.2 Prediction Using Continuous Data-points

Table 6.1 to Table 6.6 show the accuracy calculated using one to six consecutive data points.

Considering both accuracy and standard deviation, we need to find the number of consecutive data-points required to produce the most optimal output. The comparison of accuracy and standard deviation is pictured in Figure 6.4.

Table 6.5: Prediction using 5 points

	True 0	True 1	Class Precision
Predicted 0	122	77	61.31%
Predicted 1	17	15	46.88%
Class Recall	87.77%	16.30%	

Table 6.6: Prediction using 6 points

	True 0	True 1	Class Precision
Predicted 0	113	78	59.16%
Predicted 1	26	13	33.33%
Class Recall	81.29%	14.28%	

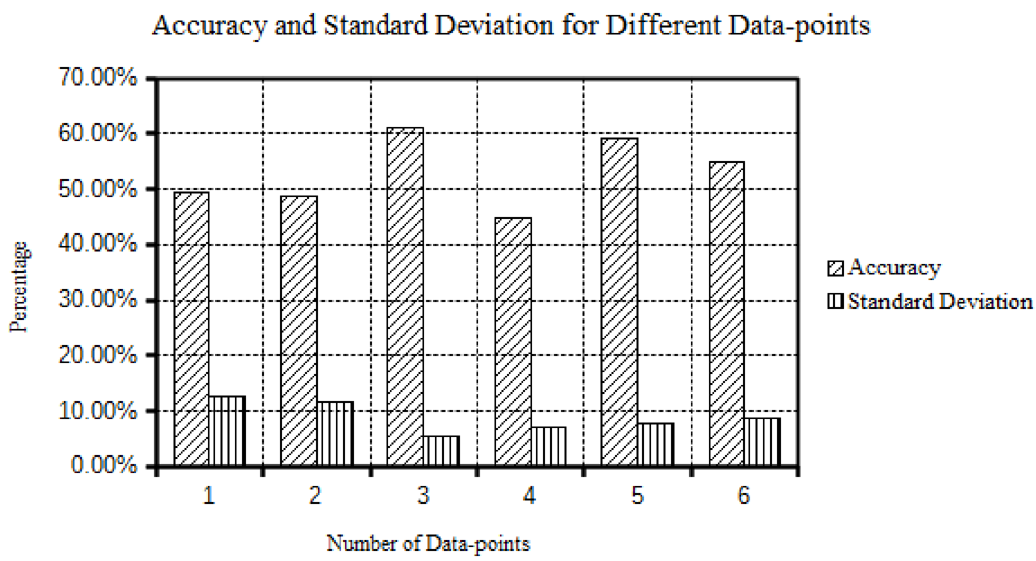

Figure 6.4: Comparison in accuracy and standard deviation for different consecutive data-points

6.2.3 Prediction Using Skipped Data-points

In order to conserve energy, we decided to check if skipping one or more data-points can produce similar accuracy. Table 6.7 and Table 6.8 show the accuracy calculated using consecutive 2 points skipping the middle one and middle two accordingly.

The comparison in accuracy and standard deviation for these two cases are given in Table 6.9

Table 6.7: Prediction using 2 points skipping the middle one

	True 0	True 1	Class Precision
Predicted 0	19	13	59.38%
Predicted 1	74	128	63.37%
Class Recall	20.43%	90.78%	

Table 6.8: Prediction using 2 points skipping the middle two

	True 0	True 1	Class Precision
Predicted 0	46	22	67.65%
Predicted 1	94	70	42.68%
Class Recall	32.86%	76.09%	

Table 6.9: Comparison in accuracy and standard deviation for skipping points

Data-points skipped	**Accuracy**	**Standard Deviation**
2	62.84%	5.87%
3	50.00%	12.55%

6.3 Experimental Findings

From the experimental results we have found our optimal parameters for each of the components. We have considered the noise produced in sonar sensor. We have also found optimal parameters that ensures device efficiency.

6.3.1 Optimal Number of Subsequent Readings in Sonar Sensor

We have found that, taking four consecutive readings in sonar sensor ensures that we shall be able to approximate correct distance. From Figure 6.3, we see that four readings produce maximum standard deviations, so we can choose four consecutive points while using sonar sensor.

6.3.2 Prediction Using Continuous Data-points

As shown in Figure 6.4, taking three consecutive values produce the best accuracy along with the best standard deviation. So when we are using a deep learning model, we can provide consecutive three data-points as input to predict the possible state of the road,.

6.3.3 Prediction Using Skipped Data-points

Comparing accuracy from Figure 6.4 and Table 6.9, we see that taking two data-points skipping the middle one may produce a more accurate result. However, skipping 2 points causes the prediction to become random. Therefore, we may use the skipping a middle point technique in order to make the system more energy-efficient.

Chapter 7

Future Work

In our study, we have tackled several challenges ranging from noise suppression in sonar sensor to finding optimal number of consecutive data-points to predict traffic jam length. However, there is always room for improvement in any research project. In future, we would like to improve our system in the following areas.

7.1 Sonar Sensor Placement

We can make different combination of settings while deploying the sensor. We can choose from the following configurations.

- Placing sonar sensors on both sides of the road
- Using two sonar sensors at the same location with different directions
- Placing sonar sensor on top of the road

By choosing from different configurations, we can find the optimal placement for deployment of the module. It may be possible that one road suits a particular configuration while another road suits a different one.

7.2 Power Rating Calculation

Although we have considered energy-efficiency by choosing the least number of optimal data-points, we have not actually calculated the power consumption of a single module per unit time. After calculating the actual power rating, we shall get a clearer view on which component to focus for making the device more energy-efficient.

7.3 Automation of Training a Module

In order to predict the road condition, each road needs at-least 15 minutes of labeled data, ie., we need to manually tell the machine learner which data-points mean jam and which points mean that the road is free. The task is daunting and takes up significant amount of time.

However, we can use image analyzers such as OpenCV to interpret the video footage so that the system can automatically tell whether there is jam or not from the given sample footage.

7.4 Others

There are a few other things that we can try to improve. A list of them is given below.

- Further improvement on sonar sensor to make it immune to fluctuation of weather
- Reduction of sampling rate from each module so that road condition sensing can be more accurate
- Making the GSM device more reliable and robust to sudden network failure

Chapter 8

Conclusion

In this paper, we have shown how to build and use a complete system to estimate the traffic jam in the roads of Dhaka city. Using the produced data dynamic traffic signals can be generated. Our goal was to sense the length of a traffic jam from a group of sonar sensors deployed on one side of a road while considering the irregular shape and motion of vehicles on the road. In this paper we have shown detailed methodology and experimental outcomes for this system.

We have also proposed a solution to handle noisy data produced from sonar sensor. The findings for this particular problem can be reused in other experiments where sonar sensor is used to calculate distance and sudden noise point can cause disruption in the experiment.

Another challenge we have handled is to reduce error due to pedestrian movement in front of the modules. We have produced a group of formulas that will help us determine the correct placement of the module if we want to sense vehicles from certain range while maintaining precise pedestrian and vehicle clearance. In future, we hope to continue this study so that we may improve the current traffic condition of Dhaka city.

References

[1] Wenjie, Chen, Chen Lifeng, Chen Zhanglong, and Tu Shiliang. ”A realtime dynamic traffic control system based on wireless sensor network.” In Parallel Processing, 2005. ICPP 2005 Workshops. International Conference Workshops on, pp. 258-264. IEEE, 2005.

[2] Zhang, Xijun, and Zhanting Yuan. ”Traffic Flow Prediction Based on the location of Big Data.” (2015).

[3] Khler, Ekkehard, and Martin Strehler. ”Traffic signal optimization using cyclically expanded networks.” Networks 65.3 (2015): 244-261.

[4] Li, Xiaofeng, and Weiwei Gao. ”Prediction of Traffic Flow Combination Model Based on Data Mining.” International Journal of Database Theory and Application 8.6 (2015): 303-312.

[5] You, Ziyi, and Pu Chen. ”Efficient Regional Traffic Signal Control Scheme Based on a SVR Traffic State Forecasting Model.” International Conference on Internet of Vehicles. Springer International Publishing, 2015.

[6] Zhao, Jiaxu, Zhide Chen, and Yali Zeng. ”Intelligent Traffic Light Model Based on Grey-Markov Model and Improved Ant Colony Optimization for Dynamic Route Guidance.” Cyber Security and Cloud Computing (CSCloud), 2015 IEEE 2nd International Conference on. IEEE, 2015.

[7] Datta, Sreemana, and Joydeep Mukherjee. ”Real-Time Traffic Control System using Fuzzy Logic based Edge Detector for Images.” International Journal of Computer Applications 119.14 (2015).

[8] Kanungo, Anurag, Ayush Sharma, and Chetan Singla. ”Smart traffic lights switching and traffic density calculation using video processing.” Engineering and computational sciences (RAECS), 2014 recent advances in. IEEE, 2014.

[9] Abbas, Naeem, Muhammad Tayyab, and M. Tahir Qadri. ”Real Time traffic density count using image processing.” International Journal of Computer Applications 83.9 (2013).

[10] Parthasarathi, V., et al. ”Smart control of traffic signal system using image processing.” Indian Journal of Science and Technology 8.16 (2015): 1.

[11] Ahmed, F., and Y. E. Hawas. ”An integrated real-time traffic signal system for transit signal priority, incident detection and congestion management.” Transportation Research Part C: Emerging Technologies 60 (2015): 52-76.

[12] Iwasaki, Yoichiro, Masato Misumi, and Toshiyuki Nakamiya. ”Robust vehicle detection under various environmental conditions using an infrared thermal camera and its application to road traffic flow monitoring.” Sensors 13.6 (2013): 7756-7773.

[13] Bahnsen, Chris, and Thomas B. Moeslund. ”Detecting road user actions in traffic intersections using RGB and thermal video.” Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on. IEEE, 2015.

[14] Kannegulla, Akash, et al. ”Thermal Imaging system for precise traffic control and surveillance.” International Journal of Scientific & Engineering Research 4.11 (2013): 464-7.

[15] Jo, Youngtae, and Inbum Jung. ”Analysis of vehicle detection with WSN-based ultrasonic sensors.” Sensors 14.8 (2014): 14050-14069.

[16] Jeon, Soobin, Eil Kwon, and Inbum Jung. ”Traffic measurement on multiple drive lanes with wireless ultrasonic sensors.” Sensors 14.12 (2014): 22891-22906.

Appendix A

Codes

A.1 Arduino Code

```
#include <EEPROM.h>
#define Trigger 5
#define Echo 4

int MEDIAN_ARRAY_SIZE = 9;
int ITERATION = 0;

unsigned long _GetDistance() {
  unsigned long ret;
  digitalWrite(Trigger,LOW);
  delayMicroseconds(2);
  digitalWrite(Trigger,HIGH);
  delayMicroseconds(5);
  digitalWrite(Trigger,LOW);

  digitalWrite(Echo, HIGH);
  ret=pulseIn(Echo,HIGH);
  ret=(ret/58.138)*0.39;
  return ret;
}

void WakeGsmUp() {
  //Serial.println("A");
  Serial.println("AT");
```

```
  delay(30);
}

String GetDistance()
{
  String s = "";
  for (int i = 0; i<4; i++)
  {
    s = s + String(_GetDistance()) + ",";
  }
  return s;
}
void DataUpload()
 {

    Serial.println("AT+SAPBR=3,1,\"Contype\",\"GPRS\"");
    delay(40);

    Serial.println("AT+SAPBR=3,1,\"APN\",\"gpinternet\"");
    delay(40);

    Serial.println("AT+SAPBR=1,1");
    delay(40);

    Serial.println("AT+HTTPINIT");
    delay(40);

    Serial.println("AT+HTTPPARA=\"CID\",1");
    delay(40);

    String _url = "supersavior.info/traffic/add_traffic_data.php?A=";
    String upCommand = "AT+HTTPPARA=\"URL\",\"" + _url + GetDistance() +
    Serial.println(upCommand);
    delay(40);

    Serial.println("AT+HTTPACTION=1");
    delay(50);
    Serial.println("AT+HTTPTERM");
    delay(40);
```

```
 }



 void setup() {
  Serial.begin(115200);
  pinMode(6,OUTPUT);
  pinMode(Trigger,OUTPUT);
  pinMode(Echo,INPUT);
  delay(100);
  WakeGsmUp();
}

void loop() {
      DataUpload();
}
```

A.2 PHP Server Script

```
<?php
if (isset($_GET['A']))
        {
        $s = $_GET['A'];
        date_default_timezone_set('Asia/Dhaka');
        $date = date('Y-m-d␣H:i:s');
        echo "$date";
        require 'traffic/connect.php';
        $query = "INSERT␣INTO␣traffic_data␣(distance,␣time_stamp)␣VALUES␣('$s',␣
        $result = mysqli_query($link, $query);
        $link->close();
        echo "Added";
        }
?>

<?php
    $host_name  = "localhost";
    $database   = "supersav_main";
    $user_name  = "supersav_main";
    $password   = "Zaheen1234";

```

```
    $link = mysqli_connect($host_name, $user_name, $password, $database);

    if(mysqli_connect_errno())
    {
        echo '<div␣class="popup">Connection␣Error:␣'.mysqli_connect_error().'</d
    }
?>
```

A.3 Database Initialization Code

```
CREATE USER 'traffic_admin'@'localhost' IDENTIFIED BY 'traffic_admin_123456';

create table traffic_data
(
id int(11) primary key auto_increment,
distance varchar(100),
time varchar(100)
);
```

Generated using Undegraduate Thesis LaTeX Template, Version 1.3. Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

This thesis was generated on Sunday 19th February, 2017 at 10:07am.