Traffic Congestion Detection and Optimizing Traffic Flow Using Object Detection, Optical Flow and Fluid Dynamics

by

Md. Shahriyar Hossain
21141017

Md. Imtiaz Bhuiyan
18101688

Marjahan Akther Dulali
17301010

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2022

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

shahriyar

Md. Shahriyar Hossain
21141017

Md. Imtiaz Bhuiyan

Md. Imtiaz Bhuiyan
18101688

Dulali

Marjahan Akther Dulali
17301010

Approval

The thesis titled "Traffic Congestion Detection and Optimizing Traffic Flow Using Object Detection, Optical Flow and Fluid Dynamics" submitted by

1. Md. Shahriyar Hossain (21141017)
2. Md. Imtiaz Bhuiyan (18101688)
3. Marjahan Akhter Dulali (17301010)

Of Fall, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on January 20, 2022.

Examining Committee:

Supervisor: (Member)

Md. Ashraful Alam, PhD
Assistant Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator: (Member)

Md. Golam Rabiul Alam, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department: (Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Bangladesh has been suffering a severe traffic congestion issue ever since it has been on a high paced development roadmap. Researches regarding solving such traffic issue has been in the talks but has never reached a proper conclusion and far from implementation. It has slowly grown into a towering challenge to overcome. And with an aim to topple that tower, we propose a 3 layer architecture to solve this problem. The proposed model consists of object detection, speed measurement and decision based on traffic flow. Using neural network object detection algorithms, it will detect congestion and the speed of the congestion. Then, it will use fluid dynamics based model to get the traffic flow, pass data between other signals and provide correct traffic signals. All signals would interact with each other like hive mind to maximize the traffic flow in any intersection. With the working model we had at our hand, we ran rigorous experiments to check whether our model works or not. Our results indicate that our model surpasses all other similarly implemented models by a noticeably large margin.

Keywords: AITS, ITSC, Deep Learning, Image Recognition, Self-Adaptive Traffic System, ATS, City Traffic, Fluid Dynamics, Numerical Simulation, Traffic Simulation, Object detection, Optical Flow, Traffic Flow

Dedication

Dedicated to the people who work hard for the people of this country.

Acknowledgement

Firstly, all praises to the almighty Allah, by whose will we have completed this thesis without having to face any major issues or interruptions.
Secondly, to our supervisor Dr. Ashraful Alam for guiding us throughout the thesis and providing us directions and resources, with the help of which, we were finally able to complete the thesis properly.
And finally to our parents who have supported us right from the beginning and showed their kindness during our tough times.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AI Artificial Intelligence

BN Bayesian Network

$CCTV$ Closed Circuit Television

CNN Convolutional Neural Network

GPU Graphical Processing Unit

IoT Internet Of Things

ITS Intelligent Traffic System

$ITSC$ Intelligent Traffic System Control

KNN K Nearest Neighbour

$LSTM$ Long Short Term Memory

LWR Lighthill-Whiteham-Richards Model

mAP Mean Average Precision

$RCNN$ Recurrent Neural Network

$RFID$ Radio Frequency Identification

RGB Red Greed Blue

ROI Region Of Interest

$YOLO$ You Only Look Once

Chapter 1

Introduction

Traffic congestion is one of the biggest nightmares for any megacity. Our Dhaka city is no exception to this as it still has manually regulated traffic systems because of unplanned road design, weak infrastructure and also for reckless drivers who have no regard for traffic regulations [15]. According to available statistics, Dhaka inhabitants spend an average of 170 hours per year on the road, [19] resulting in a staggering loss of BDT 1 million each day [13]. For this sole reason, modernizing Bangladesh's traffic management system has become an urgent need.

There have been numerous studies conducted on technologies such as IoT-based cloud solutions.[14], [16], [19]. RFID-based vehicle detection, as well as V2I roadside infrastructure that combines AI and big data. Considering Bangladesh has many cars that are not equipped with contemporary technology, almost all of these vehicles missing the smart characteristics that we want to establish an IoT-based architecture. Which makes implementation of the highly effective, efficient but sophisticated much more difficult.[3], [21], [33], [37], [41], [43]

Thankfully, some effective research on forecasting short-term and long-term traffic flow fluctuations using an image-based technique have been published.[34], [39]. Furthermore, anticipating traffic at least 5 to 10 signal cycles beforehand provides the system with a solid perspective [35] of how the traffic should be managed and allows the system to proactively modify the traffic signals. In addition, the system may adjust the length of traffic signals based on traffic load.[5], [36]

This paper will go into detail on the approaches used, such as parametric and non-parametric models. [36]. Non-parametric models can also be considered as a good alternative options as recent strides in computational capabilities and data quantity. We can now do several researches with not only Long Short Term Memory, Recurrent Neural Networks, Convolutional Neural Network, Graph CNN but also faster computation with SVM, KNN, Bayesian Network Random forest [6], [7], [12], [28], [33], [35], [40], [47], [48].

1.1 Research Problem

Traffic congestion causes huge mental pressure on people because it causes rush among people[28]. Not to mention the amount of carbon emission occurring from the vehicles since only a scarce amount of cars are hydrogen cell cars. If it is looked at statistically, Bangladesh loses an unbelievable amount of resources due to the traffic problem. Around 37 thousand Bangladeshi taka is lost and approximately 4 million profession working hours are being spent on the road[41]. So it can be implied that, traffic congestions significantly damages our economy, health and environment.

In an approach to solve the problems, researchers went to various domains of scientific discoveries to find which solutions could be implemented appropriately. One such domain is the new emerging domain of machine learning and artificial intelligence. They have noticed the possibility of integrating these technologies into these issues, providing a new way to solve these scenarios [20], [21]. Additionally, a lot more efforts and investments are already being made into this field in order to make great strides in solving traffic congestion problems.

To put in simple terms, the traffic system that Bangladesh currently maintains doesn't deem to even be acceptable. The system is broken and unusable. We respect all the traffic policemen for their contribution, but it simply isn't in any shape to be even recovered. It has a very heavy latency due to manual human intervention. Not to mention the recurring miscommunication, and it is not even the start [15]. They communicate with their counterparts on each intersection, but the overwhelming traffic flow becomes infeasible for human interpretation [30]. In addition to that, fatigue catches up to them quite easily due to rough environment and weather conditions, and they lose motivation to conduct the traffic flow properly [28]. But if we can imagine a big centralized AI controlling everything with end to end control, we can easily avoid all the drawbacks we mentioned above. Of course, it is not just traffic police that we are considering weakness. The city has heavy concentration of vehicle, no strict rules, malfunctioning equipments are just to name a few [33]. Either way, we can safely assess that machine based solutions would perform the greatest optimization for a scenario like this.

1.1.1 Motivation

To the following issue, why can't we adopt IoT-based alternatives? [11], [14], [27], [29]? Bangladesh has about 3.4 million registered and 3 lacs unregistered cars, according to the BRTA database.Almost all of these vehicles lack the necessary modern technologies for an IoT-based solutions.[10], [37]. However, it is not viable to initiate the process of modernizing automobiles by attaching IoT devices to them right now. It would not only be an extraordinarily expensive infrastructure, but it will also be highly unlikely given the country's current status. [14]. As a result, an immediate and efficient substitute approach is necessary. With all that in mind, the question like which appropriate option to select for the best solution emerges. Is this even possible to find remedies for Dhaka? Yes, considerable study has been undertaken on this issue in order to select the most appropriate image-based solution customized to the demands of each unique city in order to solve their own set of traffic challenges.[4], [15], [21], [27], [30], [34], [39], Not only have academics discovered several methods for analyzing urban flow of traffic, but some methods have also been substantially deployed.[1], [2], [16].

A deep learning-based traffic state categorization approach based on our studied articles is suggested to identify congestion problems at urban crossings. A popular method is to depend on numerical data such as vehicle flow [40], [47]and speed.[40], [47]. Furthermore, utilizing the YOLOv3,CNN, R-CNN,HOG, or SSD algorithms, we can gather this sort of data from surveillance cameras. [22] [48] Afterwards, we can either use them directly using an image-based deep learning method or transform to formulation that can process numerical data. [12]. We discovered a hybrid approach, that incorporates YOLOv5 with LK optical flow, to more accurately pinpoint the matter. [20], [25], [48]. Together, these two methods offer benefits in data processing speed and accuracy. It moreover outperforms other object detection methods such as HOG, SSD, RCC, Fast R-CNN, Faster R-CNN.[21], [30], [35], [48].

Whenever it comes to anticipating traffic status, [36]AI-based methods [31], [42] can deliver the most precise prediction to a platform. However, a comparable but IoT-based system has been deployed and investigated more on the traffic control system in Bucharest's congested metropolis.[36]. This study, on the other hand, will show an updated version of a comparable model that will employ image-based data rather than IoT-based devices. Algorithms such as the Lucas-Kanade (Lk) optical approach, for example, can predict speed and anticipate movement of vehicles. [21], [30], [48]. CNN, a deep learning-based model, produces the best results when there is a huge dataset. [46]. It is, in fact, frequently employed in traffic speed prediction.[39] and traffic flow prediction[34].For these circumstances, a modified CNN with LSTM is also utilized [35]. This method is applicable among most urban places, even though in this case, we will just address the traffic issue in Dhaka.

To be more specific, the article covers what we will use and analyze the obtained data to enable for platform reconfiguration [38] for optimal traffic flow [19]. Accurate forecasts in a complex system like urban traffic flow are a difficult undertaking, especially when random disruptions from the environment are present[12], [26]. Moreover, in this example, meteorological circumstances have a role. [26]. We will also work past these barriers to develop a strong self-adaptive traffic management system.

1.1.2 Research Objectives

Automated traffic signal control, with optimized route suggestions and provide traffic flow, is our main goal for this topic. We aim to apply an image based model with the accuracy of the robust IoT based model. We set our standards to that level, although achieving 15% of it is more than enough for our purposes. Likewise, we also target to achieve this model by implementing a framework as simple as possible so that the system can be implemented as easily as possible. Such high standards may not be gained in this paper, but we will follow up with our works to see an end to finally establish a practical and predictable mode. The key goals of our research are-

1. To gain knowledge if ML-AI based models optimize traffic flow
2. To get insight into how object algorithm detects density
3. To build a smart ML based flow prediction algorithm
4. To judge the model's performance based on the precision it provides
5. To compare it with flagship models to assess the practical usability of our model

1.2 Thesis Orientation

The following chapters of the paper maintains the order as described below: Chapter 2 discusses literature review, the architecture we are looking for and similar in the field and existing methodologies and gives a strong analysis of the topics and related information that has relevance with our work. Chapter 3 presents the methodology of our work which includes the workflow, proposed model and its details inner workings. Chapter 4 is where we practically attempt to build and implement the model and results of our findings, as well as compare them with similar models of the same criteria. Finally, Chapter 5 contains the concluding notes of the thesis paper so far and discusses future plans regarding the thesis.

Chapter 2

Literature Review

Solving a complex traffic congestion problem like this is not easy. For this, we looked into already existing researches regarding the topic. It turns out that approach to solving traffic congestion is not a novel idea at all. To be precise, there has been at least a thousand researches conducted solely optimizing traffic flow. Optimization of traffic flow is not an easy feat, and we also don't intend to do so. What we want to achieve is a solution that Bangladesh can easily welcome. Automated solutions are obviously our main goal but if that requires implementation of infeasible and expensive infrastructure then it's safe to say that such a research would go in vain as the country cannot benefit from it too much. Therefore, IoT based solutions are to be avoided to implement a feasible solution for the country that is image based solutions.

To get started with our research, we started searching for documents in popular site like Scopus, Google Scholar. We search for "Traffic Optimization", "Intelligent Traffic Control System", "Smart Traffic Control", "Image based Traffic Solution". We got a lot of researches and shortlisted to about 1200 documents. Now we went into advanced filter where we searched for "Image Based Solution". This helped us narrowed down to 120 documents. Then we looked for traffic optimization and traffic solution researches, narrowing down to 80 documents. We also had to remove some of the false positives that included RFID/IoT/V2I infrastructure. With this filtration, we managed to drop to 60 researches and article that we deemed worthy of looking into for our research paper.

2.1 Intelligent Traffic System Control (ITSC)

Commonly used for traffic management, it mostly uses managetic detector and signal systems using loops [2]. These generally have the capability to control traffic dynamically and optimize traffic flow. It intercepts traffic condition some signal cycles ahead and directs the flow accordingly to avoid gridlocks. This system is an integral part of smart city concepts [19]. Two important concepts are to be noted here, SCOOT [2] system in Ontario and SCAT [1], both are ITSC concepts, based on dynamic traffic coordination [37]. These systems are extremely competent in getting accurate traffic flow.

2.2 Used Architecture

The feasible approach that we are planning to implement are more than capable to solve the problem in our opinion. The approaches we will be following are brilliant in their own respect field but they have never been used together in a single system. The approaches we will considering are below:

We will use the following 3 layer architecture for our task that we think would more than appropriate to have a go at providing optimal solution at our problem.

1. **Object Detection Algorithm:** The YOLO algorithm is one of the most widely used object detection algorithm [20], [22], [48]. It uses backpropagation, convolutional layer, CSP and spatial pyramid pooling for object detection. YOLO was in action more than eight times [4], [6], [27], [47], [48], among our literature review. So we can safely presume it to be the most formidable model for object detection.

2. **Optical Flow :** Speed detection is a crucial part of our system and optical flow seems the most comfortable choice. Mentioned 5 times in literature review [18], [19], [45], [12], this model can use frame differencing to calculate global speed of a group of traffic more specifically, congestion.[48].

3. **Numerical Simulations Module :** This module is used in coherence with above two models to indicate the flow stream of traffic and suggest optimal route. Using fluid dynamics, it approaches to solve the problem in mathematical order. [32].

2.3 Related Works

This section provides a detailed review of previously completed researches where smart or intelligent traffic models have been implemented. To be precise, we are reviewing mostly image based data, but we also review RFID/IoT/V2I infrastructure where we extracted and tried to bring up relevant information regarding the case we are solving.

In the past few years, ordinary computers' computing capabilities have risen dramatically, enabling developing and testing numerous RL and DL based algorithms to handle traffic congestion problems [3], [7]. The first entirely self traffic signal uses a Q-estimation network to dynamically learn how to control-based on the Cerebellar Model Articulation Controller (CMAC). [6]. Silva et al. and Oliveira et al. [49] also presented a context detector in combination with RL. [8].

Nevertheless, in the last 20 years, artificial intelligence and machine learning have been explored extensively in the domain of traffic light management. [31], [36], [37], [42], [44]. Mikami et al. introduced a distributed reinforcement learning system that employs a genetic algorithm, popularly known as the Q-algorithm, to properly improve a system's traffic flow. [3]. Later, empirical correlations based on fuzzy logic and neural networks, such as Bingham's projected RL for parameter search, were demonstrated. [4], Choy's RL-based adaptive large-area cooperative system [5]. Nevertheless, because processing capacity was insufficient to do such tasks, the majority of these preliminary works were limited to theoretical fields.

Depending on these studies, a Hybrid Petri Net model [36] was developed that relied on traffic-dependent phase lighting. Through a control architecture, this was accomplished at a reactive level in Bucharest. [28], [34]. In a number of publications, the topic of smart cities with ITSC integration was also discussed.

Using deep learning approaches [31], [43], the study states [36] an upgraded version of the presently used ITSC systems that can forecast traffic circumstances 5-10 signal cycles ahead of time and dynamically change the signal light.

This system is critical to our work since we will be working on producing an optimized version of the recommended architecture with help for overcoming our country's technical shortcomings.

Chapter 3

Methodology

The focus of this research is to develop an entirely automated traffic management system. To do our tasks efficiently, we must first obtain photographs from surveillance systems. To execute out this task, we must first collect raw picture data and label it into various train and test sets, after which we would train our suggested model using the train set.Likewise, we inject photos into our algorithms and test them with real-world data to ensure the correctness of the projected traffic flow. Ultimately, we compare its precision to that of other systems in order to determine the best one. Figure 1 depicts a high-level overview of the entire process.

This research aims to present a deep learning based AI model that can detect congestion and the motion of it, and can predict its next destination. To achieve this, first we need raw image data that we can use to build our neural network model. We will need a properly labeled dataset and, using the train set, we will train the model. Then using motion detection algorithm we will prepare it for the prediction part. To easily describe our works, we have divided the whole process into 5 sections. These are:

1. **Dataset Collection:** Due to the ongoing pandemic, we couldn't collect first hand or primary data. Thus, we had to settle for secondary data collected from the internet. We will utilize this dataset to train our model and run further validation.

2. **Data Pre-Processing:** In this step, we prepared the dataset for our use by applying some color filter, cropping and other modifications which we assume will increase the accuracy of tests.

3. **Training the Object Detection Model:** In this phase, we will train and test the object detection model, which will be later used for congestion detection.

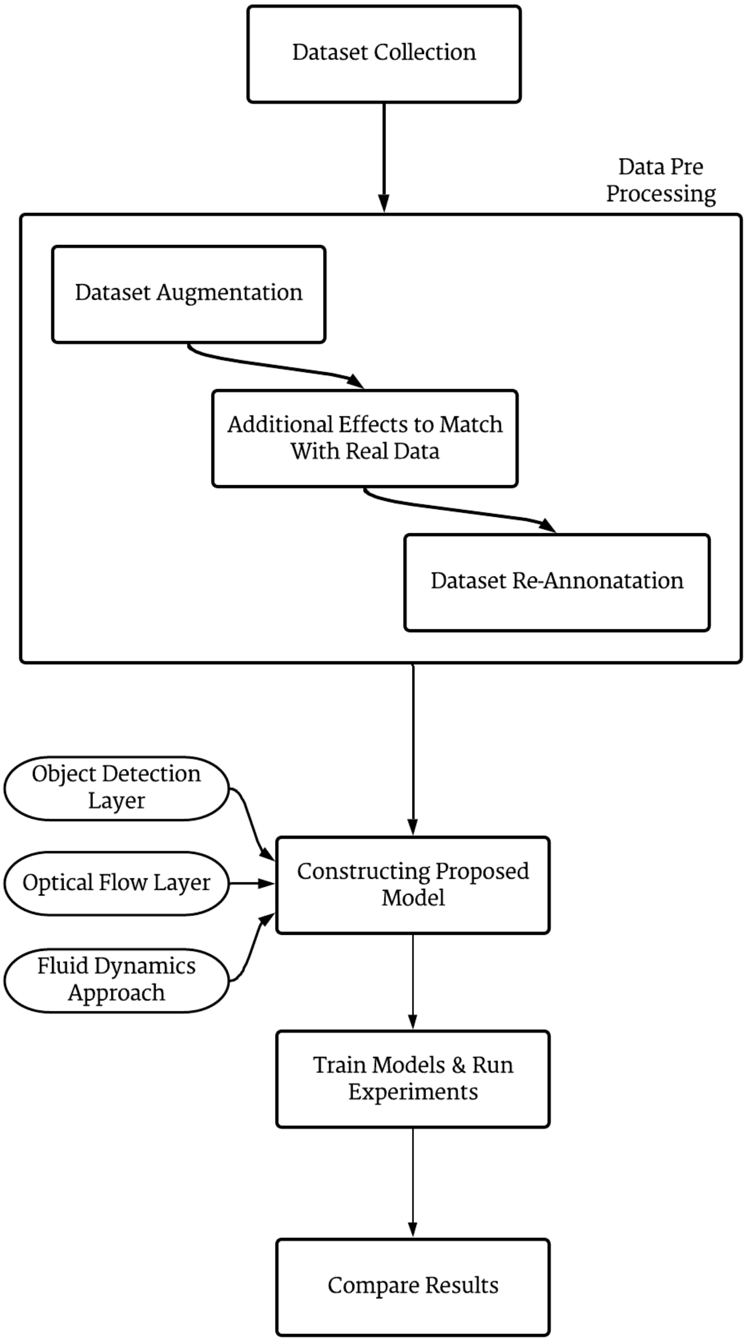

Figure 3.1: Workflow diagram of the proposed model

4. **Adding Flow Detection:** Here, we use the optical flow model to detect the traffic flow, which will help us detect the congestion.

5. **Add the optimization module:** In this stage, we add a LWR traffic flow model

6. **Run Experiments and Comparisons:** In this stage, we run our model in parallel with other image based AI models and leverage the capabilities of our model with others.

3.1 Data Collection

For simulating our ITS, we need data for two of our models. Firstly, to train the object detection model. Secondly, to test the ITS simulation. These two datasets are discussed briefly below.

3.1.1 Object Detection Data

Since we are working to solve a problem regarding the traffic scenario of Bangladesh, commonly used datasets of images or traffic couldn't be used as they were providing incorrect result most of the time. Thus, we had to look deeper to find dataset collected by our countrymen. We collected a dataset that was uploaded by faculties of Green University. This dataset contained images of city traffic in various locations inside the Dhaka division. This particular dataset contained 3503 distinct labeled images of vehicles in various areas of Dhaka city. This is a labelled dataset that contains 21 labels which indicates the vehicle type. The images were labeled using .xml format, which is not suitable for use with YOLOv5[49]. Thus, we had to change the label files to .txt format using an online tool called Roboflow. Following is a brief overview of the dataset original dataset:

Table 3.1: Dataset Summary

Number Of Classes	**Total Images**	**Train Set**	**Test Set**
21	3503	3303	200

3.1.2 Traffic Simulation Data

After implementing our proper model, we need some video data to evaluate the performance of our model. Our goal is to at least reach the accuracy of the IoT models already in existence. For this module, we decided to collect first-hand data by going to several over bridges of Dhaka city and recording 4 sided footage. Since over bridges are usually close to an intersection, it represents a more suitable position of where actual CCTV cameras would be. We opted for video data for this phase, since we will only be doing inferencing to check the accuracy of our model. Following is a brief overview of our collected dataset.

Table 3.2: Collected Data For Inferencing

Videos	Total Video Size(GB)	Compressed Size(GB)	FPS
8	10	1.1	24

3.2 Data Preprocessing

Since this is a dataset collected from the internet, a lot of changes had to be made in order to properly suit with our needs for the proposed system. Also, the dataset require some standard preprocessing anyway due to a lot of noisy or unusable images. The following preprocessing tasks were performed to prepare the data for an optimal condition.

1. **Cropping and Resizing:** The images were captured with either camera or smartphone, the framing of which does not match with CCTV format. Since our goal is applying this model for surveillance systems, we cropped and resized the image to appropriate format, removing unnecessary parts, only keeping the road in focus.

2. **Desaturating the Images:** Most surveillance cameras use monochrome color format. Especially if it used infrared. Thus, to match the dataset closer to our expected inferencing data, we desaturated (turned monochrome) those images that were RGB.

3. **Adding Noise:** To match CCTV image, we added slight amount of noise which represent the color and footage type of surveillance systems.

4. **Re-Annotating the Images:** Since we cropped the images for more focused and narrow field of view, we had to re-annotate some images.

5. **Converting to Text Files:** The bounding box information was recorded in .xml files. To use with YOLOv5, we needed to convert it to .txt file. For

this we used the same online tool, Roboflow, that we also used for our image manipulation in the above steps.

6. **Removing Labels During Annotation:** Instead of keeping all 21 labels, we narrowed it down to only 1. Since our goal is vehicle identification, we just used only 1 label, "vehicle", which will increase the F1-Score of the model. This we further discussed in the Implementation section.

As we can see some preprocessing steps might require us to remove some images which don't provide required information for the model and may decrease the accuracy of the model, after these preprocessing steps our dataset now looks like the following:

Table 3.3: Dataset Summary

Number Of Classes	**Total Images**	**Train Set**	**Test Set**
1	2912	2912	0

Here is the data preprocessing steps shown on a sample image:

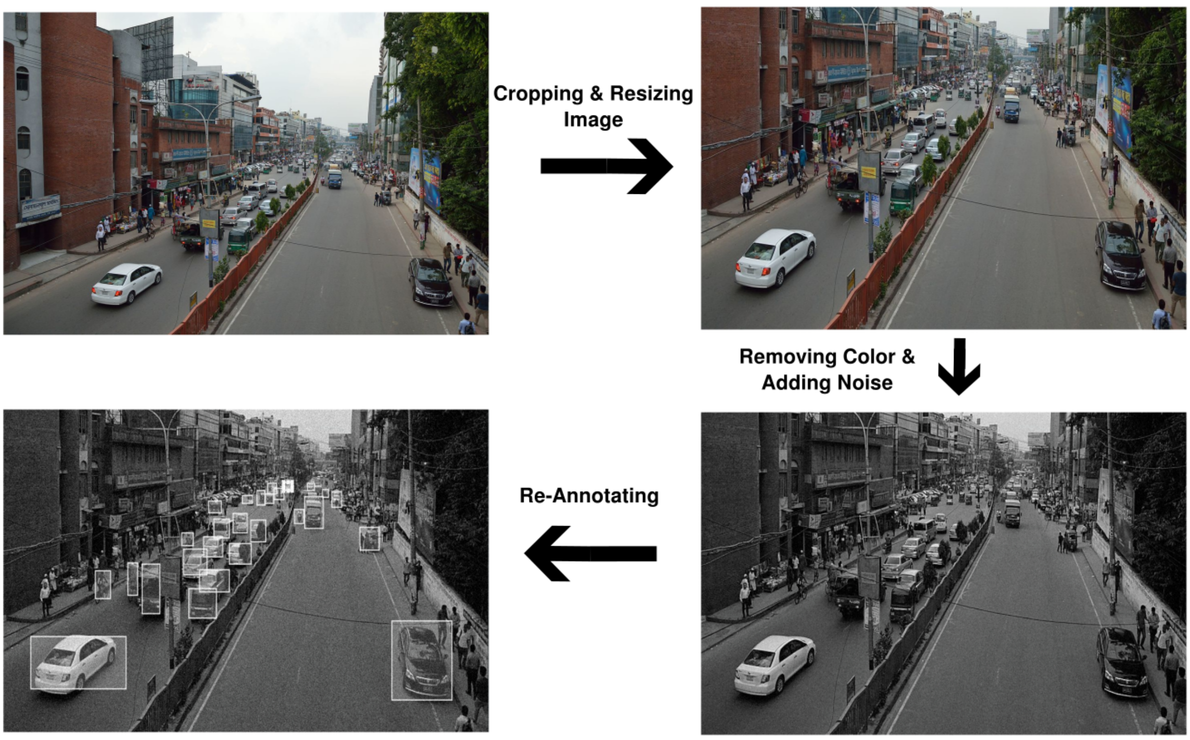

Figure 3.2: Preprocessing Steps Example

Although we chose the whole dataset for our training, we decided to collect primary data later on for the cross validation and inferential statistics.

3.3 Model Architecture

Having a labeled dataset in our hand, we can start training the model through supervised classification. First, we used the benchmark dataset, Coco128, to see how well the model performs. As expected, the inference wasn't accurate enough. Thus, we proceeded with the baseline model and trained it with our dataset. We trained the model on several parameters to get the maximum precision possible. Our constructed model are being discussed below:

1. **Object Detection Layer:** First we need to detect congestion on the road before proceeding further with motion detection and congestion prediction. For this, we used YOLOv5 for vehicle detection. Based on the intensity of the

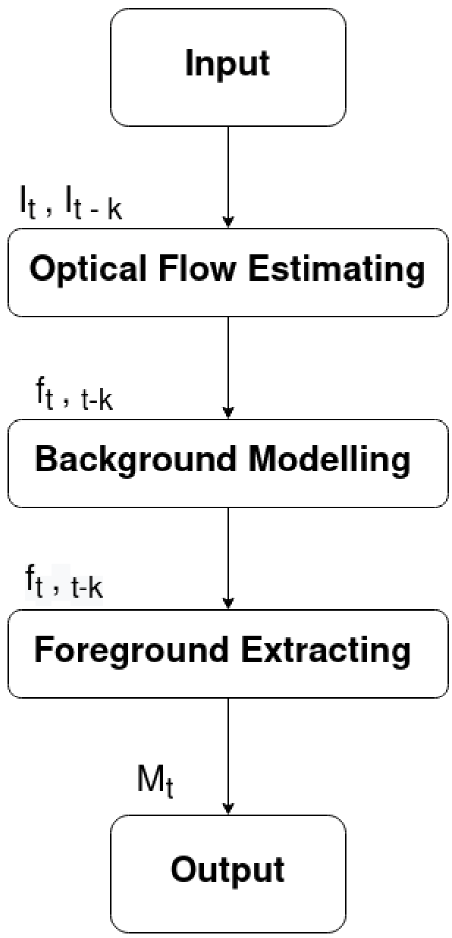

Figure 3.3: Yolov5 Inferencing Steps

detected vehicle, the model decides whether there is congestion on the road or not.

2. **Flow Detection Layer:** If there is congestion, the model then proceeds to detect motion of the vehicle (ultimately the whole congestion). This layer is built using a modified LK Optical Flow model. This can detect vehicle motion by frame differencing and interpolation of pixels. The mathematical formula that is working in LK Optical [23]:

$$f_{t,t-k} = [u \quad v \quad 0]^T \tag{3.1}$$

   Additionally, to determine the swarm success rate, we need the summation of all the vehicles bodies moving in a single frame. Thus, we can get the macroscopic flow of the traffic using the following formulation:

$$SR = \frac{1}{N} \sum_{i=1}^{N} (FM_i > T_{FM}) \tag{3.2}$$

   [23]

3. **Numerical Simulation Model:** For this layer, we will be using the LWR model. The LWR model used the fluid dynamics approach to describe traffic flow in a single road[17]. This scalar hyperbolic conservation law can predict the flow of the traffic. More described below.

3.3.1 Global Speed Detection Algorithm

To detect global speed, we will a use a modified LK Optical Flow algorithm. For that, we will use an algorithm with YOLOv5. There are 2 reasons for choosing this. First, the ease of implementation. We do not have to work on another sophisticated deep learning network. Also, implementing the algorithm is much more convenient for us.
Given the ROI and YOLOv5 optimally detects all the vehicles to an acceptable percentage, we can safely assume with the algorithm above we can detect the speed of the vehicles. The above algorithm is basically a simplified algorithm of the following equation [23]:

$$f(x, y, t = f(x + dx, y + dy, t + dt) \tag{3.3}$$

Global Speed Detection Algorithm [23]
1. Using LK Optical Flow method, we can differentiate the frame's from first frame to the next (from $\pi to \pi + 1$). Using the similar function, we can denote as πr from the position of $\pi + 1$. for i = 1; i ≤ n; i++ do if d = $\lvert \pi - \pi r \rvert <$ Td then: retain pi else: discard pi # construct a coordinate, computing θ with respect to $\pi \rightarrow \pi + 1$ if 90 ≤ 0 ≤ 140, then: reserve corner point else: discard corner point done: # Compute mean value of N frame speed as global vehicle speed V

3.4 Fluid Dynamics Approach

3.4.1 Mathematical Formulation

The LWR model is a widely praised, popular model for traffic flow simulation. It is based on fluid dynamics approach where it defines the relation between density(k), flow(q) and local speed, V, of a single road. The formulation is as follows[17]:

$$\begin{cases} \frac{\delta \rho}{\delta t} + \frac{\delta q}{\delta x} = 0 \\ \rho\left(x, t = 0\right) = \rho_0\left(x\right) \end{cases} \tag{3.4}$$

Here, $\rho \; \epsilon \; [0, \rho_{max}]$ defines the density at place and x and time t, max indicating maximum possible density. Moreover, q also represents the vehicles flux (number of vehicles across a point in a unit time). We can calculate flux using:

$$q(\rho) = \rho v \tag{3.5}$$

v referring to average speed of vehicles

Assuming vehicle speeds get adjusted depending on surrounding conditions, we can formulate v as following:

$$v(\rho) = \left(1 - \frac{\rho}{\rho_{max}}\right) \tag{3.6}$$

v_max represents maximum speed where ρ approaches 0

Therefore, the model approaches to define an initial-value problem (Riemann Problem),

$$\begin{cases} \rho_L, for x \leq x_0 \\ \rho_R, for x > x_0 \end{cases} \tag{3.7}$$

3.4.2 Computational Functions

Described in Erwin B Setiawan et al., [23] considering the finite volume method for solving the LWR, they formulated the stability of the numerical solution using the following, cite49, cite52

$$\delta t \leq min_{j=1,\ldots,M} \frac{\delta x}{|1 - \frac{\rho_j^n}{\rho_{max}}| v_{max} \rho_j^n} \tag{3.8}$$

The probable solutions can be mapped in the following table, the Riemann solution for the LWR model[17],

Table 3.4: Riemann Solution Mapping

Case	**Structure Of The Solution**	$\rho_{j+1/2}^{n+1/2}$
$\rho_j^n \geq \rho_j^n$ & $\rho_j^n < \rho_{max}/2$	Rarefaction wave heading to the right	ρ_j^n
$\rho_j^n \geq \rho_{max}/2$ & $\rho_{j+1}^n \leq \rho_{max}/2$	Rarefaction wave heading to the left	$\rho_{max}/2$
$\rho_j^n \geq \rho_{j+1}^n$ & $\rho_{j+1}^n \geq \rho_{max}/2$	Rarefaction wave to both direction	ρ_{j+1}^n
$\rho_j^n < \rho_{j+1}^n$ & $\rho_j^n + \rho_{j+1}^n \leq \rho_{max}$	Shock wave heading to the right	ρ_j^n
$\rho_j^n < \rho_{j+1}^n$ & $\rho_j^n + \rho_{j+1}^n > \rho_{max}$	Shock wave heading to the left	ρ_{j+1}^n

These 3 modules are key part of our traffic simulation system. Since we solely are relying on the architecture at hand, we strongly assert that the system we are aiming for will not only be highly modular but also provide significant results in various tech stack prepared for real time traffic simulation and intelligent traffic systems[9].

Chapter 4

Implementation and Results

So far, we have discussed how the model works theoretically and approaches that should be taken into account to get optimal results. In this section, we proceed to implement the models practically to build the actual ITS with self adaptive re correction capabilities. We will try to some extent to mimic the behavior, however we won't approach to completely build the model as it is not possible given the resources we had at hand during these implementations.

4.1 Implementation Roadmap of Proposed Models

In this section, we describe how we implemented the proposed ITS to some extent using popular object detection libraries and other python libraries such as PyTorch. We have discussed our implementations methods below. image model training python libraries, PyTorch and Keras. We lay out our roadmap for implementing the model below.

1. **Object Detection Layer Construction:** For this part of the model, we used a popular object detection model called YOLOv5 and trained it on the dataset described in section 3.1.1. We will use online notebooks because of their high GPU capabilities. We will train our model for a certain epoch till we get an acceptable accuracy to proceed to our next module[48].

2. **Flow Detection Layer Selection:** We chose a modified version of the LK Optical Flow model[21] to detect the congestion. For congestion detection we need to learn the traffic flow[19],[38], to learn whether the traffic is moving or not. LK optical flow will be used to detect that traffic flow.

3. **Numerical Simulation Of Traffic Flow:** To identify the relation between flow, density, local speed we will use LWR Traffic Flow model.[17] By using the

finite method for the model that we formulated and described in this section 3.4.2, we can an estimated simulation of traffic and implement the prediction model.[24]

4.2 Object Detection Layer

The first module for our Self Adaptive Intelligent Traffic System is to detect vehicles on the road, as well as the number of cars. This detection will help us find out the number of vehicles as well as the density of the vehicles using the ROI approach. Using the dataset (section 3.1.1, we started training our YOLOv5 model, which is based on a sophisticated Neural Network design.[25]

4.2.1 Training The Object Detection Model

Our goal for this training was to achieve at least 80% detection rate in detecting vehicles of Dhaka City streets from grainy/noisy footage. We obtained the following results on varied epochs. We have laid down our test parameters below, as well as the achieved. Before starting the test, we chose a pre-trained YOLOv5 large model which is trained on COCO128, instead of starting from scratch.
Following are the parameters of our Object Detection Model training

```
train.py --img 640 --batch 16 --epochs 100 --data data.yaml
--weights yolov5l.pt --project vehicle-detection-yolov5 --save-period 10
```

It took us approximately 6 hours to complete all 100 epochs. We logged our data using an online tool, and kept track of the results that we were getting.

The summary of our achieved results are described in the table below 4.2:

Table 4.1: Training Set Results

No. Of Epochs	**mAP**	**Recall**	**Precision**	**Training Loss**
20	75.2%	72.02%	86%	2%
50	80.65%	70.80%	85.92%	1.8%
100	82.11%	73.99%	86.73%	1.8%

A graphical overview of the achieved results are shown in following:

Table 4.2: Validation Set Results

No. Of Epochs	mAP	Recall	Precision
20	70.2%	72.02%	83.33%
50	81.65%	76.80%	85.92%
100	82.11%	74.99%	86.73%

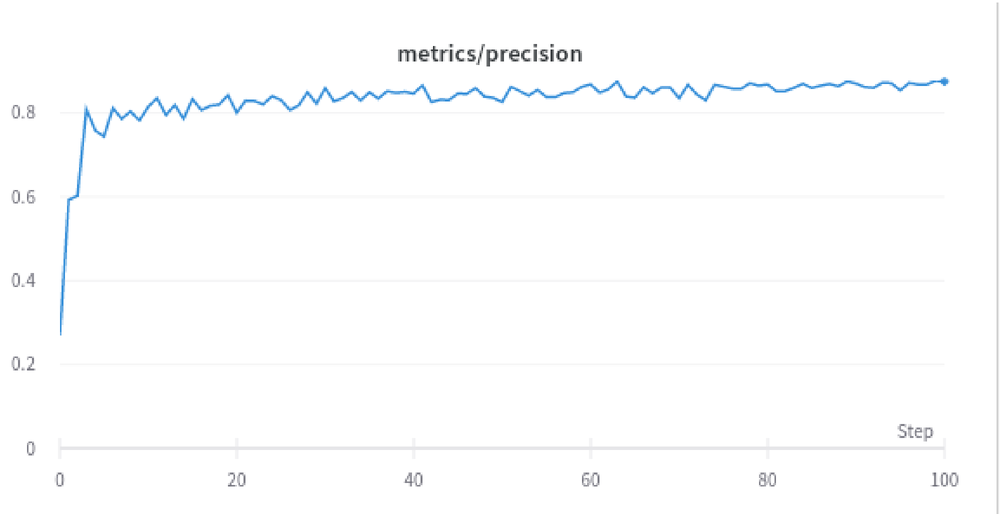

Figure 4.1: Training Precision

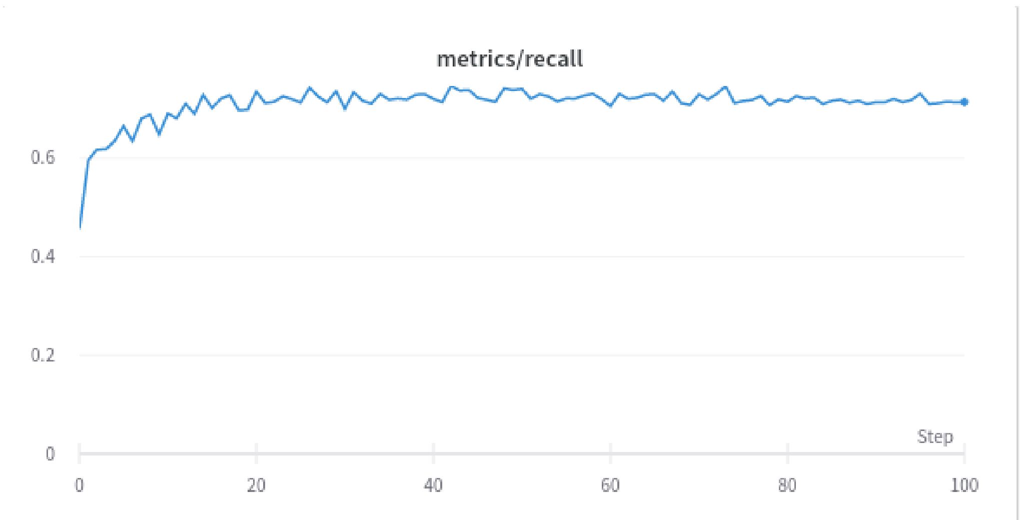

Figure 4.2: Training Recall

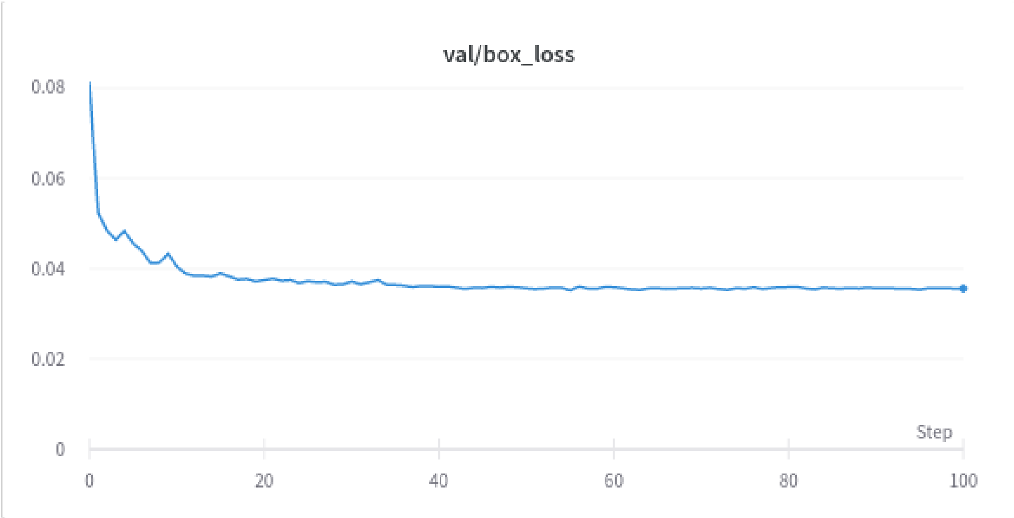

Figure 4.3: Validation Loss

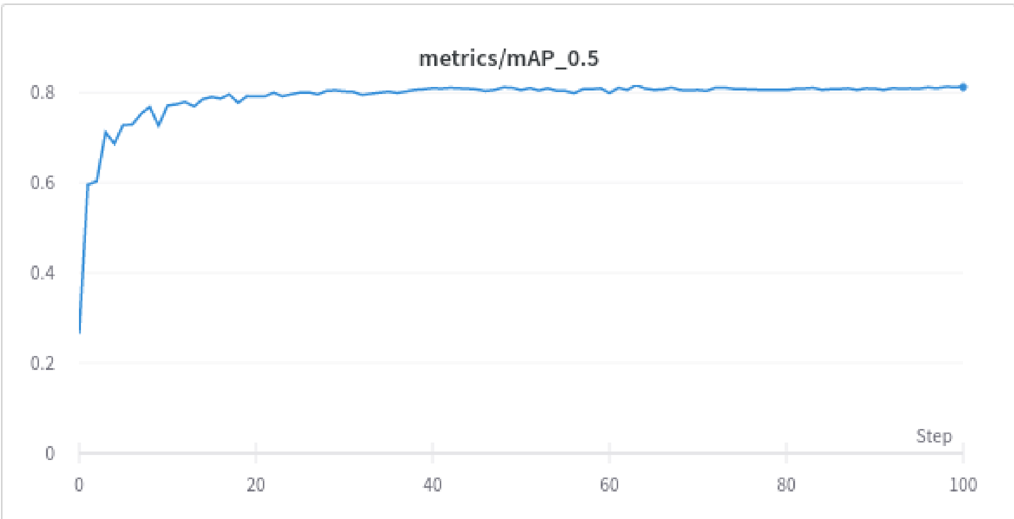

Figure 4.4: Mean Average Precision

After the 100th epoch, we saw a sudden and unexpected drop in accuracy for the next few epochs. But after that the graph stabilized. But we didn't see any better results than the 100th epoch provided. After 10 more epochs, we noticed the validation loss gradient was 0. Meaning, the model wasn't improving any more and after 10 more epochs we noticed an increase in the validation loss graph, indicating that the model has started to overfit. Thus, we proceeded with the model that we received after 100th epoch, which we deem has the best accuracy of 82.11%. The precision gradient over time can be seen in 4.4.

Of course, providing additional data would most certainly improve the F1-score. However, we didn't bother to invest any more of our time for this layer, as greater than 80% is satisfactory for our purpose.

4.2.2 Inferencing and Results

After training, we set on to collect some new data for inferencing and checking if the model provides sufficient detection as we need. It turns out that the trained model works as we planned. We modified the YOLOv5's *detect.py* file slightly. We added the code for ROI and vehicle counting and outputting the values in a file for further use. An example of inferencing working properly using video footage is shown in figure 4.5.

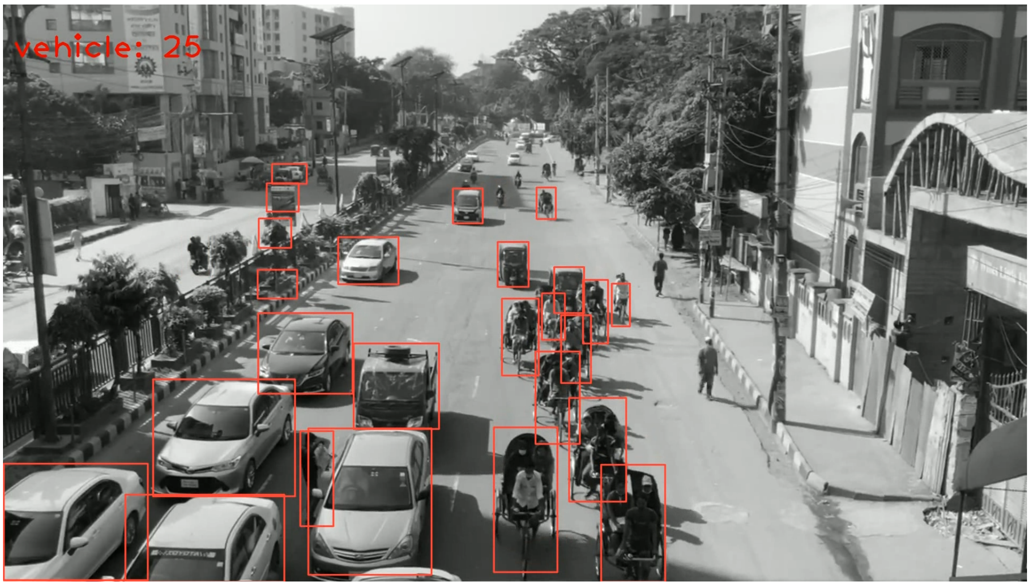

Figure 4.5: Inferencing Results of Trained Model

4.3 Optical Flow Layer

In combination with YOLOv5, LK Optical flow can provide the global speed of vehicles using frame differentiation, It can be more clearly seen in the figure below:

Figure 4.6: Inferencing Results of Trained Model

Using this algorithm along with YOLOv5, we can run some inferencing on our collected dataset, the results are described in comparison with a previous test in table 4.3 we get the demonstration of our achieved results in the figure 4.6.

Table 4.3: Optical Flow Accuracy

Method	Detection Accuracy	Detection Time	Speed Accuracy
Faster R-CNN	81.25%	233ms	85.5%
YOLOv5	82.38%	184ms	86.3%

Visibly, we can detect a vehicle's speed from this data and send this data to Numerical Simulations Layer for further actions. Since, the LWR model requires density and local speed, we have all the necessary data to get an estimated simulation and gather prediction data.

4.4 Numerical Simulations Layer

With the initial problem given Riemann problem, defined in 3.4.2, the convergence of finite volume method for solving LWR model, we can calculate the norm error values from the formulation provided by Erwin B. Setiwan et al., where L^2 is the error ratios,

$$||\rho(x,t) - \rho_{exact}(x,t)||_{L^2} = (\int_0^l [\rho(x,t) - \rho_{exact}(x,t)]^2 \, dx) \tag{4.1}$$

The benefit of calculating the L^2 norm values is that we can get an estimation of errors before hand. By calculating the error we can adjust the offset of our model. This will widely vary based on the results the previous 2 layers provide.

4.4.1 Test Results

In this first test, we try to simulate the traffic light triggered shock wave. We took data from our first video (following is a snapshot of the image) where we estimated the road length to be approximately 150 m. Globally we consider x = 0 m and cars are coming from 150 m. Since the road is empty, the traffic light is currently at red. From the data, we found $\rho_max = 0.25v/m$ $v_max = 30km/h$. Vehicle density is calculated by calculating the area from the ROI. Now we have ρ_L, ρ_R. Therefore, we can use the following formulation to get an estimated simulation.[17]

$$\begin{cases} \rho_L, if(x-x_0)/t < c_s \\ \rho_R, if(x-x_0)/t > c_s \end{cases} \tag{4.2}$$

Here. $c_s = [q(\rho_L) - q(\rho_R)]/(\rho_L - \rho_R)$

If we run this consecutively for our test dataset, we can see how the traffic signals work during different traffic load. Based on the achieved ration, the traffic lights will act accordingly to adjust traffic flow.

$$\begin{cases} \rho_L, if(x-x_0)/t < \lambda(\rho_L) \\ \frac{1}{2}\rho_{max}(1 - \frac{x-x_0}{v_{max}t}), if\lambda(\rho_L) < (x-x_0)/t < \lambda(\rho_L) \\ \rho_R, if(x-x_0)/t > \lambda(\rho_L) \end{cases} \tag{4.3}$$

Here, $\lambda(\rho) = (1 - 2\rho/\rho_{max})v_{max}$

We have recorded our test results in table 4.4:

Table 4.4: L^2 norm values

M	$\|\|\rho_{num} - \rho_{exact}\|\|_L^2$	**Ratio**
20	0.135	-
40	0.100	0.36
80	0.903	0.48
100	0.711	0.56

In the figure 4.7, we have tried to show a simplified version of our recorded results. We have used the finite volume method against the exact solution. As you can notice in the figure that it varies significantly when shock waves and rarefaction waves are registered.

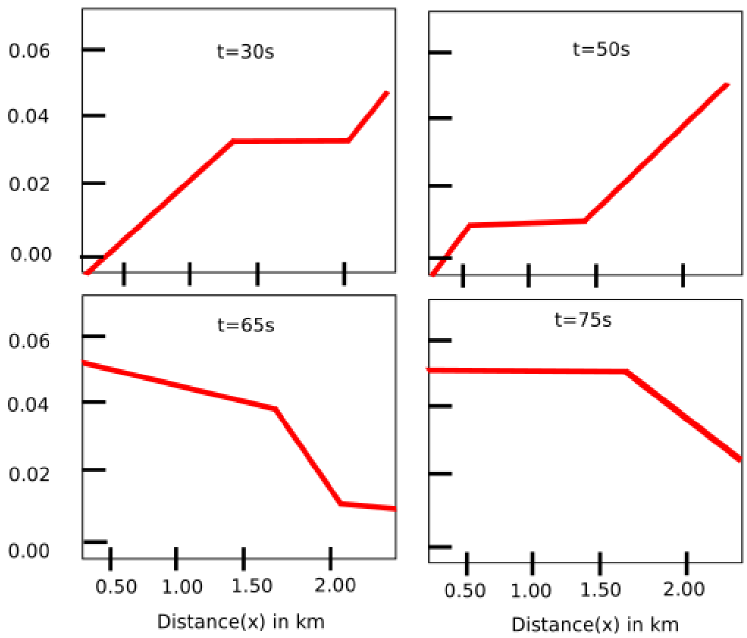

Figure 4.7: Testing Final Volume Method Against Exact Solution

These results are a numeric representation of what we would call congestion and flow. Of course, it does not seem as intuitive at first glance because it is solely purposed for machines to operate on this data. Additionally, we will have a closer look at how the traffic light simulation will work in the next section.

4.5 Traffic Light Simulation

If we were to theoretically explain how the light simulation works, we first assume a scenario in a single lane road where initial density is ρ_0, approaching the traffic light, the cars stop. gradually over time, we see an increasing number of cars, and our YOLOv5 model signals that it detects congestion because the vehicle numbers have passed a certain value. First, let us define the initial variables in table 4.5,

Table 4.5: Simulation Parameters

Variable	Parameter
l	Length of computation domain
ρ_l	Vehicles flux
v_{max}	Max Speed
v_R	Vehicle Speed
ρ_0	Initial Density
ρ_{max}	Max Density
δ_t	Time Step
δ_x	Cell Size

Assume a hypothetical sceniario such as the figue 4.8At ρ_0, we have our traffic signal giving red. Because our object detection layer is not detecting any congestion. As time passes, $\rho_0 \rightarrow \rho_{max}$, our object detector model wakes up the optical flow model for a scan. Now, if the cars are to flow at a given satisfactory speed, it is safe to assume that the pile up hasn't occurred yet. However, given, the optical flow layer detects low speed, meaning $v_r = 0$, then the numerical simulations model will be triggered.

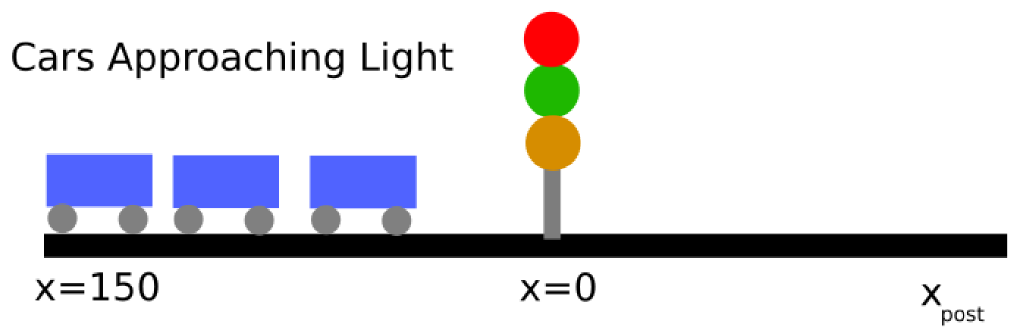

Figure 4.8: Testing Final Volume Method Against Exact Solution

With the trigger of the numerical simulation model, it gathers the real time data from other traffic around it. If the other lights have been dormant for a while, it seeks for the latest data. With the data collected, the simulation model can simulate traffic flow and seek for $v_R = v_{max}$ and $\rho_{max} \rightarrow \rho_0$.
We can expand this problem to more than one traffic light without any major complications. These traffic lights can communicate with each other and will try to find

out the most optimal route for each case. No, traffic light will show green at the same time because, it will only react to the lane with maximum flow.

Chapter 5

Conclusion

Excessive congestion is not uncommon in countries such as Bangladesh, which has a massive community and a great amount of automobiles. Road congestion not just causes financial loss, but it also has a detrimental influence on human health and wastes a huge amount of resources. Under such conditions, implementing an adaptive intelligent traffic system for handling this problem would be a huge 'quality of life' benefit for all metropolitan residents. Another important issue is that increased computer capacity has permitted the creation of a plethora of AI-based models that can already execute at a significantly quicker rate. Undoubtedly, AI-based deep learning techniques have made significant progress in tackling such types of complex challenges. Even with that in mind, this research article outlines our endeavor to create a self-adaptive ITSC system that can regulate traffic intelligently based on estimates of traffic conditions in the nearish term. As a consequence, it can control traffic by delivering the appropriate set of instructions via traffic signals and giving precedence to crowded lanes. Finally, we compare different AI-based methods with statistics-based approaches in an attempt to discover the best model for our proposed model.

5.1 Future Works

In the future, we plan to extend our work by implementing a centralized AI model, preferably based on Deep learning network that can take instantaneous decision based on the data that the traffic lights are providing which are based on numerical simulations. We have looked at such models and also found some interest in Google's graph neural network technology suggested by our faculty. Such a robust system can be applicable anywhere, in any infrastructure, and can later be swapped with smart city features if required. We plan to extend our dataset and realm of testing to make our already established plan more robust, as well as centralized AI hive mind technology to predict and optimize traffic flow in the whole city.

Bibliography

[1] P. Hunt, D. Robertson, R. Bretherton, and M. C. Royle, "The scoot on-line traffic signal optimisation technique," *Traffic Engineering & Control*, vol. 23, no. 4, 1982.

[2] P. Lowrie, "Scats, sydney co-ordinated adaptive traffic system: A traffic responsive method of controlling urban traffic," 1990.

[3] S. Mikami and Y. Kakazu, "Genetic reinforcement learning for cooperative traffic signal control," in *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, IEEE, 1994, pp. 223–228.

[4] E. Bingham, "Reinforcement learning in neurofuzzy traffic signal control," *European Journal of Operational Research*, vol. 131, no. 2, pp. 232–241, 2001.

[5] M. C. Choy, D. Srinivasan, and R. L. Cheu, "Hybrid cooperative agents with online reinforcement learning for traffic control," in *2002 IEEE World Congress on Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems. FUZZ-IEEE'02. Proceedings (Cat. No. 02CH37291)*, IEEE, vol. 2, 2002, pp. 1015–1020.

[6] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *Journal of Transportation Engineering*, vol. 129, no. 3, pp. 278–285, 2003.

[7] D. de Oliveira, A. L. Bazzan, B. C. da Silva, *et al.*, "Reinforcement learning based control of traffic lights in non-stationary environments: A case study in a microscopic simulator.," in *EUMAS*, 2006.

[8] B. Silva, D. d. Oliveira, A. Bazzan, and E. Basso, "Adaptive traffic control with reinforcement learning," in *Proceedings of the 4th Workshop on Agents in Traffic and Transportation (AAMAS 2006)(May 2006), ALC Bazzan, BC Draa, and F. Kl, Eds*, 2006, pp. 80–86.

[9] J. J. S. Medina, M. J. G. Moreno, and E. R. Royo, *Evolutionary computation applied to urban traffic optimization*. IntechOpen, 2008.

[10] L. Xiao and Z. Wang, "Internet of things: A new application for intelligent traffic monitoring system," *Journal of networks*, vol. 6, no. 6, p. 887, 2011.

[11] L. Hernández, C. Baladrón, J. M. Aguiar, *et al.*, "A study of the relationship between weather variables and electric power demand inside a smart grid/smart world framework," *Sensors*, vol. 12, no. 9, pp. 11 571–11 591, 2012.

[12] D. Stefanoiu, J. Culita, and F. Tudor, *Experimental approaches in process and phenomena identification*, 2012.

[13] M. S. Ali, M. Adnan, S. M. Noman, and S. F. A. Baqueri, "Estimation of traffic congestion cost-a case study of a major arterial in karachi," *Procedia Engineering*, vol. 77, pp. 37–44, 2014.

[14] A. Nkoro and Y. A. Vershinin, "Current and future trends in applications of intelligent transport systems on cars and infrastructure," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2014, pp. 514–519.

[15] P. S. Paul *et al.*, "Traffic density estimation and flow control for video surveillance system," Ph.D. dissertation, BRAC University, 2014.

[16] Y. Yuan, H. Van Lint, F. Van Wageningen-Kessels, and S. Hoogendoorn, "Network-wide traffic state estimation using loop detector and floating car data," *Journal of Intelligent Transportation Systems*, vol. 18, no. 1, pp. 41–50, 2014.

[17] E. B. Setiawan, D. Tarwidi, and R. F. Umbara, "Numerical simulation of traffic flow via fluid dynamics approach," *International Journal of Computing and Optimization*, no. 3, pp. 93–104, 2015.

[18] Y. Ding and X. Fu, "Kernel-based fuzzy c-means clustering algorithm based on genetic algorithm," *Neurocomputing*, vol. 188, pp. 233–238, 2016.

[19] K. Nellore and G. P. Hancke, "A survey on urban traffic management system using wireless sensor networks," *Sensors*, vol. 16, no. 2, p. 157, 2016.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[21] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4161–4170.

[22] J. Sochor, R. Juránek, and A. Herout, "Traffic surveillance camera calibration by 3d model bounding box alignment for accurate vehicle speed measurement," *Computer Vision and Image Understanding*, vol. 161, pp. 87–98, 2017.

[23] J. Huang, W. Zou, J. Zhu, and Z. Zhu, "Optical flow based real-time moving object detection in unconstrained scenes," *arXiv preprint arXiv:1807.04890*, 2018.

[24] ——, "Optical flow based real-time moving object detection in unconstrained scenes," *arXiv preprint arXiv:1807.04890*, 2018.

[25] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[26] B. Deb, S. R. Khan, K. T. Hasan, A. H. Khan, and M. A. Alam, "Travel time prediction using machine learning and weather impact on traffic conditions," in *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)*, IEEE, 2019, pp. 1–8.

[27] E. Ferrara, L. Fragale, G. Fortino, *et al.*, "An ai approach to collecting and analyzing human interactions with urban environments," *IEEE Access*, vol. 7, pp. 141 476–141 486, 2019.

[28] W. Ghazali, C. Zulkifli, and Z. Ponrahono, "The effect of traffic congestion on quality of community life," in *4th International Conference on Rebuilding Place. The European Proceedings of Multidisciplinary Sciences*, 2019, pp. 759–766.

[29] W. H. Money and S. Cohen, "Leveraging ai and sensor fabrics to evolve smart city solution designs," in *Companion Proceedings of The 2019 World Wide Web Conference*, 2019, pp. 117–122.

[30] D. Impedovo, F. Balducci, V. Dentamaro, and G. Pirlo, "Vehicular traffic congestion classification by visual features and deep learning approaches: A comparison," *Sensors*, vol. 19, no. 23, p. 5213, 2019.

[31] A. Miglani and N. Kumar, "Deep learning models for traffic flow prediction in autonomous vehicles: A review, solutions, and challenges," *Vehicular Communications*, vol. 20, p. 100 184, 2019.

[32] ——, "Deep learning models for traffic flow prediction in autonomous vehicles: A review, solutions, and challenges," *Vehicular Communications*, vol. 20, p. 100 184, 2019.

[33] S. Ramteke and B. Gite, "Ai based traffic signal control system," *Traffic*, vol. 6, no. 11, 2019.

[34] W. Zhang, Y. Yu, Y. Qi, F. Shu, and Y. Wang, "Short-term traffic flow prediction based on spatio-temporal analysis and cnn deep learning," *Transportmetrica A: Transport Science*, vol. 15, no. 2, pp. 1688–1711, 2019.

[35] T. Bogaerts, A. D. Masegosa, J. S. Angarita-Zapata, E. Onieva, and P. Hellinckx, "A graph cnn-lstm neural network for short and long-term traffic forecasting based on trajectory data," *Transportation Research Part C: Emerging Technologies*, vol. 112, pp. 62–77, 2020.

[36] J. Culita, S. I. Caramihai, I. Dumitrache, M. A. Moisescu, and I. S. Sacala, "An hybrid approach for urban traffic prediction and control in smart cities," *Sensors*, vol. 20, no. 24, p. 7209, 2020.

[37] K. S. Desikan, V. J. Kotagi, and C. S. R. Murthy, "Topology control in fog computing enabled iot networks for smart cities," *Computer Networks*, vol. 176, p. 107 270, 2020.

[38] A. Essien, I. Petrounias, P. Sampaio, and S. Sampaio, "A deep-learning model for urban traffic flow prediction with traffic events mined from twitter," *World Wide Web*, pp. 1–24, 2020.

[39] R. Ke, W. Li, Z. Cui, and Y. Wang, "Two-stream multi-channel convolutional neural network for multi-lane traffic speed prediction considering traffic volume impact," *Transportation Research Record*, vol. 2674, no. 4, pp. 459–470, 2020.

[40] J. J. Vázquez, J. Arjona, M. Linares, and J. Casanovas-Garcia, "A comparison of deep learning methods for urban traffic forecasting using floating car data," *Transportation Research Procedia*, vol. 47, pp. 195–202, 2020.

[41] J. Wang, J. Li, S. Yan, *et al.*, "A novel underwater acoustic signal denoising algorithm for gaussian/non-gaussian impulsive noise," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 1, pp. 429–445, 2020.

[42] P. Xie, T. Li, J. Liu, S. Du, X. Yang, and J. Zhang, "Urban flow prediction from spatiotemporal data using machine learning: A survey," *Information Fusion*, vol. 59, pp. 1–12, 2020.

[43] C. Zhang, "Design and application of fog computing and internet of things service platform for smart city," *Future Generation Computer Systems*, vol. 112, pp. 630–640, 2020.

[44] R. Zhang, A. Ishikawa, W. Wang, B. Striner, and O. K. Tonguz, "Using reinforcement learning with partial vehicle detection for intelligent traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[45] A. H. Ahmed and L. Z. Fragonara, "Adaptive intelligent traffic control systems for improving traffic quality and congestion in smart cities," 2021.

[46] M. Akhtar and S. Moridpour, "A review of traffic congestion prediction using artificial intelligence," *Journal of Advanced Transportation*, vol. 2021, 2021.

[47] S. Majumdar, M. M. Subhani, B. Roullier, A. Anjum, and R. Zhu, "Congestion prediction for smart sustainable cities using iot and machine learning approaches," *Sustainable Cities and Society*, vol. 64, p. 102 500, 2021.

[48] X. Yang, F. Wang, Z. Bai, F. Xun, Y. Zhang, and X. Zhao, "Deep learning-based congestion detection at urban intersections," *Sensors*, vol. 21, no. 6, p. 2052, 2021.

[49] F. Zhou, H. Zhao, and Z. Nie, "Safety helmet detection based on yolov5," in *2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA)*, IEEE, 2021, pp. 6–11.