

Modeling Head-on Crash Probability on Two-Lane Undivided Highway using Vision-Based Classified Trajectory

by

Nazmul Haque

A Thesis Submitted to the Department of Civil Engineering in Partial
Fulfillment of the Requirement for the Degree of
MASTER OF SCIENCE IN CIVIL ENGINEERING (TRANSPORTATION)




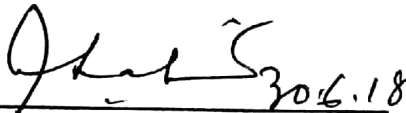
Department of Civil Engineering
**BANGLADESH UNIVERSITY OF ENGINEERING AND
TECHNOLOGY (BUET)**


June, 2018


The thesis titled “**Modeling Head-on Crash Probability on Two-Lane Undivided Highway using Vision-Based Classified Trajectory**” submitted by Nazmul Haque, Student number 0416042411 P and session April, 2016 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of **Master of Science in Civil Engineering (Transportation)** on 30th June, 2018.

Board of Examiners

- | | |
|---|--------------------------|
| 

(1) Dr. Md. Hadiuzzaman
Associate Professor
Department of Civil Engineering,
BUET, Dhaka - 1000 | Chairman
(Supervisor) |
| 

(2) Dr. Ahsanul Kabir
Professor and Head
Department of Civil Engineering,
BUET, Dhaka - 1000 | Member
(Ex-officio) |
| 

(3) Dr. Md. Mizanur Rahman
Professor
Department of Civil Engineering,
BUET, Dhaka - 1000 | Member |
| 

(4) Dr. Farzana Rahman
Professor
Department of Civil Engineering,
University of Asia Pacific, Dhaka - 1205 | Member
(External) |

Declaration

It is hereby declared that the work presented in this thesis has been done by the author that full or part of this thesis has not been submitted elsewhere for the award of any degree or diploma.

June, 2018.

Nazmul Haque.

Nazmul Haque

Student No. 0416042411

ABSTRACT

Head-on crash on rural undivided highways has become most inevitable event now-a-days. Thousands of people are either losing their life or accepting life-long disability in these crashes due to lack in traffic safety; whereas traffic safety aims at the reduction of fatalities causing from crashes among road users. However, this aim of traffic safety cannot be achieved without proper understanding of the crash mechanism. Traffic safety analysis requires historical records of crash data and these crash data lack in availability and quality. Furthermore, to prevent crashes by using historical records is a reactive approach requiring large amount of crash data. Hence, these problems related to crash data motivate the development of surrogate measures of safety. Surrogate safety measures are based on crash probability rather than on the observation of actual crashes. Thus, this thesis endeavors to develop a model that estimates head-on crash probability from vision based classified vehicle trajectory.

The crash probability estimation model formulation considered: (1) drivers' overtaking decision (OD); and (2) time-to-collision (TTC) on two-lane undivided highway. Drivers' overtaking decision was modeled using nonlinear random parameter multivariate binary logistic regression. It considered variables related to both traffic (i.e. vehicle speed and spacing) and drivers' characteristics (i.e. aggressiveness). In contrast, TTC was determined using a new formulation that considered the dynamic acceleration of the vehicles in addition to their speed and spacing. Incorporation of two new parameters, i.e. overtaking importance factor (OIF) and crash frequency parameter (CFP) enabled the estimation of crash probability combining OD and TTC. However, calibration of these models (OD and TTC) requires high frequency and well-structured vehicle trajectory data. In this regard, background subtraction technique along with Kalman filter was used to obtain vehicle trajectories from real-time video. Background subtraction technique was applied using a newly developed background estimation model. A number of theories were proposed to define different components of a video image. Specifically, first-order model for illumination variation and Fourier series for incorporating traffic arrival patterns were considered to define background and foreground, respectively. These definitions were utilized to formulate the traffic detection problem and subsequently three adaptive dynamic background models were developed to solve it. The third model, which incorporates both luminance and pollution controlling parameters addresses the traffic detection problems and limitations faced by the first and second models. This final model consists of two parameters: (1) luminance controlling parameter and (2) pollution controlling parameter. Furthermore, this study reveals newly discovered 'ghost' formation due to taking geometric mean of background and the subject frame, which has been termed as transparency

effect. Foreground segmentation was done to get a binary image. Foreground segmentation uses a new heuristic dynamic threshold-difference ($\tau - \delta$) function for determining per pixel threshold. The shadow of the vehicle was removed considering its physical characteristics by newly presented PNS (Positive Negative Segmentation) technique. Impulse flow waves and aggregated pictorial speed were computed after shadow removal. Impulse flow waves were eventually rectified and cumulated into actual flow. On the other hand, pictorial speed was converted into actual speed using calibration equation considering perspective error. After the foreground segmentation, connected component analysis is applied to find the geometric properties (i.e. centroid, area) of the detected object. Kalman filter was applied to get the tracking data from the detected object. This tracking data is aggregated into trajectory by means of data processing algorithm. Three different types of data were collected for this thesis work. The first one consists of six videos for calibration and validation of the background model. These videos contains a mixture of mild to hard challenges such as gradual to sudden illumination variation, stop and go traffic situation. In the second one, three different locations in Dhaka city were chosen to validate the traffic measurement mechanism. In the third one, a video (9000 sec) was captured from a two-lane undivided rural highway containing high speed uninterrupted vehicles. To avoid detailed object detection, the mounting height of the cameras was kept at 20ft and their angle was less than 45 degrees in each of the cases.

Variable inputs required for calibrating the OD model were generated by constructing adjacency matrices among the detected vehicles from the third video. Analysis over these inputs shows that, lower front vehicle speed invokes the overtaking maneuver. Moreover, the bus and car drivers are found to be more aggressive drivers while overtaking. Exploiting these inputs, Metropolis-Hastings algorithm was applied to obtain calibrated parameters of the OD model for different classes of vehicle. Calibration result shows that subject vehicle speed and the subject-opposing spacing are the most significant variables influencing the overtaking decision on two-lane undivided highway. In another way, lead vehicle speed and subject vehicle aggressiveness also influences the overtaking decision largely. Opposing vehicle speed found to be least influencing in making overtaking decision. Besides, the maximum head-on crash probability for different types of vehicles while completing overtaking maneuver were determined and it was found that bus has the largest one. Finally, the nomographs established in this thesis ensures easy determination of the crash probability.

ACKNOWLEDGEMENTS

All praise is due to the Almighty, the most merciful and the most beneficent.

I would like to express deepest gratitude and indebtedness to my supervisor, Dr. Md. Hadiuzzaman, Associate Professor, Department of Civil Engineering, BUET, Dhaka for his guidance, encouragement and continuous support throughout the progress of the work.

Sincere appreciation goes to the members of my M.Sc. defense committee: Dr. Ahsanul Kabir, Dr. Md. Mizanur Rahman and Dr. Farzana Rahman for their thoughtful question, valuable comments and suggestions.

I would like to acknowledge the research grant received for this study from the Committee for Advanced Studies and Research (CASR) of BUET, Dhaka.

I am also thankful to Fahmida Rahman, Farhana Mozumder Lima and Md. Rayeedul Kalam Siam, Research Assistant, Department of Civil Engineering, BUET for his extensive support and help.

Finally and most importantly I am grateful to my parents for their love, concern, care and faith without which this study would have been impossible.

Table of Contents

ABSTRACT	iii
ACKNOWLEDGEMENTS	v
Table of Contents	vi
List of Figures	ix
List of Tables	xi
Glossary	xii
Chapter 1 Introduction	1
1.1. Background of the study	1
1.2. Statement of the problems and opportunities	3
1.2.1. Absence of high-resolution data collection technique	3
1.2.2. Absence of perfect background model for vehicle detection	3
1.2.3. Inadequate Statistical Model Infrastructure	3
1.2.4. Existing Perilous Time-to-Collision Formulation	3
1.3. Research objectives and scope of work	4
1.4. Organization of the thesis	4
Chapter 2 Literature Review	6
2.1. Introduction	6
2.2. Overtaking Models	6
2.2.1. Driving Simulators	6
2.2.2. Connected Vehicle Environment	8
2.2.3. Microscopic Simulation	9
2.3. Time-to-Collision	10
2.4. Background Modeling	10
2.5. Vehicle Detection	12
2.6. Vehicle Trajectory	15
2.7. Summary	16
Chapter 3 Background Modeling	17
3.1. Introduction	17
3.2. Related Definitions	17
3.2.1. Understanding Frame, I_n	17
3.2.2. Defining Background (B) and Foreground (F)	18
3.2.3 Understanding the background, B_n	20
3.2.4 Understanding <i>the foreground</i> , F_n	22

3.3. Problem Formulation	23
3.4. Solution Models	26
3.4.1. Model 1(M1)	26
3.4.2. Model 2 (M2).....	29
3.4.3. Model 3 (M3)	30
3.5. Summary	31
Chapter 4 Pixel Based Heterogeneous Traffic Measurement	32
4.1. Introduction.....	32
4.2. Methodology for Traffic Measurement.....	33
4.2.1. Live Video Capture Module.....	34
4.2.2. User Input Module.....	34
4.2.3. Background Estimation Module.....	36
4.2.4. Vehicle Detection Module.....	38
4.2.5. Shadow Removal Module	39
4.2.6. Flow Measurement Module.....	40
4.2.7. Speed Measurement Module.....	41
4.3. Basic Features of PARTS.....	42
4.4. Summary	44
Chapter 5 Crash Probability Estimation	45
5.1. Introduction.....	45
5.2. Live Video Capture	45
5.3. Detection	46
5.4. Trajectory Estimation.....	46
5.5. Vehicle Classification	47
5.6. Model Formulation and calibration procedure.....	48
5.6.1. Overtaking Decision Model (OD).....	48
5.6.2. Time-To-Collision (TTC).....	50
5.6.3. Crash Probability Estimation.....	51
5.7. Summary	52
Chapter 6 WINBUGS: Marcov Chain Monte Carlo	53
6.1. Introduction.....	53
6.2 MCMC Algorithm.....	53
6.3. Markov Chain Mote Carlo Simulation using WinBUGS.....	54
6.4. Summary	61
Chapter 7 Data Collection and Analysis	62
7.1. Introduction.....	62

7.2. Data Collection.....	62
7.2.1. Data Collection for Background Modelling.....	62
7.2.2. Data Collection for Real-Time Traffic Measurement	63
7.2.3. Data Collection for Crash Probability Estimation.....	63
7.4. Data Analysis	64
7.5. Summary	68
Chapter 8 Calibration and Validation	69
8.1 Introduction.....	69
8.2. Calibration and Validation of Background Model.....	69
8.1.1. Performance Indicators.....	69
8.1.2. Parameter Calibration.....	70
8.1.3. Validation	74
8.2. Field Testing of Model M3	75
8.3. Field Testing of PARTS.....	77
8.4. Model Calibration Procedure	80
8.5. Summary	81
Chapter 9 Results and Discussion.....	82
9.1. Introduction.....	82
9.2. Correlation among the parameters	82
9.3. Chart for Practical Use	89
9.4. Summary	91
Chapter 10 Conclusion and Recommendations	92
10.1. Concluding Remarks.....	92
10.2. Limitations of the Study.....	97
10.3. Recommendations for Future Research	98
REFERENCES.....	99

List of Figures

Figure 2.1 Overtaking taking probability of male and female in different context.	8
Figure 2.2 A glimpse of driving simulator.....	8
Figure 2.3 Phases of an overtaking maneuver in DSRC system.....	9
Figure 3.1 Expressing actual background B_t using static background B^* and linear illumination variation $\left(\frac{dK_t}{dt}\right)_t$	22
Figure 3.2 Representation of traffic F_t	23
Figure 3.3 Traffic detection using different background models.....	25
Figure 3.4 Representation of residue of the lost \bar{F} due to accumulation of vehicle.	28
Figure 4.1 Connectivity among different components of vision based wireless surveillance	33
Figure 4.2 Flow chart showing different modules incorporated in PARTS tool.	35
Figure 4.3 User Input Module.....	36
Figure 4.4 Rectifying and cumulating inflow waves for flow measurement.....	41
Figure 4.5 Measuring speed from displacement of object centroid at every segment.....	42
Figure 4.6 Interface of PARTS showing panels and menus.	43
Figure 5.1 Flow chart showing the methodology of crash probability estimation	45
Figure 5.2 Definition of subject vehicle and surrounding vehicles	51
Figure 6.1 Specification Tool.....	55
Figure 6.2 Checking the syntax.....	55
Figure 6.3 Debugging the model.....	56
Figure 6.4 Model Correctness.....	57
Figure 6.5 Data input.	57
Figure 6.6 Tabular Data Input.....	57
Figure 6.7 Sample Data structure.....	58
Figure 6.8 Data Loading.	58
Figure 6.9 Model Compilation.....	59
Figure 6.10 Loading Initial values.	59
Figure 6.11 Model Initialization.	60
Figure 6.12 Updating the model.	60
Figure 6.13 Setting number of iterations.	60
Figure 6.14 Setting parameter observer.	61
Figure 7.1 Traffic dataset.	62
Figure 7.2 Traffic measurement dataset.....	63
Figure 7.3 Dataset for crash probability estimation.....	64
Figure 8.1 Parameters optimization.	71
Figure 8.2 Optimum threshold for optimum luminance and pollution controlling parameters.....	72
Figure 8.3 The effect of normalized parameter on performance indicator considering $\delta_c \in [0, 255]$, $r_l \in [0, 1]$, $N_r = [0, 1700]$, $N_l = [0, 1700]$ and $\tau_{max} = [0, 3000]$	74
Figure 8.4 Field testing of M3 in different locations.	76

Figure 8.5 Performance evaluation of PARTS-based flow and speed measurement at different locations; (a)-(b) Urban intersection; (c)-(d) Urban arterial link; and (e)-(f) Rural highway.....	78
Figure 9.1 Change in probability with explanatory variables.....	89
Figure 9.2 Nomographs.....	91

List of Tables

Table 7.1 Traffic and driver characteristics while overtaking	67
Table 8.1 Foreground segmentation using optimized parameters.....	74
Table 8.2 Statistical evaluation of PARTS-based measurement with respect to ground truth	79
Table 9.1 Correlation among the considered variable.....	82
Table 9.2 Values of the calibrated parameters (β, ω).	84

Glossary

Term	Definition
Adjacency Matrices	It is a square matrix used to represent a finite graph.
Background	An image frame which remains stationary for a considerable period of time.
Background Model	A model which estimates the background image.
Binary Image	The image having its pixel value either 0 or 1.
Buffer	A space in the memory, which is allocated dedicatedly.
Camera Jitter	Vibration of camera.
Camouflage	The foreground having pixels values and pattern near to the background.
Crashes	An incident in which a vehicle collides with another vehicle, pedestrian, animal, road debris, or other stationary, obstruction (such as a tree, pole or building) resulting in injury, death, and property damage.
Field of Vision	The area covered by the camera aperture.
Following Vehicle	The vehicle behind the subject vehicle having a defined headway.
Foreground	An image frame which changes with time.
Front Vehicle	The vehicle in front of the subject vehicle.
Ghost	A transparent vehicle in the image.
Head-On Crash	Head-on crash occurs, when a vehicle crosses the centerline of an undivided road either intentionally or unintentionally and collides with an opposing vehicle making an impact angle of zero.
Illumination Variation	Change in intensity of the video images due to diurnal variation.
Image Intensity	The value of pixel in a image.
Lead Vehicle	The front vehicle having a defined headway with the subject vehicle.

Monochromatic Image	An image having its color distribution from black (0) to white (255).
MPEG4	It is a method of describing compression of audio and visual digital data.
Multimodal	A probability distribution having more than single peak.
Opposing Vehicle	The vehicle coming from the opposite lane.
Overtaking	Catching up with and pass while travelling in the same direction.
Pictorial Speed	Speed of an object in an image.
Pixel	Smallest unit of an image.
Push Button	A rectangular sized object used in software development to invoke in built handles.
Residue Of The Lost	A term used to indicate the pixel values of foreground in the background after leaving the vehicle.
Resolution	The total area of the image.
Run-Time	A term used to indicate ‘during the run of a software’.
Shadow	A shadow is a lower intensity area in the background where light from a light source is blocked by an opaque object.
Thresholds	A cut-off value.
Time-To-Collision	The unsafe time difference between the subject and the front vehicles that could lead to traffic crashes, given that the vehicles keep their current speed without any appropriate evasive maneuvers.
Unimodal	A probability distribution having single peak.

Chapter 1

Introduction

1.1. Background of the study

Traffic safety aims at the reduction of fatalities causing from crashes among road users (pedestrians, cyclists, motorists). These aims of traffic safety cannot be achieved without proper understanding of the crash mechanism. Traffic safety analysis requires historical records of crash data. However, the crash data are associated with well-recognized problems due to lack in availability and quality. Furthermore, to prevent crashes by using historical records is a reactive approach requiring large amount of crash data. Hence, these problems related to crash data motivate the development of surrogate measures of safety. Surrogate safety measures are based on crash likelihood rather than on the observation of actual crashes. As such, crash probability can be used as a surrogate or complementary measure of road safety.

Every year, more than 3000 crashes occur in a small country like Bangladesh (56,977 square mile), which leads to 2700 casualties (1). Of these crashes, head-on crash account for about 15% of all reported crashes, while they are responsible for nearly 65% of fatal crashes (1). Head-on crash occurs, when a vehicle crosses the centerline of an undivided road either intentionally or unintentionally and collides with an opposing vehicle making an impact angle of zero (2). According to the studies (3, 4), it is the most severe type of crashes, which is of great concern to road safety authorities. For example, Wegman (5) reported that head-on crashes are responsible for nearly 25% of fatal crashes occurring on rural roads in Organisation for Economic Co-operation and Development (OECD) member countries. According to U.S. statistics on traffic accident fatalities for the year 2014, head-on crashes comprised only 0.1% of total crashes. However, they accounted for 5.1% of fatal crashes (6). As such, efforts are required to reduce the frequency and severity of this crash type. These involve the identification of factors associated with the crash occurrence. The factors can be identified by developing crash prediction models, which determine the relation between crash outcomes and a set of contributing factors. Moreover, the number of crashes can be minimized exploiting the outcomes of these models. Eventually, these outcomes may help transportation engineers and road safety authorities to ensure traffic safety.

The probability of head-on crash is dependent on time-to-collision (TTC), one of the most important safety indicator factors. In general, time-to-collision (TTC) between the subject vehicle and its front vehicle is used in estimating the crash probability as well as a time-based surrogate safety measure to identify the probability of head-on crash. In case of head-on crash, opposing vehicle represents front vehicle. In details, TTC is defined as the unsafe time difference between the subject and the front vehicles that could lead to traffic crashes, given that the vehicles keep their current speed without any appropriate evasive maneuvers. The subject vehicle having smaller TTC, i.e. the value is close to '0', is expected to collide with the front vehicle. It means that the crash potential approach becomes closer to '1'. In contrast, a vehicle that has a larger TTC has a smaller crash potential. From this perspective, it can be stated that the estimation of TTC is directly dependent on the occurrence of the overtaking maneuver.

Overtaking in two-lane highway is a complex task having a significant effect on capacity, level of service and safety (7). It is also a mentally complicated task (8) that substantially affects the highway performance. For a long time, it is considered critical in traffic engineering. This involves chain of actions related to accelerating, braking and steering of the vehicle. At the same time, the driver may have a priori knowledge or concurrently have to observe, analyze and judge in order to take the actions. The lack of the ability to overtake may lead to formation of large queues, decrease in traffic capacity, and environmental impacts (9). Overtaking involves abrupt and short-term decision making that may cause an increased in the risk of crash. Conversely, the understanding of driver's overtaking behavior and their decision-making on two-lane highways can significantly contribute to safety analysis, level of service evaluations and develop traffic simulation models. However, only limited research studies have been conducted to develop overtaking models using field data that capture real world overtaking decision-making.

To determine the overtaking status, high resolution traffic data is required. The traffic data can be obtained using ITS- based data collection system. For example, vehicle trajectory extraction using image processing can be used to obtain more detailed traffic data from the field (10, 11). Using video sensors, several automated systems have been developed for traffic monitoring. For instance, Saunier and Sayed (11) propose a framework for estimating crash probability for two vehicles at an intersection by a vision-based vehicle tracking system. In another study, Oh

and Kim (12) develop a methodology for estimating rear-end crash potential using individual vehicle trajectory data extracted by vision-based software.

1.2. Statement of the problems and opportunities

1.2.1. Absence of high-resolution data collection technique

Within the vast literature on crash prediction model, surprisingly few studies have worked with head-on crash using field data. Such limited research is primarily attributed to the difficulty of high-resolution data collection. Here loop detectors are unsuitable due to measurement errors (29). Moreover, traffic cameras for vehicle detection are often absent along the corridor. But given that accurate high-resolution traffic data is the pre-requisite for developing a crash prediction model, this research attempts to establish data collection (i.e. traffic flow and vehicle trajectory) technique based on image processing which will be able to collect data with reasonable accuracy. Also the developed technique is expected to be robust and easy to use.

1.2.2. Absence of perfect background model for vehicle detection

Background subtraction has become a popular technique in vehicle detection. However, it suffers from various visual challenges. These visual challenges include illumination variation (gradual and sudden), shadow, camera jitter, dynamic background (i.e. river, rain or snow) camouflage etc. Unfortunately, very few researches have solved them individually and at the same time. Moreover, no such vision based system has been developed to monitor real-time traffic flow in non-lane-based traffic stream.

1.2.3. Inadequate Statistical Model Infrastructure

The existing overtaking models lack adequate variables representing drivers' characteristics (i.e. aggressiveness). Thus, the true nature of overtaking could not be captured accurately by overlooking such an important variable. Moreover, the models lack flexibility due to having a less number of parameters. It arises due to their old traditional regression structure with linear expression. As a result, the models do not have enough flexibility to fit into the data. Thus, this research attempts to develop a new mathematical model to achieve more goodness of fit into the data.

1.2.4. Existing Perilous Time-to-Collision Formulation

Literature study also reveals that no formulation considers the acceleration of the subject vehicle while calculating time-to-collision. This research endeavors to find the exact contribution of acceleration in time-to-collision by utilizing equations of motion. This deep investigation will add

new dimension to the definition of safety in two lane – two way rural highway. Overlooking this important parameter causes in erroneous time-to-collision with increases pseudo-factor of safety.

1.3. Research objectives and scope of work

This study is concerned with the development of a model that estimates head-on crash probability from classified vision based vehicle trajectory. The specific objectives are:

- a) Developing dynamic illumination adaptive background model and foreground segmentation formulation to detect heterogeneous vehicles from real-time video.
- b) Deriving Overtaking Decision (OD) model incorporating different traffic flow variables obtained from vision based trajectory.
- c) Establishing analytical formulation of time-to-collision (TTC) considering the dynamic acceleration of heterogeneous vehicles.
- d) Conjugation of OD model and TTC formulation within a new crash probability model to estimate the probability of crash of overtaking vehicles.

1.4. Organization of the thesis

Chapter 1 gives an introduction of the relevant research background, statement of problems as well as the objectives and scope of this research.

Chapter 2 comprehensive reviews previous works on overtaking model, time-to-collision and vehicle trajectory. The overtaking models are reviewed with respect to their categories in terms of level of detail and data collection methodology.

Chapter 3 shows the complete formulation a background model. A number of theories are proposed to define different components of an image. These definitions have been utilized to formulate the traffic detection problem and subsequently three adaptive dynamic background models have been developed to solve it.

Chapter 4 presents a wireless real-time traffic measuring system using surveillance camera for both lane-based and non-lane-based traffic streams. The system includes a new background estimation model for foreground segmentation using traditional background subtraction technique (BGS).

Chapter 5 presents the methodology of estimating crash probability. It includes the video capturing method, vehicle detection method, tracking method, vehicle classification method.

This chapter also describes the skeleton of the overtaking model and the formulation of time-to-collision. Ultimately, these two aspects are conjugated within the crash probability model.

Chapter 6 presents the step by step procedure of Bayesian analysis using Markov chain Monte Carlo (MCMC) methods in WinBUGS.

Chapter 7 includes a detailed description of data collection and data analysis methodology. Pictorial data for calibrating background model vehicular data for overtaking modeling is presented here. The novelty of the temporal overlapping matrix is explained in this chapter. The method of extracting all the required variables for calibrating the model is also revealed in this chapter.

Chapter 8 presents the calibration and validation of the background modeling and the overtaking model. Field testing of the models are also included in this chapter.

Chapter 9 includes a detailed analysis over the analyzed data. Initially the correlation among the data have been revealed. Afterwards, the data has been fitted into different models and the performance of the models have been compared. Ultimately, the model have been fitted into the newly proposed model in chapter 3.

Chapter 10 concludes the thesis with recommendations and future research direction.

Chapter 2

Literature Review

2.1. Introduction

This chapter reveals the literature review of five groups of study, which are: (1) overtaking model; (2) time to collision formulation; (3) background model; (4) vehicle detection; and (5) vehicle trajectory estimation. From methodological point of view, these groups of study are from different framework, which can be analytical, statistical, empirical framework or complex mixture of these frameworks. However, this thesis work agglomerates this complex mixture into common single framework to estimate the crash probability. Thus, elaborate literature study on the components of this complex mixture is necessary for this novel thesis work. Accordingly, the studies related to the different framework group are discussed in the subsequent sections.

2.2. Overtaking Models

Overtaking maneuvers are complex tasks that require the driver to process multiple sources of information and make decisions in short time periods. In case of two-lane highways, analytical approaches to model overtaking behavior are found to be dominating in literatures (7, 13). The approaches focus on modeling the sight distance for overtaking model in two-lane highways. Manual data collection from video recordings has been popular as well as manual data collection from observations and moving observer methods (14). Additionally, researches on overtaking in two-lane highways include modeling of the speeds of both the vehicle overtaking and the lead vehicle (15) or the speed difference between the vehicle overtaking and the lead vehicle (16), as well as the gap acceptance of impatient drivers (17). However, recent studies consider the use of alternative approaches to analyze two-lane roads, such as driving simulators, connected vehicle environment and microscopic simulation.

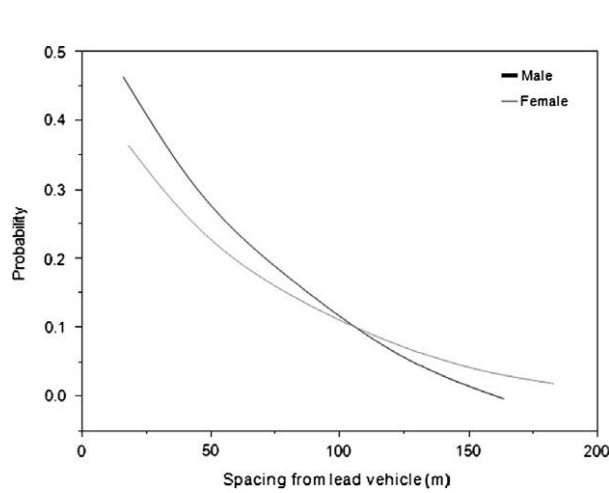
2.2.1. Driving Simulators

Exploiting driving simulators, single passing maneuvers or binary choice processes are modeled in several studies (18, 20). Nonetheless, these models use simulation data rather than field data, where all the scenarios are predesigned. Vlahogianni and Golias (21) use Bayesian Networks (BN) to model the uncertainty hindering in the overtaking behavior of young drivers in two-lane highways. They also reveal the microscopic traffic characteristics influencing the

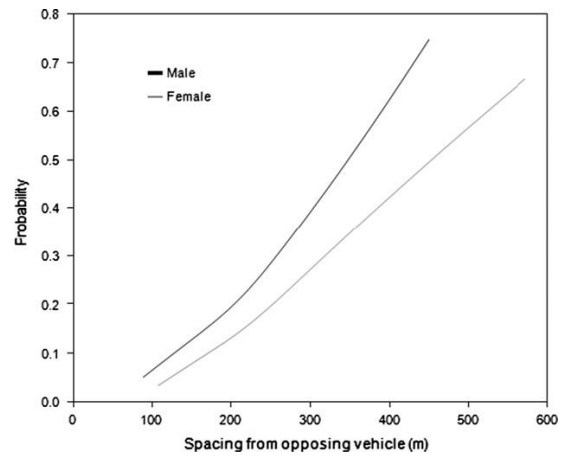
driver's decision making process. A BN defines a unique joint probability distribution over X given by:

$$P_B(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P_B(X_i | \Pi_{X_i}) \quad (2.1)$$

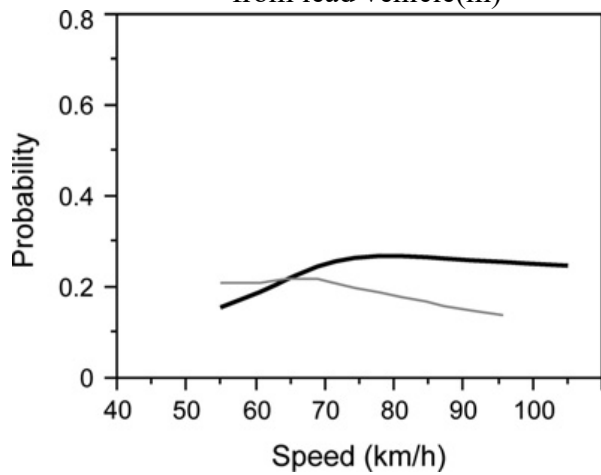
They have estimated the probability of overtaking both for male and female. Some of the results are shown in figure 2.1.



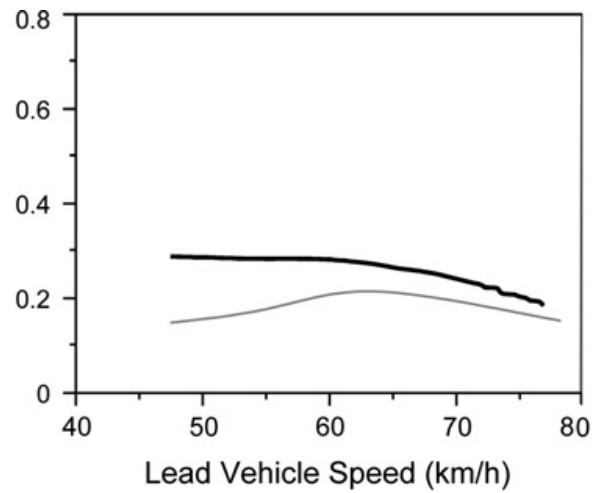
(a) Probability of overtaking vs. Spacing from lead vehicle(m)



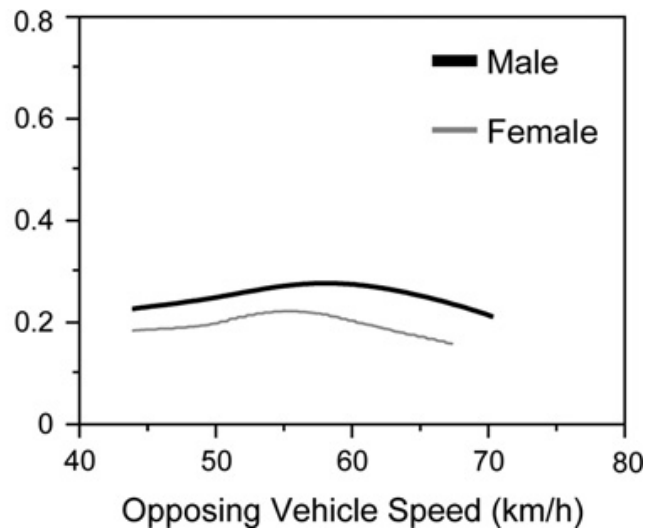
(b) Probability of overtaking vs. Spacing from opposing vehicle



(c) Probability of overtaking vs. Speed of the subject vehicle



(d) Probability of overtaking vs. Speed of the lead vehicle



(e) Probability of overtaking vs. Speed of the opposing vehicle

Figure 2.1 Overtaking taking probability of male and female in different context.



Figure 2.2 A glimpse of driving simulator.

2.2.2. Connected Vehicle Environment

Motro et al. (22) assess the effectiveness of a dedicated short-range (DSRC) communication system, called the overtaking assistant, devised for detecting unsafe overtaking maneuvers on two-lane rural highways. Nevertheless, sensor error and estimation inaccuracies are found to increase the rate of false warnings. Figure 2.3 shows the phases of DSRC system.

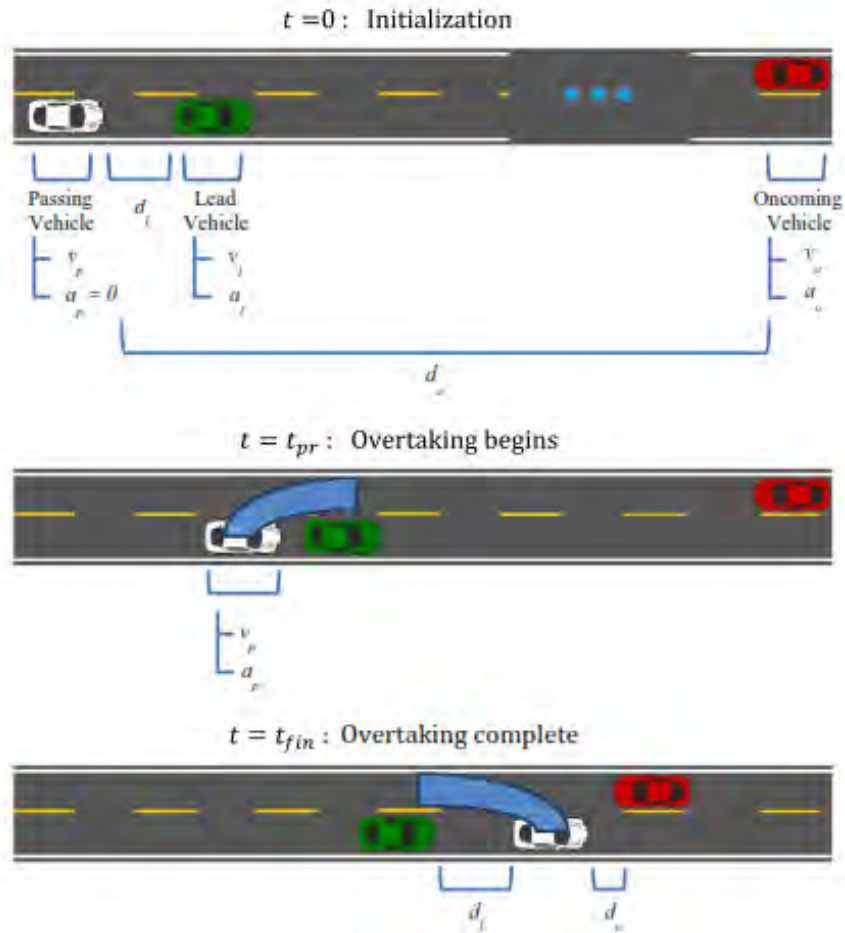


Figure 2.3 Phases of an overtaking maneuver in DSRC system.

2.2.3. Microscopic Simulation

Tapani (23) develops a Rural Traffic Simulator (RuTSim) with simulation models specific to rural road environments. The model has been used by Hegeman et al. (24) to evaluate an overtaking assistant showing significant increase of safety in overtaking maneuvers. Several other research studies also develop their own customized micro-simulators to explore different approaches to model overtaking behavior (25, 26). All of the above simulators use ADAS (Advanced Driver Assistance Systems) which attempts to proactively anticipate and prevent crashes. However, it does not consider potential uncertainties (or errors) in the information obtained for predicting crashes. In fact, most studies mentioned above do not even discuss whether the information is obtained through sensors, V2V communications, or other means. Llorca et al. (19) develop a micro-simulation model for two-lane rural highways, which analyzes the effect of human factor, road geometry, and vehicle characteristics on passing maneuvers. Nonetheless, the effect of the variables cannot be explained by this model.

2.3. Time-to-Collision

Time-to-collision (TTC) is considered as one of the most widely used safety indicators. It can be applied to different types of conflicts such as rear-end, head-on and right-angle crashes (36). The concept of TTC was introduced by Hayward (37) and it is defined as the time required for two vehicles to collide if they continue at their present speed and on the same path. According to Svensson (38), TTC is an indicator for a traffic conflict and is, thus, inversely related to accident risk (smaller TTC values indicate higher accident risks and vice-versa). This TTC information is an important cue for the driver in detecting potentially dangerous situations. The study by Farah et al. (39), TTC is chosen as a measure of risk, which indicates the remaining gap to the oncoming vehicle at the end of the passing maneuver. This measure is available for accepted passing gaps. However, it is not available for rejected passing gaps. In (40), the authors present an improved definition for calculating TTC in situations in which two vehicles interact in the same lane for a period of time (conflict lines), including lane-changing, merging, and rear-end conflicts. However, existing literatures do not give any modification of TTC for overtaking maneuvers.

2.4. Background Modeling

Previously, conventional basic models such as mean (42), median (43) and histogram (44) were used for background modeling. These models suffered biasness of central tendency of an aggregated data. Afterwards, such background models were forced into several parametric distributions, such as Gaussian (45-46), to overcome the limitation of the basic models. Unfortunately, these unimodal models could not handle dynamic backgrounds. Accordingly, Gaussian Mixture Model (GMM) (47) was introduced to model background. However, background having fast variations cannot be accurately modeled with a few Gaussians. Advanced statistical models removed the flaws of the old statistical models by making themselves robust. Student-t Mixture Model (STMM) (48) and Dirichlet Mixture Model (DMM) (49), which use distribution other than Gaussian are proven to be robust in dynamic background than GMM. It is because of their more heavily-tailed nature. However, real-time implementation of both STMM and DMM are difficult due to massive computational complexity.

Non-parametric model (50) was also introduced to estimate per pixel background probabilities from many recent samples over time, using Kernel density estimation (KDE). For approximation of the background color distribution, Ding et al. (51) used a mixture of KDE

and GMM. Barnich et al. (52) proposed Visual Background Extractor (ViBe), a sample-based algorithm which uses random selection policy that ensures a smooth exponentially decaying lifespan. However, it faced problem with challenging scenarios such as darker background, shadows, and frequent background changes. Hofmann et al. (53) proposed Pixel-Based Adaptive Segmenter (PBAS) which models the background by a history of recently observed pixel values.

Nevertheless, more sophisticated statistical models were explored to seek improvement such as support vector machine (SVM) (54), support vector regression (SVR) (55) and support vector data description (SVDD) (56). In these models, the background estimation is not based on the probability function of the background like parametric and non-parametric techniques. Thus, the memory requirements are less. However, a major problem with these data-driven statistical models is the limited number of examples while training rare or anomalous behaviors. To overcome this limitation, subspace learning using Principal Component Analysis (SL-PCA) (57) was applied on a number of images to construct a background model. However, Bouwmans (58) found two limitations in SL-PCA: (1) foreground objects must be smaller in size and they should not be in static state for a long period in the training sequence; and (2) batch mode PCA is computationally intensive for the background maintenance. To overcome the limitations of traditional subspace learning model, several discriminative and mixed subspace learning models were proposed. Attention was caught only by reconstructive subspace learning models such as PCA (Principal Component Analysis) and NMF (Non-negative Matrix Factorization). Farcas et al. (59) introduced discriminative and supervised approach which was based on an incremental discriminative subspace learning algorithm named as Incremental Maximum Margin Criterion (IMMC). For attaining more robustness, Marghes et al. (60) used a combination of a reconstructive method (PCA) with a Linear Discriminant Analysis (LDA) (61).

Besides SL, filter based approaches were adopted to dominate over statistical methods. Wallflower, a pixel-level algorithm, was proposed by Toyama et al. (62) which makes probabilistic predictions about the background pixel values using a single step Wiener prediction filter. It works well for periodically changing pixels; however, disadvantage occurs when a moving object corrupts the history values. Karnna et al. (63) proposed Kalman filter based background estimation which is an optimal estimator of the state of processes. Although it gives the optimal solution to the estimation problem when all the processes are Gaussian, it offers a sub-optimal behavior in non-Gaussian arising challenging situations.

Cluster based algorithms were adopted to reduce computational complexity. Cluster models assume that clusters are temporal representation of each pixel in the frame. Butler et al. (64) proposed an algorithm that assigns a group of clusters to each pixel in the frame using K-means algorithm. To amplify the robustness, Xiuman et al. (65) used genetic K-means algorithm which alleviates the disadvantages of the traditional K-means algorithm. Particularly, traditional K-means algorithm has random and locality aspects causing lack of global optimization. Alternatively, Kim et al. (66) proposed to estimate the background using a codebook—a type of cluster model. Depending on color distortion metric and brightness, samples at each pixel are clustered into a set of codewords.

Neural network (NN) was also applied to represent a background which is suitably trained on a number of clean frames. It learns how to classify each pixel as background or foreground. For background modeling and foreground detection, Culibrk et al. (67) used NN architecture to form an unsupervised Bayesian classifier. In another study, Luque et al. [68] used a method based on multi-valued discrete NN to detect foreground. Moreover, Palomo et al. (69) proposed a growing hierarchical NN which has a structure divided into layers and each layer is composed of different single Self Organizing Neural Networks (SONN) with adaptive structures that are determined during the unsupervised learning process. Furthermore, fuzzy concepts were introduced in the different stages of background subtraction. Kim et al. (70) proposed a fuzzy c-means clustering model. It allows reducing color variations generated by background motions and highlights foreground objects which produces better dynamic backgrounds.

Some authors proposed to isolate the background and the foreground in a different domain. Wren et al. (45) estimated the background model for capturing spectral signatures of multi-modal backgrounds by using Fast Fourier Transform (FFT) and inconsistent signatures were used to detect changes. Using Walsh Transform (WT), Tezuka et al. (71) modeled the background incorporating the GMM which is applied on multiple block sizes. Gao et al. (72) adopted Marrwavelet kernel and used binary discrete wavelet transforms to achieve foreground detection. Guan (73) used Dyadic Wavelet (DW) to detect foreground objects where the difference between the background and the current images is decomposed into multi-scale wavelet components.

2.5. Vehicle Detection

Background subtraction is the most widely used technique for vehicle detection, where an estimated background is subtracted from the current frame to give a differential image. In this

process, a constant or dynamic threshold value is applied on the resulting differential image to give the foreground, which contains the objects of interest. Advancements have been made in adapting background image to illumination variation and addressing the effect of shadows (80, 81). Frame differencing is another method which is closely related to background subtraction. However, it becomes challenging to use frame differencing method when it cannot cope with noise, sudden illumination variation or periodic movement of background as trees. In this regard, special care has been taken to suppress the influence of noise in the algorithm as presented in (82). The W4 model (83) uses three values to represent each pixel in the background image: the minimum and maximum intensity values, and the maximum intensity difference between consecutive images of the training sequence. A small improvement to the W4 model along with the incorporation of a technique for shadow detection and removal is presented by Jacques et al. (84).

For background modeling, a popular technique is the background averaging method. In this method, all video frames are summed up for averaging. The learning rate specifies the weight between a new frame and the background. The use of averaging, usually for computational reasons, is reported in (85). In (86), detection accuracy of 90% has been achieved through this method. However, in this method, contamination of background occurs due to moving objects resulting into tails behind them. These tails generated by moving vehicle has been reduced in (87) by averaging the instantaneous estimated background instead of the current frame, although this leads to erroneous background estimation for poor threshold value. In (88), mode of frames has been taken to estimate the background image. However, robustness towards light variation and long term operation have not been demonstrated. Median filtering is another common approach of background modeling (89, 90). In this method, the estimated background is considered to be the median at each pixel location of all the frames in the collection. It is assumed the pixel stays in the background for more than half of the frames in the collection. In (89), median filtering has been extended for different colors by replacing median with medoid. A simple recursive filter to estimate the median has been proposed in (91). In (92), the median filtering technique has been applied for background estimation in the context of urban traffic monitoring. The running estimated median is incremented by one if the input pixel is larger than the estimate, and decreased by one if smaller.

A Single Gaussian background model has been used in (93) and (94) to improve robustness compared to the averaging method. This method produces mean and variance images for modeling background. A new pixel is classified, depending on the position in the Gaussian distribution, which is the statistical equivalent to a dynamic threshold. To cope up with

multimodal background, a probabilistic method is proposed in (95) for background subtraction, which models the values of a particular pixel as a mixture of Gaussians, assuming Gaussians correspond to background colors. The pixel value that excludes from these Gaussians is considered as foreground. This system has robustness over illumination changes, repetitive motions of scene elements, tracking through cluttered regions and slow- moving objects (95). To analyze the features of the detected objects in the foreground, different types of algorithms are available. Among them, connected component labeling collects and merges foreground pixels into objects by judging their eight-connectivity relationship with the adjacent neighboring pixels. It searches for and labels possible candidates by dividing foreground pixels into groups. In (95), Gaussian mixture model has been used to extract the foreground pixels which are ultimately segmented into regions by a two-pass connected components labeling. Since this procedure is effective in determining the whole moving object, moving regions can be characterized not only by their position, but also size, moments, and other shape information. These characteristics are useful for later processing and classification, and can aid in the tracking process. In (96), a two stepped refining method is introduced to enhance the performance of labeling algorithm. Edge detection, another type of methodology (97, 98), which is not capable of detecting low intensity objects; opens a prospect of error under low illumination conditions.

Shadow related approaches such as, Zvi et al. (99) combines two different methods to detect and remove shadows from RGB images of traffic videos. They used the method introduced in (100) to remove the shadows by zeroing the shadows' borders in an edge representation of the image, and then re-integrating the edge using the method introduced in (101). In (102), an entropy minimization technique is introduced for removing shadow. However, the results obtained are not satisfactory. A region-based approach to detect and remove the shadows from an image is proposed in (103). Germain et al. (104) tried to remove shadows by illuminating the shadow region until it has the same illumination as the surroundings and then reinstating the texture. Color and near infrared images are used for shadow detection and removal. Using shadow invariant, shadow variant and near-black features, a method is proposed in (105) to remove shadows from monochromatic image. However, this method enjoyed little success in removing the shadows. Removal of dark cars (cars having similar illumination as shadows) is an important consideration while addressing the shadow problem.

Some commercially used tools for vehicle detection include: AUTOSCOPE, CCATS, TAS, IMPACTS and TraffiCam (106, 107, 108, 33 and 109). The systems typically allow a user to specify several detection regions in the video image. These systems look for image intensity

changes in the detection regions to indicate vehicle presence/passage. AUTOSCOPE (111) detects vehicles correctly and extracts traffic parameters (such as traffic count, vehicle speed, vehicle length, distance between successive vehicles and road occupancy rate) if the traffic flow is not overloaded severely. However, detection errors generate due to shadow effects. Congested traffic conditions leads to erroneous background estimation as AUTOSCOPE only updates the background when vehicles are not present. Furthermore, stopping vehicle causes occultation into the background (111). The commercial tool Citilog is used for traffic data collection (speed, volume, occupancy, level of service) for planning and real-time road user information by using existing CCTV cameras. Another commercial tool – TrafiOne is an all-round detection sensor for traffic monitoring and dynamic traffic signal control which uses thermal imaging and Wi-Fi tracking technology to provide traffic engineers with high-resolution data on vehicles, bicycles and pedestrians at intersections and in urban environments. However, evaluations of commercial VIPS (Video Image Processing Systems) found the systems had problems with congestion, high flow, occlusion, camera vibration due to wind, lighting transitions between night/day and day/ night, and long shadows linking vehicles together (106, 108,109 and 110). The need for traffic surveillance under all conditions has led to research in more advanced video-based vehicle detection. For instance, Chao et al. (112) have developed an algorithm to differentiate vehicles from shadows.

2.6. Vehicle Trajectory

Vehicle trajectory can be obtained by vehicle tracking from video sequences, which involves accurate foreground detection. For foreground detection, different types of segmentation techniques such as Gaussian Mixture Model (GMM), Visual Background Extractor (ViBe), Pixel-Based Adaptive Segmenter (PBAS) are used in recent studies (27, 28). In (29), the authors develop a new dynamic background estimation model for foreground segmentation which addresses both of the following environmental challenges, i.e. illumination variation (sudden and gradual) and shadow along with tail backs from vehicles. To analyze the features of the detected objects in the foreground, algorithms such as connected component labeling and edge detection method are applied in different research studies (30, 31). After the successive vehicle detection, the following tracking algorithms are commonly used: region-based, snake-based, feature-based, model-based or multi-thread (32). Another group of algorithms deals with object tracking using kalman filter (33). In (34), the authors use Kalman filter to establish object motion model, using the current object's geometric (e.g. centroid)

information to predict its position. As such, it can reduce the search scope and time of moving object to achieve fast tracking. The proposed method is able to ensure an efficient and robust tracking with merge and split of multi-object. The coordinate from tracking can be interpreted as object's trajectory. There have been varied approaches to handle the trajectory of moving objects analysis based on video and some solutions have already been proposed. For example, in (35), trajectory analysis is conducted in order to detect the following traffic events: illegal changing lane, stopping, retro-gradation, sudden speeding up or slowing down.

2.7. Summary

Attempts have been made to review the recent studies regarding the overtaking models, time-to-collision, background modeling, vehicle detection and vehicle trajectory estimation. It is evident that most of the studies use simulated data for modeling the overtaking phenomena rather than real-time field data. Conversely, the traditional formulation of TTC considers the spacing and the speed differential between the vehicles only. It does not consider dynamic accretion of vehicles, which leads to over estimation of TTC. It is also evident that vehicle detection is the most convenient procedure for traffic measurement. However, it suffers from various visual challenges and very few researches have solved them individually and at the same time. Moreover, no such system has been developed to monitor real-time traffic flow in non-lane-based traffic stream. From the literature survey, it has been revealed that none of background models give any explicit solution to illumination variation while traffic detection. Moreover, now-a-days various State-of-Art tracking methods are available. A tracking method based on Kalman filter was proposed method, which is able to ensure an efficient and robust tracking with merge and split of multi-object.

Chapter 3

Background Modeling

3.1. Introduction

Vision based systems have become very popular in Intelligent Transport System (ITS) to measure the real-time macroscopic and microscopic parameters of the traffic stream through object identification and tracking. Conventional technology for traffic measurements including inductive loops, sonar and microwave detectors suffer from drawbacks for being expensive to install, causing traffic disruption during installation or maintenance, not being portable and unable to detect slow or stationary vehicles. On the contrary, vision based systems are easy to install, maintain, can be integrated as a portion of signal control, and has the potential to utilize the extant traffic surveillance infrastructure. Moreover, these vision based systems can be easily upgraded and offer flexibility to reform the system and functionality by simply altering the system algorithms. Currently, automated traffic state measurement employing vision system is a key technology in the management of the transportation facilities. Vision-based systems uses several detection techniques. Among them, background subtraction is the most widely used technique for detecting traffic, where an estimated background is subtracted from the current frame to give a differential image. In this process, the accuracy of traffic detection largely depends on the fidelity of background estimation. Thus, the estimation model should be robust to various challenges, particularly illumination variation. It causes increase or decrease in the intensity of pixels, resulting in false positive or overestimation. Although numerous models were proposed to capture the background dynamics, none of those specifically concentrated on the issue of illumination variation in traffic detection. This chapter endeavors to develop a background model which can address illumination variation.

3.2. Related Definitions

3.2.1. Understanding Frame, I_n

Let, \mathcal{V} be the collection of frames I_n where $n = \{1, \dots, N\}$, N being the total number of frames. Frame I_n is a space containing pixels P of the n^{th} frame where $P = f(\lambda)$. λ is the space variable which can be defined as $\lambda = \{(i, j, c) \mid i, j, c \in (\omega \times \eta \times \pi)\}$, where $\omega = \{1, 2, \dots, W\}$, $\eta = \{1, \dots, H\}$ and $\pi = \{1, 2, \dots, \Pi\}$. W and H are the total number of pixels along width and height of the surface

I_n , respectively. Γ is the number of color channels within the space of I_n . Further definition of these parameters are given as follows,

$$\left. \begin{array}{l} \forall N \\ \forall W \\ \forall H \\ \forall \Pi \end{array} \right\} \in \mathbb{Z}^+ \text{ and } \forall P \in \mathbb{Z}_u^+$$

Where, $\mathbb{Z}^+ = \{x \mid \forall x \in \mathbb{Z}, x > 0\}$ and $\mathbb{Z}_u^+ = \{x \mid \forall x \in \mathbb{Z}, 0 \leq x \leq 255\}$.

Using these definitions, I_n can be defined as $I_n = \{P(\lambda) \mid \forall \lambda \in \langle \omega \times \eta \times \pi \rangle\}$, which contains all the pixels _{P} within the space. For simplicity, Γ is assumed to be 1, which converts the frame I_n into a planar surface from a space. It indicates monochromatic frame containing pixels having color distribution from black to white; whereas, black denotes the weakest and white denotes the strongest color intensity. From this assumption, I_n can be redefined as $I_n = \{P(\lambda) \mid \forall \lambda \in \langle \omega \times \eta \rangle\}$. The cardinality of I_n turns into $\omega\eta$ representing the area of the surface.

3.2.2. Defining Background (B) and Foreground (F)

In a particular frame, there are two types of pixels: one is the background pixel (B), which does not include the object of interest and the other is the foreground pixel, which includes the object of interest. In case of traffic detection, vehicle/pedestrian is object of interest and the rests are considered background. The object of interest is also known as foreground (F). Furthermore, the form of occurrence of background and foreground pixel on the surface I_n is totally different. The physical characteristics of the background and foreground can be expressed using the following inequality,

$$\textit{Theorem 1.} \quad \frac{\partial^2 I_n}{\partial n \partial \lambda} \Big|_{\lambda \in \lambda_B} \neq \frac{\partial^2 I_n}{\partial n \partial \lambda} \Big|_{\lambda \in \lambda_F}, \forall (\lambda_B, \lambda_F) \in \lambda, |\lambda_B| + |\lambda_F| = |\lambda|$$

Proof. Let, I_{n_1} and I_{n_2} are two frames at time n_1 and n_2 which do not contain any object of interest, where $n_1 \in n, n_2 \in n$. Thus, they are two ideal backgrounds. If the pattern of background pixels remains temporally similar, same change will be experienced by all the background pixels. Let the difference between these backgrounds to be Δ , where $\Delta \in \mathbb{R}$.

Hence, considering λ_B be the set denoting position of background pixels and $\lambda \in \lambda_B$,

$$\left(\frac{dI}{d\lambda} \right)_{n=n_1} = \lim_{(\lambda_2 - \lambda_1) \rightarrow 0} \frac{I_{\lambda_2} - I_{\lambda_1}}{\lambda_2 - \lambda_1}$$

$$\left(\frac{dI}{d\lambda} \right)_{n=n_2} = \lim_{(\lambda_2 - \lambda_1) \rightarrow 0} \frac{I'_{\lambda_2} - I'_{\lambda_1}}{\lambda_2 - \lambda_1} = \lim_{(\lambda_2 - \lambda_1) \rightarrow 0} \frac{(I_{\lambda_2} + \Delta) - (I_{\lambda_1} + \Delta)}{\lambda_2 - \lambda_1} = \lim_{(\lambda_2 - \lambda_1) \rightarrow 0} \frac{I_{\lambda_2} - I_{\lambda_1}}{\lambda_2 - \lambda_1} = \left(\frac{dI}{d\lambda} \right)_{n=n_1}$$

Therefore,
$$\frac{\partial^2 I_n}{\partial n \partial \lambda} = \lim_{\substack{n_2 - n_1 \rightarrow 0 \\ \lambda \in \lambda_B}} \frac{\left(\frac{dI}{d\lambda}\right)_{n=n_2} - \left(\frac{dI}{d\lambda}\right)_{n=n_1}}{n_2 - n_1} = 0$$

Again, let, I_{n_1} and I_{n_2} are two frames at time n_1 and n_2 which contain the object of interest, where $n_1 \in n, n_2 \in n$. Thus, they are two ideal foregrounds. As the foreground is temporally dynamic, the pixels which are left by foreground faces change and other remains same. Thus, any change in foreground will result in different amount of changes in the foreground pixels.

Hence, considering λ_F be the set denoting position of foreground pixels and $\lambda \in \lambda_F$,

$$\frac{\partial^2 I_n}{\partial n \partial \lambda} = \lim_{\substack{n_2 - n_1 \rightarrow 0 \\ \lambda \in \lambda_F}} \frac{\left(\frac{dI}{d\lambda}\right)_{n=n_2} - \left(\frac{dI}{d\lambda}\right)_{n=n_1}}{n_2 - n_1} \neq 0.$$

Remark. The inequality in *Theorem 1* implies that the rate of change in foreground is different than that of background. Thus, this inequality warrants distinguishing background pixels set from the foreground.

Theorem 2. No two pixels of different object can co-exist at same λ in I_n .

Proof. Let, any two objects A and B occupy two points λ_1 and λ_2 respectively in I_n . Thus, $\lambda_1 \in A$ and $\lambda_2 \in B$. However, $\lambda_1 = (\omega_1, \eta_1)$ and $\lambda_2 = (\omega_2, \eta_2)$ where ω and η are two orthogonal index vectors which does not contain any repetitive value. Thus, $\omega_1 \cap \omega_2 = \phi$ and $\eta_1 \cap \eta_2 = \phi$. Therefore, λ_1 and λ_2 must be disjoint sets and $\lambda_1 \cap \lambda_2$ must be equal to ϕ . Hence A and B cannot occupy same λ in I_n .

Remark. From *Theorem 2*, it is clear that no two type pixels can co-exist at same place in a same frame. Therefore, background and foreground pixels are apart spatially on the surface I_n .

From this particular *Theorem 2*, the foreground and background pixels can be defined precisely as in Equation (3.1),

$$\begin{aligned} F &= F(n) = F_n = \left\{ \forall \lambda_F \in \langle \omega \times \eta \rangle \mid P(\lambda_F), \forall \lambda_F \mid P(\lambda_F) \in I \right\} \\ B &= B(n) = B_n = \left\{ \forall \lambda_B \in \langle \omega \times \eta \rangle \mid P(\lambda_B), \forall \lambda_B \mid P(\lambda_B) \in I \right\} \end{aligned} \quad (3.1)$$

Where,

$$\bigcup_{B, F \in I} (\lambda_B, \lambda_F) = \lambda,$$

$$\bigcap_{B, F \in I} (\lambda_B, \lambda_F) = \emptyset$$

$$\bigcup_{\lambda_B, \lambda_F \in \lambda} (B_n, F_n) = I_n$$

$$\bigcap_{\lambda_B, \lambda_F \in \lambda} (B_n, F_n) = \emptyset$$

From the above conditions, it is clear that B_n and F_n do not share any common position. Thus, they are disjoint sets.

Theorem 3. $I_n = B_n + F_n$

Proof. From *Theorem 2*, it is evident that two pixels of different objects cannot occupy same position in I_n . Thus, foreground pixels are different than background pixels. Hence, $\{\lambda_B\} \cap \{\lambda_F\} = \phi$ and the space variable can be written as a union of background and foreground pixel positions. Therefore, $\lambda = \{\lambda_B, \lambda_F\}$, where $|I_n| = |\lambda_B|_n + |\lambda_F|_n$. In this case, If I_n contains WH pixels, then the background and foreground pixels are expressed as, $B_n = I_n \alpha$ and $F_n = I_n \beta$ respectively, where,

$$\sum_{i \in \omega, j \in \eta}^{W,H} \alpha = |\lambda_B|, \alpha \in [0,1], \beta \in [0,1]$$

$$\sum_{i \in \omega, j \in \eta}^{W,H} \beta = |\lambda_F|, \alpha\beta = 0, \alpha + \beta = 1$$

Thus, $B_n + F_n = (\alpha + \beta)I_n = I_n$.

Remark. Using this theorem, any frame I_n can be written as addition of background and foreground as in Equation (3.2).

$$I_n = B_n + F_n \quad (3.2)$$

3.2.3 Understanding the background, B_n

As background is complex, it needs further definitions to express its auxiliary components, i.e. illumination variation, residue of the lost, shadow, and camera jitter. In this study, the first two are focused for illustration. A static background is that on which auxiliary components do not have any effect; whereas, the actual background changes with time, although their physical characteristics are same. In the subsequent sections, the time variables n and t will be used interchangeably, where $\rho : n \rightarrow t$. In this equation, ρ depends on the streaming characteristics of the frame receiver and sender. Hereby, the static background can be defined as follows,

Definition 1. Static background B^* is such a frame I which follows

$$\left. \begin{array}{l} B \in I \mid \left(\frac{dI}{dn} \neq 0 \right) \\ \left\{ \left(\frac{dI}{dx} = \frac{dB_n}{dx}, \frac{dI}{dy} = \frac{dB_n}{dy} \right) \mid F_n \right\} = \phi \\ \exists!(n = N_B) : B_n - B^* = 0 \\ \forall n : n \neq N_B, B^* - B_n \neq 0 \\ \sum_{x \in [1, \omega]} \left| \frac{dB_n}{dx} \right| + \sum_{y \in [1, \eta]} \left| \frac{dB_n}{dy} \right| \neq 0 \end{array} \right\} \forall I \in I_n$$

Thus, there exists a temporal difference between static background and actual background, which can be expressed as, $B^* = B(t \pm t_1)$; where $t_1 \in \Delta t : \Delta t = |t_{B^*} - t_{B_n}|$. Since the definition of the

static background is available by now, the actual background can be expressed in terms of a static background. However, several assumptions are needed to be made before defining the actual background.

Assumption 1. At any time t , the difference between the actual background B_t and the static background B^* is only the illumination variation.

Remarks. If illumination variation is expressed as K_t ; using *Assumption 1*, actual background at any time t can be expressed in Equation (3.3) combining the static background defined earlier,

$$B_t = B^* \pm K_t \quad (3.3)$$

The definition of the illumination variation can be given as a g th order polynomial varying with respect to time as follows,

$$K_t = \left\{ \forall \lambda \in \langle \omega \times \eta \rangle \mid P(\lambda) \right\}, \quad \forall t: \frac{dK_t}{dt} = \sum_{g \in \mathbb{Z}^+, g \geq 0}^L m_g t^g$$

$$\left\{ K_t \pm (B_t \vee B^*) \right\} \in P, \quad m \in \mathbb{N}$$

For simplification, another assumption is needed to be made.

Assumption 2. The gradient of illumination variation curve is assumed to be a straight line.

Thus, putting $L=1$ and integrating the gradient constraint of K_t Equation (3.4) can be obtained.

$$K_t = mt_\Delta + K_0 \quad (3.4)$$

Where, K_0 is the initial illumination variation at the reference point; the gradient of illumination

$$\text{variation } m: \frac{dK_t}{dt} \in \mathbb{R} \quad \text{and} \quad t_\Delta = t_{B_t} - t_{B^*}.$$

If the frames in consideration exist within the same time as the static background, no variation of illumination will occur. Thus, if the static background resides in reference point, there exists no illumination variation. Moving forward through time will cause changes in illumination with respect to the static one. Therefore, $\exists t: t \rightarrow 0, B_t \rightarrow B^*, K_0 \rightarrow 0$. Considering this, Equation (3.3) can be rewritten, where illumination variation is expressed in a linear form.

$$(t_\Delta = t_{B_t} - t_{B^*}): \left\{ \lim_{|t_{B_t} - t_{B^*}| \rightarrow 0} B_t \rightarrow B^* + t_\Delta \left(\frac{dK_t}{dt} \right)_t \right\} \leftrightarrow \left\{ \left(\frac{d\mathbb{B}}{dt} \right)_t \rightarrow \left(\frac{dK_t}{dt} \right)_t \right\} \leftrightarrow \left\{ I_t \rightarrow B^* + t_\Delta \left(\frac{d\mathbb{B}}{dt} \right)_t + F_t \right\} \quad (3.5)$$

Let, $t_\Delta = t_{B^*} - t_{B_t}$, the relationship between B^* and B_t can be reformed from Equation (3.5),

$$B_t = B^* \pm t_\Delta \left(\frac{dK_t}{dt} \right) = B^* \pm mt_\Delta \quad (3.6)$$

Thus, we obtain a definition of actual background with respect to the static background and the linear illumination variation, which are showed in Figure 3.1.

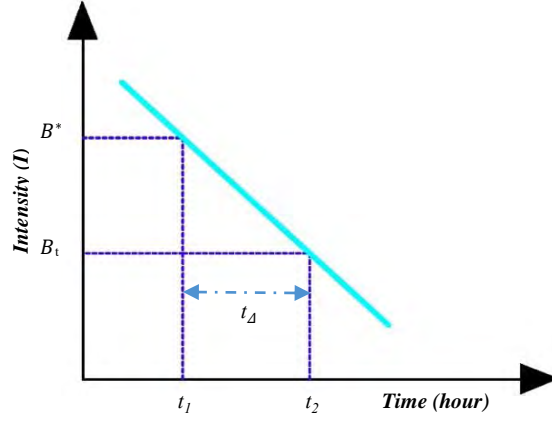


Figure 3.1 Expressing actual background B_t using static background B^* and linear illumination variation $\left(\frac{dK_t}{dt}\right)_t$.

3.2.4 Understanding the foreground, F_n

After defining the actual background, it appears that the foreground needs to be defined properly. Equation (3.1) only gives the symbolic definition of foreground. However, a vehicle should be considered as foreground during its detection. Note that, vehicles follow a particular pattern of occurrence as object on the frame. The properties of occurrence are stated below,

- (1) A vehicle occurs at any point on the surface I_n instantaneously and causes an impulse.
- (2) The occurrence interval depends upon the traffic state and control scheme.

These properties indicate two important phenomena: (1) the period of vehicle occurrence; and (2) the response of the vehicle on surface I_n . The first one depends on the vehicle occurrence pattern and the second one depends on the form of the response. For example, the occurrence pattern can be of different distribution, such as, vehicle follows Poisson distribution in comparatively free flow situation. On the other hand, the vehicles follow binomial distribution in congestion situation.

For simplicity, it is assumed that the vehicle may occur periodically with different distributions at different traffic state and it will form a square response wave as shown in Figure 3.2 with a period of Ω , which can be defined as,

$$\left. \begin{array}{l} \left(-\frac{t'+t''}{2} < t < -\frac{t'}{2}\right) \rightarrow 0 \\ \left(-\frac{t'}{2} < t < \frac{t'}{2}\right) \rightarrow l_t \\ \left(\frac{t'}{2} < t < \frac{t'+t''}{2}\right) \rightarrow 0 \end{array} \right\} F_t \in [0, l_t, 0], l_t \subseteq F_t$$

The premise of square response wavelets can be represented as a Fourier series.

$$F_t = \frac{l_t t'}{t'+t''} + \frac{l_t}{n\pi} \sum_{n=1}^{\infty} \left[\sin n\pi \left(\frac{t'}{2} + \frac{2t}{t'+t''} \right) + \sin \left(\frac{t'}{2} - \frac{2t}{t'+t''} \right) \right] \forall l \in P, t'+t'' = \Omega \quad (3.7)$$

And, the value of Ω for free flow can be obtained by solving the following equations

$$\text{Free flow: } \sum_{n=1}^{\infty} \frac{q\Omega \left[\frac{(q\Omega-1)}{(q\Omega+1)} \right]^{2n-1}}{(2n-1)} + \frac{1}{2} = 0$$

$$\text{Congestion: } \Omega - \frac{\sqrt{N_v}}{2q} (1 \pm \sqrt{N_v - 4}) = 0, N_v = \int_0^{n\Omega} q dt$$

Where, q = Average flow rate.

Combining Equation (3.4), (3.2) and (3.6), the l_t can be redefined with respect to a static background B^* , illumination variation K_t and vehicle F_t . Ultimately, the equation becomes,

$$I_t = B^* + t_{\Delta} \left(\frac{dK_t}{dt} \right)_t + \frac{l_t t'}{t'+t''} + \frac{l_t}{n\pi} \sum_{n=1}^{\infty} \left[\sin n\pi \left(\frac{t'}{2} + \frac{2t}{t'+t''} \right) + \sin \left(\frac{t'}{2} - \frac{2t}{t'+t''} \right) \right] \quad (3.8)$$

Where, t_{Δ} is the time difference between the l_t and B^* .

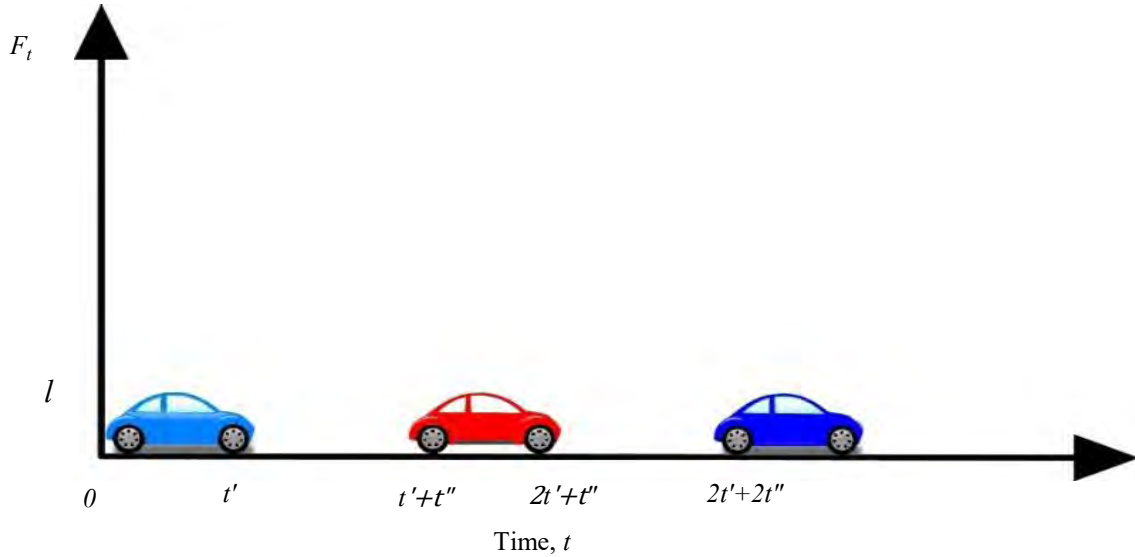


Figure 3.2 Representation of traffic F_t

3.3. Problem Formulation

Up to this point, both background and foreground have been defined properly. Thus, in this section, the traffic detection procedure has been introduced. This detection procedure has been explained by incorporating different backgrounds as defined in section 3.2.1. In each step of detection, limitations will arise and consequently the limitations will be overcome.

The most widely used technique for traffic detection is background subtraction [35]. In this regard, a background is subtracted from a frame to obtain a distance matrix. The distance matrix

is compared with a set of threshold value to convert it into binary image b_t , where, $b_t \in [0,1]$. The definition of distance matrix is given below.

Definition 2. The distance matrix can be obtained from the difference between the Frame I_t and any other frame Υ , where, $\Upsilon \in \mathcal{V}$

$$\delta_t = I_t - \Upsilon \quad (3.9)$$

Equation (3.2) can be combined with Equation (3.9) to form a relationship in between distance matrix and foreground.

$$F_t \rightarrow |\delta_t| \pm \tau_t : \Upsilon = \mathbb{B}$$

Where,

$$\mathbb{B} \in \bigcup_{i \in \mathcal{T}}^{T \rightarrow N} B_i, \quad \tau_t \in P : \tau_t \rightarrow \delta_t$$

If, $\mathbb{B} = B_t$, the distance matrix becomes the foreground itself. Thus,

$$F_t = |\delta_t| \quad (3.10)$$

As the distance matrix is defined by now, traffic can be extracted from the distance matrix.

Using Equation (3.5), (3.7) and (3.9), placing $\Upsilon = B^*$, the following relations can be obtained,

$$\delta_t = \pm \left\{ t_\Delta \left(\frac{dK_t}{dt} \right)_t + \frac{l_t t'}{t' + t''} + \frac{l_t}{n\pi} \sum_{n=1}^{\infty} \left[\sin n\pi \left(\frac{t'}{2} + \frac{2t}{t' + t''} \right) + \sin \left(\frac{t'}{2} - \frac{2t}{t' + t''} \right) \right] \right\} \quad (3.11)$$

Where, t_Δ is the time difference between δ_t and B^* .

While dealing with detection, it is not important that the difference is positive or negative unless any operation is done over the positive and negative images. Therefore, an absolute value of δ_t is used to extract the object.

$$\begin{aligned} |\delta_t| &= \left| t_\Delta \left(\frac{dK_t}{dt} \right)_t + \frac{l_t t'}{t' + t''} + \frac{l_t}{n\pi} \sum_{n=1}^{\infty} \left[\sin n\pi \left(\frac{t'}{2} + \frac{2t}{t' + t''} \right) + \sin \left(\frac{t'}{2} - \frac{2t}{t' + t''} \right) \right] \right| \\ &= \left| t_\Delta \left(\frac{dK_t}{dt} \right)_t \right| + \left| \frac{l_t t'}{t' + t''} + \frac{l_t}{n\pi} \sum_{n=1}^{\infty} \left[\sin n\pi \left(\frac{t'}{2} + \frac{2t}{t' + t''} \right) + \sin \left(\frac{t'}{2} - \frac{2t}{t' + t''} \right) \right] \right|, \quad [\because \lambda_{K_t} \cap \lambda_t = \emptyset] \end{aligned} \quad (3.12)$$

Now, during extracting the binary image b_t , this $|\delta_t|$ is compared with a set of threshold value τ_t . The mathematical structure shows that δ_t has two components: one is the vehicle l_t and the other is the illumination variation mt . For each time and for each pixel, both of these components will be compared with τ_t to decide whether the pixel is vehicle or not. Naturally, there is no need to adopt a threshold if the equality (3.10) is held true. Otherwise, the need of threshold values becomes predominant. In other words, the threshold values must be a function of K_t to suppress it, such that $\tau_t \rightarrow (K_t, l_t)$. The segmentation of vehicle can be done by,

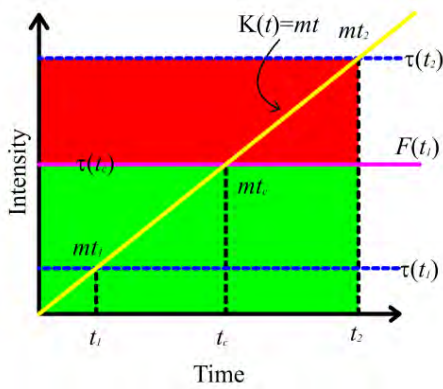
$$b_t = \begin{cases} 1, & \delta_t \geq \tau_t \\ 0, & \text{otherwise} \end{cases}, \quad \forall \lambda : (mt \geq \tau_t) \vee (l_t \geq \tau_t) \rightarrow 1 \quad (3.13)$$

From this equation, two such conditions can occur for a particular pixel:

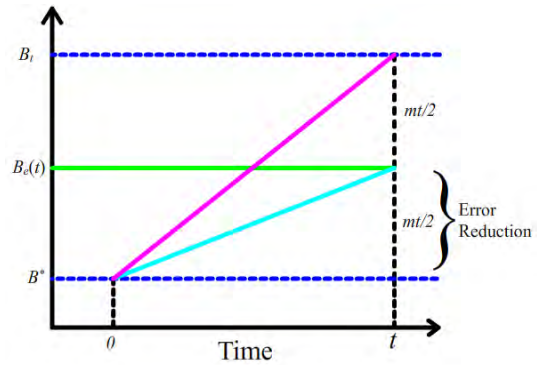
1. $(\exists \lambda: \tau_i < mt) \rightarrow b_i(\lambda) = 1$, which means that the threshold value is not enough to suppress the effect of illumination variation in b_i ; and

2. $(\exists \lambda: \tau_i > F_i) \rightarrow b_i(\lambda) = 0$, which means that the threshold value suppresses the vehicle.

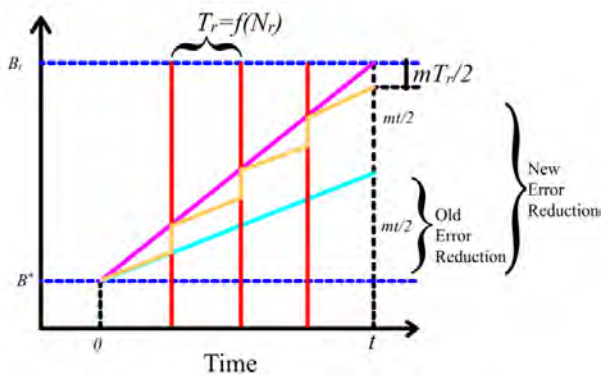
Figure 3.3(a) shows the aforementioned situations. If the value of the illumination variation increases with time, the threshold value also increases. At a certain time t_c , the threshold value is equal to vehicle pixel intensity, which suppresses vehicle as per Equation (3.12). After t_c , the condition 2 will occur. At this stage, as the threshold value becomes greater than the vehicle pixel intensity, hence the vehicle gets excluded from the binary image after being computed by Equation (3.13). However, even if the condition 1 prevails, the detection process generates an error $\left| t_{\Delta} \left(\frac{dK_t}{dt} \right)_t - \tau_t \right|$ in segmentation of F_i .



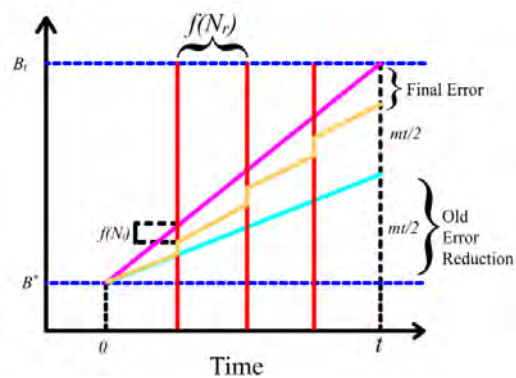
(a) Static background



(b) Estimated background using Model 1 (M1)



(c) Estimated background using Model 2 (M2)



(d) Estimated background using Model 3 (M3)

Figure 3.3 Traffic detection using different background models.

3.4. Solution Models

As described in Section 3.3, the main limitation of using the static background in traffic detection is the suppressing of vehicle pixel due to higher threshold value while accommodating larger illumination variation. In this section, three models will be presented which progressively overcome this limitation.

3.4.1. Model 1(M1)

Equation (3.5) can be rearranged after integration within time interval $[t_1, t_2]$ as follows.

$$t_{\Delta}(B_t - B^*) = \int_{t_1}^{t_2} I_t dt - B^* t_{\Delta} - \int_{t_1}^{t_2} F_t dt \quad (3.14)$$

The further derivation considers $t' \ll t_{\Delta}$, where $t_{\Delta} = t_2 - t_1$. The presence of a vehicle in the field of vision (FOV) is defined by a time interval of t' that is much smaller than t_{Δ} . This feature draws on a premise that the duration of any particular object in the FOV is miniscule. Thus, the residence time of the vehicle within the frame is much smaller with respect to whole video sequence. In another aspect, the vehicular properties (area) remain constant temporally; whereas, the centroid of the area varies within the FOV. If the centroid is the representative of all the pixels occupied by a vehicle, the intensity over the centroid can be expressed as $I_t = l$, where l does not vary with time.

$$\int_{t_1}^{t_2} F_t dt = M \int_0^{t'} F_t dt + M \int_{t'}^{t''} F_t dt = M \int_0^{t'} F_t dt$$

$$\int_0^{t'} F_t dt = \int_0^{t'} F_t dt = \int_0^{t'} \left\{ \frac{lt'}{t'+t''} + \frac{l}{n\pi} \sum_{n=1}^{\infty} \left[\sin n\pi \left(\frac{t'}{2} + \frac{2t}{t'+t''} \right) + \sin \left(\frac{t'}{2} - \frac{2t}{t'+t''} \right) \right] \right\} dt = lt' \theta(n)$$

If n is greater than 200, the term $\theta(n)$ converges towards 1 and the value of $\int_0^{t'} F_t dt$ becomes lt' . This portrays a new opportunity for Equation (3.14) to be rearranged as follows,

$$(B - B^*) = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} I_t dt - B^* - \frac{Mlt'}{t_{\Delta}}. \text{ However, the residence time of the object within the frame is}$$

much smaller with respect to whole video sequence ($t' \ll t_{\Delta}$) and $\frac{Mlt'}{t_{\Delta}} \approx 0$. Therefore, the

aforementioned equation becomes $B_c = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} I_t dt = I_{avg}$. The background can now be estimated

as follows,

$$B_e = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} I_t dt = I_{avg} \quad (3.15)$$

This background is estimated over certain number of frames and it is not representative of the actual background. Thus, it is termed as estimated background B_e . From this derivation, it can be stated that the estimated background is the average of the frames within the interval $[t_1, t_2]$. When I_t is a discrete function, I_t can be replaced with I_n . Now, Equation (3.15) can be written as follows,

$$B_e(n) = I_{avg} = \frac{1}{n} \sum_{i \in n} I_n \quad (3.16)$$

Using Equation (3.15), an estimated background at any time t can be obtained. However, it will not capture the illumination variation completely. Moreover, this operation produces a small trace of vehicle \bar{F} within it, which causes pollution. It can be realized from the equation below,

$$B_e(T) = \frac{1}{T} \sum_{t=1}^T I(T) = B^* \pm m \frac{T}{2} + \bar{F} \quad (3.17)$$

If it is assumed that the frames are free of vehicles and constituted illumination variation only, then the estimated illumination variation at any time t is $mt/2$; whereas, the actual value is mt (from Equation 3.6). It implies that the error is reduced by 50% with respect to static background. Yet, 50% needs to be recovered. This model also assumes that vehicle does not occur frequently. Thus, $\bar{F} \approx 0$. If this assumption becomes invalid and most of the frames contain vehicles, it causes accumulation of the vehicle within the frame at the rate of F_t/t . The term \bar{F} is called the residue of the lost that causes background pollution significantly. If distance matrix is computed using this polluted background, the intensity of vehicle gets reduced. This residue is static in nature, since it accumulates over several points on the background surface and stays on those points for a longer time. The rate of accumulation is termed as accumulation limb and the rate of decay of this accumulated residue is termed as diminishing limb. The presence of this residue affects both background and foreground. Thus, the distance matrix becomes,

$$\delta_t = F_t + m \frac{t}{2} - \bar{F} \quad (3.18)$$

Equation (3.18) represents that the intensity of F_t gets reduced by the accumulated \bar{F} . The worst case scenario occurs when vehicle stays for the whole sampling period T . Then, the value of \bar{F} becomes F_t . Consequently, the vehicle within the distance matrix gets eliminated and no vehicle will be detected. To overcome this error, a condition needs to be applied on both collection and averaging of frames. Particularly, the selected frames must consist of a small number of

vehicles. However, the ingress of vehicle into the frame cannot be controlled. Alternatively, ingress of pixels can be controlled using a logical way. Incorporating the aforementioned error into the solution, the estimated background B_e deviates from the actual background B_i by a critical value δ_c . From recollection of previous derivation, the amount of error caused by the estimated background is as follows.

$$\int_0^t I_i dt = \int_0^t B(t) dt \pm \delta_c t = B^* t \pm m \frac{t^2}{2} \pm \delta_c t \quad (3.19)$$



(a) Before incorporating pollution controlling parameter



(b) After incorporating pollution controlling parameter

Figure 3.4 Representation of residue of the lost \bar{F} due to accumulation of vehicle.

The solution of the problem can be derived from the following steps. Firstly, any frame can be expressed as,

$$I_i = AB^* + DF_i \pm EK_i + G$$

Where, A, D, E, G are coefficients. Integrating the both sides, the following equation can be obtained.

$$\int_0^t I_i dt = AB^* t + D \int_0^t F_i dt + Em \frac{t^2}{2} + Gt \quad (3.20)$$

Equating the coefficients of Equation (3.17) and Equation (3.18), the value of these coefficients can be obtained.

$$A = 1; D = 0; E = 1; G = \pm \delta_c$$

So, the condition for pollution suppression becomes,

$$|I(x, y, t) - B(x, y, t)| \leq \delta_c \quad (3.21)$$

From this condition, it is apparent that the ‘residue of lost’ will be terminated if and only if the current frame differs from the elected background by critical different (δ_c) or less. More specifically, if $I(x, y, t)$ differs less than δ_c (case 1) from $B(x, y, t)$, $I(x, y, t)$ is preferred for background estimation. However, if the difference is greater than δ_c (case 2), a new parameter (rate of intrusion r_i) needs to be introduced to control the intrusion of vehicle into the estimated

background. The value of r_i varies from 0 to 1. However, it is recommended to keep this value as low as possible, such as, from 0 to 0.1. Otherwise, the accumulation of vehicles in the estimated background increases with the increment of this parameter value. Moreover, this parameter value should not be kept to zero, which may retard the estimation model to incorporate illumination variation. Thus, it acts as a valve to the accumulation problem. The precondition equation becomes as follows.

$$I'_n = I'(x, y, n) = \begin{cases} I(x, y, n), & \text{if } |I(x, y, n) - B_e(x, y, n-1)| \leq \delta_c \\ (1-r_i)B_e(x, y, n-1) + r_i I(x, y, n), & \text{otherwise} \end{cases}$$

These two parameters (δ_c, r_i) are called pollution controlling parameters. By applying this precondition on each frame, the background estimation can be done using Equation (3.22).

$$B_e(n) = \frac{1}{N} \sum_{n=i}^{i+N} I'_n \quad (3.22)$$

Remark. The accumulation rate becomes $r_i \frac{F_t}{T} \xrightarrow{\rho} r_i \frac{F_n}{N}$; where, the value of r_i is very small.

Moreover, the total error due to accumulation becomes $r_i \bar{F}$ from \bar{F} .

Limitation. The main limitation of this model is that it can accommodate only half of the illumination variation which can be seen from Figure 3.3(b). This is mainly due to the averaging effect by the model.

3.4.2. Model 2 (M2)

To overcome the limitation of M1, a new parameter—resampling interval (N_r), is introduced. This parameter facilitates dividing the entire time span $T, T = t_2 - t_1$ into small intervals. From Figure 3.3(b), it is evident that with the time span increment, the amount of error also increases. If the operation is confined within a smaller interval, as in Figure 3.3(c), the error reduces to a minimum value. N_r represents the predefined number of frames after which the estimation process restarts without recalling the previous frames. If this process is included in the estimation equation, Equation (3.22) looks like as follows,

$$B_e(n) = \frac{1}{n - qN_r} \sum_{n=qN_r+1}^{(q+1)N_r} I'_n, q = \left\lfloor \frac{n}{N_r} \right\rfloor \quad (3.23)$$

If N_r frames lie within T_r , the error occurs due to the averaging process as in Equation (23) is $mT_r / 2$. Since $T_r < t$, this error is less than that of Equation (3.22).

Limitation. It can be seen that after each interval the averaging process starts and the background estimated in the previous interval gets erased. However, if the sample size is small,

the estimated background is not a representative one. Moreover, if $(n - qN_r)$ is small, the value of $F_{avg} \gg 0$ and the risk of accumulation of vehicles within the background becomes very acute. Like-wise, this problem arises when limited number of frames is available at beginning of the model execution.

3.4.3. Model 3 (M3)

To eliminate the limitations of M1 and M2, a new parameter (sample lag factor N_r) is introduced to memorize the estimated backgrounds and to prevent accumulation of vehicles within the background. In particular, N_r is a weightage factor over the previously estimated backgrounds as showed in Figure 3.3(d). It should be noted that there is no estimated background available at the very beginning of the averaging process by Equation (3.24). Thus, in that case, the static background is used instead of the estimated background. The static background B^* is estimated from median of several frames. It should be noted that resampling interval (N_r) and sample lag (N_r) are considered as luminance controlling parameter.

$$B_e(n) = \begin{cases} \frac{1}{(n-qN_r)+N_r} \left(\sum_{n=qN_r+1}^{(q+1)N_r} I_n + N_l B_e(n-(q-1)N_r) \right), & \text{if } q > 0 \\ \frac{1}{(n-qN_r)+N_r} \left(\sum_{n=qN_r+1}^{(q+1)N_r} I_n + N_l B^* \right), & \text{if } q = 0 \end{cases}, q = \left\lfloor \frac{n}{N_r} \right\rfloor \quad (3.24)$$

As the final background model (M3) has been developed, traffic detection task would be achieved by a per pixel binary threshold model. In this regard, the distance matrix δ_n of frame n is determined using Equation (3.25) incorporating the estimated background $B_e(n)$ (Equation 3.23) and the input image I_n . Afterwards, the distance matrix is fed into Equation (3.26), which considers that a higher threshold value should be provided to a lower difference value within the distance matrix and vice-versa. This is essentially a simplified linear equation between difference and threshold values. It gives the per pixel threshold using the maximum and minimum thresholds (τ_{max} and τ_{min}).

$$\delta(n) = |I_n - B_e(n)| \quad (3.25)$$

$$\tau(x, y, n) = \tau_{max} - \frac{\tau_{max} - \tau_{min}}{\delta_{max} - \delta_{min}} \times (\delta(x, y, n) - \delta_{min}) \quad (3.26)$$

Where,

$$\delta_{\max} = \text{Max}(\delta(x_1, y_1, n), \delta(x_1, y_2, n), \delta(x, y, n), \dots, \delta(x_H, y_W, n))$$

$$\delta_{\min} = \text{Min}(\delta(x_1, y_1, n), \delta(x_1, y_2, n), \delta(x, y, n), \dots, \delta(x_H, y_W, n))$$

The pixels having greater difference value than the corresponding threshold are classified as foreground as in Equation (3.27). This results in a binary image (b).

$$b(x, y, n) = \begin{cases} 1, & \text{if } \delta(x, y, n) \geq \tau(x, y, n) \\ 0, & \text{otherwise} \end{cases} \quad (3.27)$$

3.5. Summary

Background subtraction is a very common approach in vision based traffic detection. However, an accurate background is needed to classify the foreground correctly. Unfortunately, it is difficult to get such background as it is not static and it is occupied with objects most of the time. Thus, the necessity of accurate background modeling emerges for accurate traffic detection. In this chapter, an efficient background model has been derived. In this context, three theorems were proposed to define different components of an image, which ultimately differentiates background and foreground. These definitions and theorems fueled the formulation of the problem, where the concept of static background is utilized. As a solution to this problem, successively three dynamic background models were developed; however, the first model cannot capture illumination variation to the full extent and the second model causes severe accumulation problem due to lack in available frames for dynamic background estimation. Hence, the third model was developed to overcome the limitations of the previous ones which became possible because of incorporating two types of parameters: (1) luminance controlling parameters; and (2) pollution controlling parameters. Luminance controlling parameters capture the illumination variation; whereas the pollution controlling parameters retard the model pollution due to accumulation of traffic into the background. Besides, a new per pixel binary threshold model related to the final model was developed for the foreground segmentation. It uses a linear relationship between threshold and distance matrix to find per pixel threshold. Although this threshold model contains two parameters, one of them is significant while making binary decision.

Chapter 4

Pixel Based Heterogeneous Traffic Measurement

4.1. Introduction

With the rapid growth of vehicles in the road network, measuring of traffic flow is becoming more and more urgent. Noticeably, estimation of travel time, vehicle trajectory, and controlling traffic signal requires flow data. This data is also used to estimate fundamental parameters: jam density, capacity, and free flow speed. While these parameters are used in building traffic flow simulation models, the calibrated road network provides a platform to analyze the important changes in traffic flow due to various control and management strategies. The prerequisite for accurate traffic measurement is the detection of vehicles precisely, which can be done by means of magnetic loop, microwave or ultrasound detectors, and video sensors. However, video based sensor, a state of art approach, is the most consummate over other detectors for real-time vehicle detection considering its ease of installation, maintenance and wide area monitoring coverage. Video sensors revolve on image processing algorithms that require image acquisition, preprocessing, segmentation, representation, recognition and interpretation. Images are retrieved from a source through image acquisition, background noise is removed applying preprocessing, and partitioning is obtained via segmentation. After that, vehicle is detected and it is interpreted as traffic measurement. Nonetheless, the accuracy of vehicle detection decreases under uncontrolled environment. This environment is incorporated with various challenges such as shadow, sudden and gradual illumination variation, camouflage, static vehicle (congestion) and noise associated with video dataset. These result in inaccurate vehicle detection. For instance, illumination variation results in increase or decrease of pixel intensity generating false positive detection and presence of shadow at vehicle edges leads to overestimation. Moreover, performing accurately both for stationary and moving vehicle conditions is another challenge for detection algorithms. Thus, in real-time context, the vehicle detection algorithm should be efficient in detecting vehicle under such uncontrolled environment. Thus, this chapter endeavors to develop an efficient traffic measurement tool, which can address the aforementioned detection challenges.

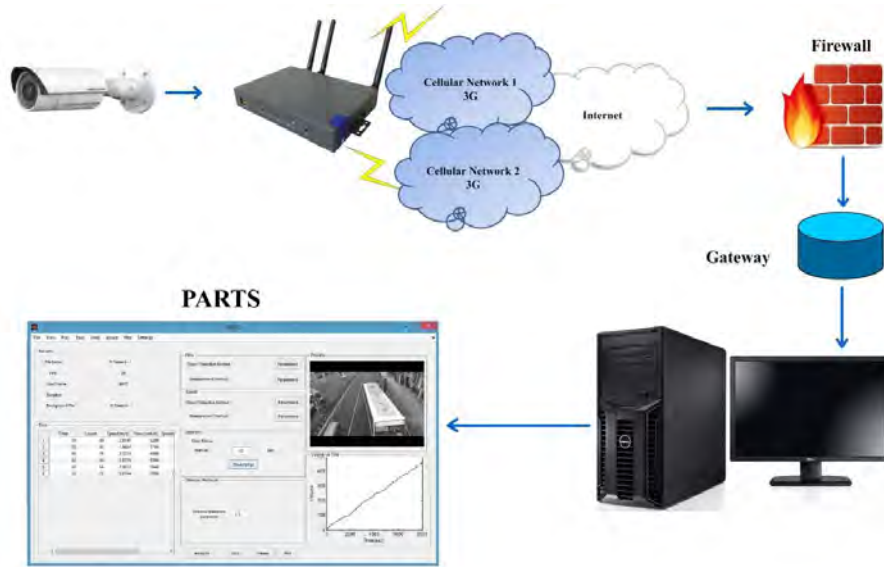


Figure 4.1 Connectivity among different components of vision based wireless surveillance system.

4.2. Methodology for Traffic Measurement

An efficient system is that which is divided into modules and each module has separate operation to perform; whereas modules can be added, removed or replaced as per requirement. The system developed in this research contains seven modules: (1) Live Video Capture; (2) User Input; (3) Background Estimation; (4) Vehicle Detection; (5) Shadow Removal; (6) Flow Measurement; and (7) Speed Measurement. Figure 1 shows the connectivity of PARTS with the other components of the system. PARTS is capable of taking both on-line and off-line video images as input source. In case of on-line source, surveillance camera uses 3G cellular network to connect to the internet through a firewall protected wireless router. Afterwards, the on-line source is accessed by user through the internet from a server computer using PARTS. However, at first, user needs to feed some information to PARTS, such as strip location or number of segments. While analyzing, background is estimated from the frame input and updated dynamically to adapt to environmental challenges like illumination variation, camera jitter, and tail back. Using this estimated background, differential image is formed by background subtraction. This differential image is then converted into binary/foreground image using different threshold values for each pixel. The threshold values are selected based on a linear threshold-difference ($\tau - \delta$) relationship. Afterwards, connected component labeling is used to detect vehicle from the binary image which connects the foreground pixel considering the neighborhood pixels. Shadow removal technique – PNS is applied on each detected vehicle before using the flow and speed measurement mechanisms. Figure 4.2 shows the whole

mechanism of PARTS using a simple flow chart and the algorithms involved are explained in details in the following sections.

4.2.1. Live Video Capture Module

A live video capture module was developed to digitize real-time video signals into readable images from remote video sources, such as IP camera. These images were then sent to the server. The combination of 'HIKVISION Network' camera and 'WLINK' Industrial 4G router (supports 3G also) enabled the full streaming of the video in real-time for this work. The camera can record 25 frames of 3MP (Megapixel) resolution each per second. Whereas 4G router can connect to the server using two 3G (third generation of mobile telecommunications technology) enabled mobile SIM (Subscriber Identity Module) alternatively, if one is out of order. The 3G enabled SIMs made the whole system wireless and mobile. The system stored the captured video automatically in a buffer located in the server in MPEG4 (.mp4) format. The later module uses this buffer to estimate background by reading the video frame by frame.

4.2.2. User Input Module

User input module requires five information to be provided by user: (1) strip location for flow measurement; (2) strip location for speed measurement; (3) field distances; (4) data interval; (5) number of segments for speed measurement. The locations of the strip needs to be given in Cartesian coordinate system considering the upper left corner of the frame as (0,0). The location information requires position and dimension of the region of interest. These values can be provided manually by the user or the user can press the push button 'From Frame' to do it graphically (Figure 4.3(a)). Pressing the button provides a window for user to select the region by drawing a blue rectangle on a frame selected from the input stream. Additionally, double clicking on the enclosed area takes coordinates of the selected location automatically. The field distances (converted into pixel distances) for camera calibration have to be provided in a table for different position which is illustrated in Figure 4.3(b). Figure 4.3(c) shows the window which facilitates the data collection interval. The flow and speed are measured for this selected interval.

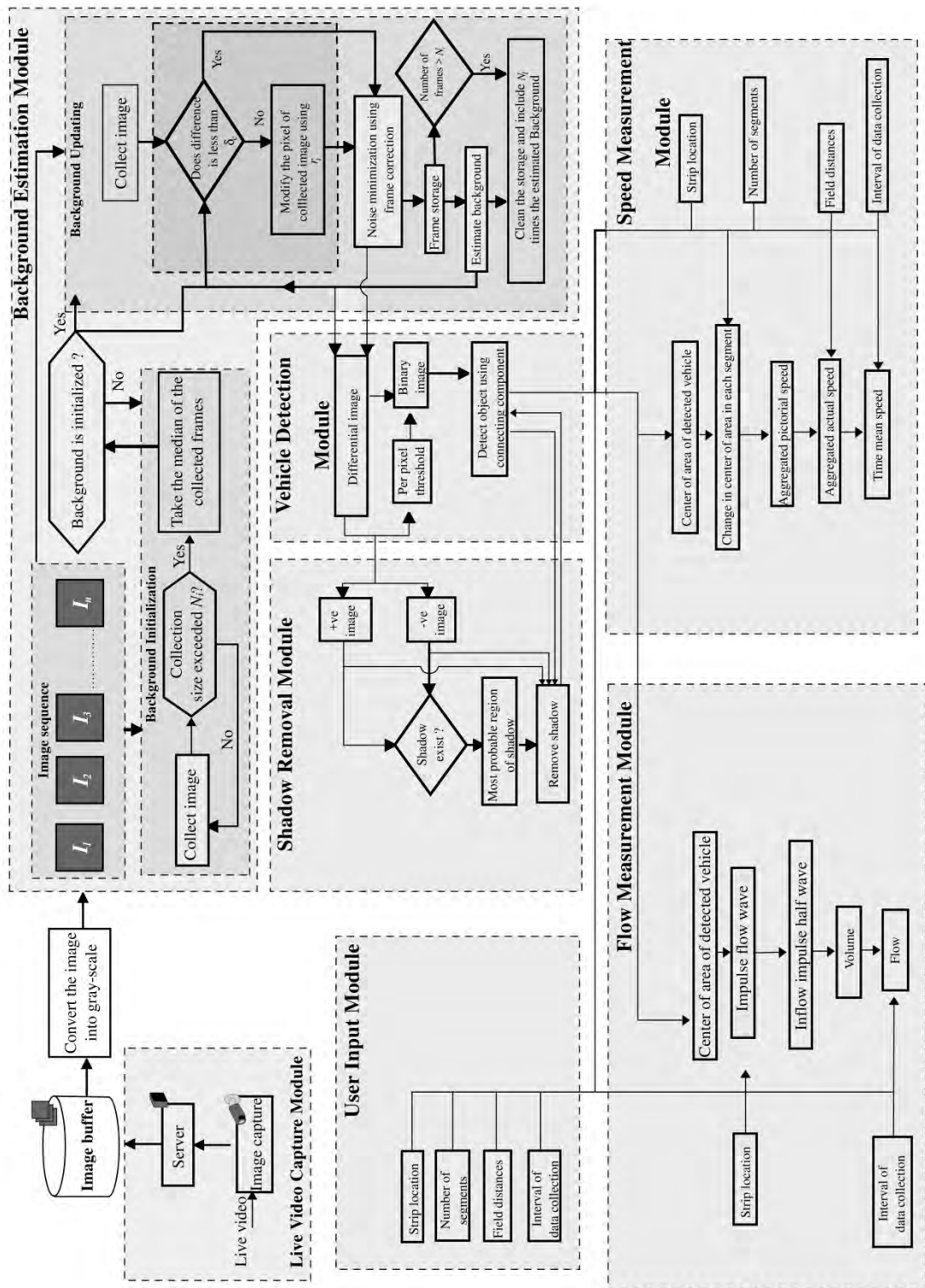
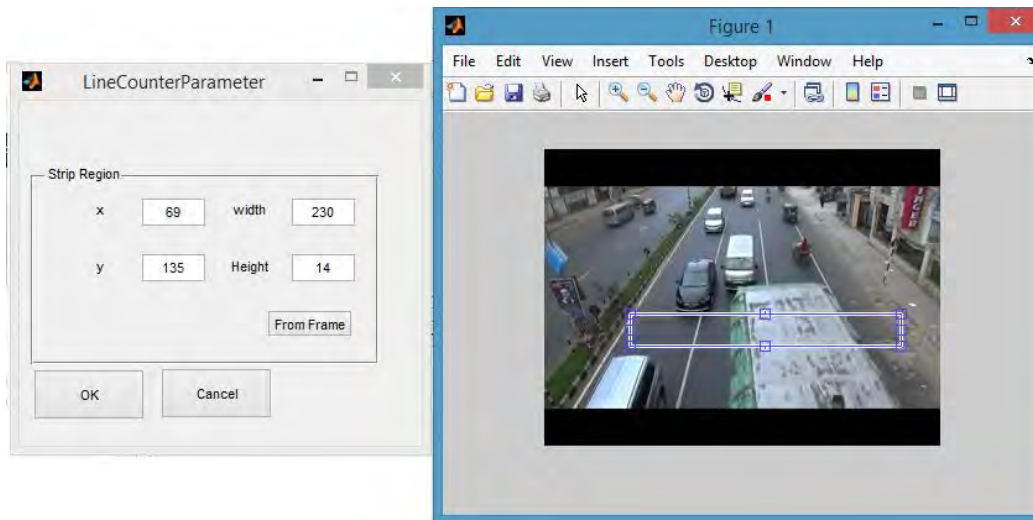
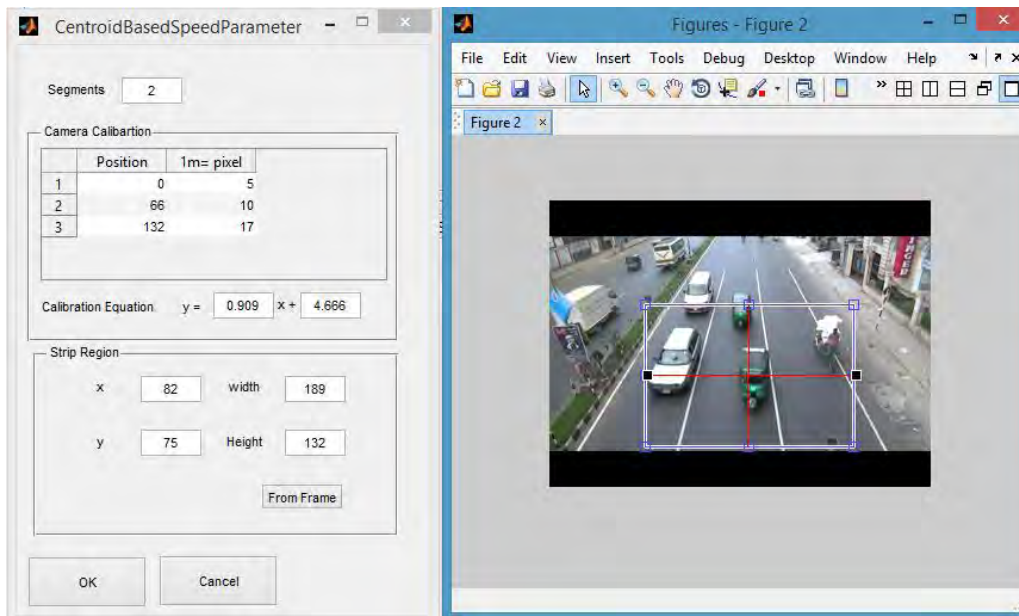


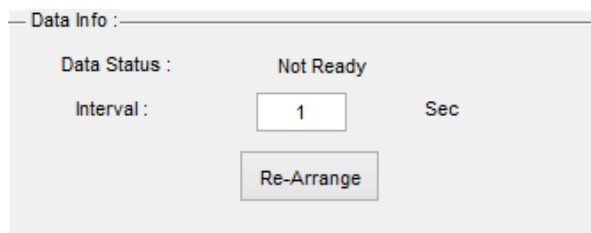
Figure 4.2 Flow chart showing different modules incorporated in PARTS tool.



(a) Region selection window for flow measurement



(b) Region selection window for speed measurement



(c) Interval selection window

Figure 4.3 User Input Module.

4.2.3. Background Estimation Module

For background estimation, an underlying assumption in PARTS is that there is no massive camera movement which may cause any loss either to the estimated background or to the

detected object. PARTS addresses sudden and gradual illumination variation, and accumulation of static objects in the background using mathematical formulation. The brief description of the formulation is as follows.

The real-time RGB images I_{RGB} obtained from the source are converted into gray-scale (monochromatic) images I using weighted mean of the color channels. Then, using equation (1) the median (\tilde{B}) of N_i number of frames is determined to initialize background; assuming that the background is presented in more than $N_i/2$ frames. N_i is determined considering the approximate average aggregate vehicle speed \bar{v} and the number of frames the video captures per second (FPS). After initialization, each of the forthcoming frames is fed into equation (2) for estimating background $B_e(n)$. However, filtration is done by equation (3) on the incoming frame incorporating previously estimated background $B_e(n-1)$; whereas \tilde{B} is used if no estimated background is available. The filtration process compares each pixel of the incoming frame with the previously estimated background. If the difference is less than a critical difference (δ_c), it skips the filtration process. And if the difference is higher, it modifies the incoming pixels using an intrusion rate r_i . In the proposed methodology, δ_c and r_i are utilized as pollution controlling parameters that control (or retard) accumulation of stopped vehicle pixels into the estimated background B_e .

Furthermore, PARTS uses resampling interval (N_r) to cope up with gradual and sudden illumination variation. Smaller value of N_r is used for accommodating both gradual and sudden illumination variation. It helps the estimation model to break the illumination variation into finite difference problem which needs smaller step size (or N_r). Conversely, smaller value of N_r causes errors in the estimated background. It is due to the presence of vehicles within the resampling interval. However, the process forgets previously estimated background immediately when a new interval starts. In this regard, sample lag factor (N_l) is used to give a weightage to the previously estimated background. This parameter keeps a link among the estimated backgrounds at each interval until the process ends. Here, N_r and N_l are utilized as luminance controlling parameters for addressing sudden and gradual illumination variations in the estimated background. Additionally, before storing the frames, it performs geometric mean between filtered image and the estimated background to reduce the effect of camera jitter.

$$B^* = \tilde{B}(N_i) = \text{Med}(I_1, I_2, I_3, \dots, I_{N_i}) \quad (4.1)$$

$$B_e(n) = \begin{cases} \frac{1}{(n-gN_r)+N_l} \left\{ \sum_{n=gN_r+1}^{(g+1)N_r} \sqrt{I(n)B_e(n-1)+N_l B_e(n-(g-1)N_r)} \right\} & \text{if } g>0 \\ \frac{1}{(n-gN_r)+N_l} \left\{ \sum_{n=gN_r+1}^{(g+1)N_r} \sqrt{I(n)B^*+N_l B^*} \right\} & \text{if } g=0 \end{cases} \quad (4.2)$$

$$I(x, y, n) = \begin{cases} I(x, y, n), & \text{if } I(x, y, n) - B_e(x, y, n-1) \leq \delta_c \\ (1-r_i)B_e(x, y, n-1) + r_i I(x, y, n), & \text{otherwise} \end{cases} \quad (4.3)$$

Where,

$$g = \left\lfloor \frac{n}{N_r} \right\rfloor, \forall g \in \mathbb{Z}$$

4.2.4. Vehicle Detection Module

In this module, the differential image $\delta(n)$ of frame n is determined using equation (4.4) incorporating the estimated background $B_e(n)$ and the corrected input image $I_c(n)$.

$$\delta(n) = |I_c(n) - B_e(n)| \quad (4.4)$$

Where,

$$I_c(n) = \sqrt{I(n)B_e(n-1)}$$

Afterwards, the differential image is fed into equation (4.5) which considers that a higher threshold value should be provided to a lower difference value within the differential image and vice-versa. This is essentially a simplified linear equation between δ and τ , which ultimately gives the per pixel threshold using maximum and minimum thresholds (τ_{\max} and τ_{\min}). This linear relationship adopts the shape of the drawdown curve of the windshield and does not allow the windshield to get eliminated during the threshold process.

$$\tau(x, y, n) = \tau_{\max} - \left| \frac{\tau_{\max} - \tau_{\min}}{\delta_{\max} - \delta_{\min}} \right| \times (\delta(x, y, n) - \delta_{\min}) \quad (4.5)$$

Where,

$$\delta_{\max} = \text{Max}(\delta(x_1, y_1, n), \delta(x_1, y_2, n), \dots, \delta(x_h, y_w, n))$$

$$\delta_{\min} = \text{Min}(\delta(x_1, y_1, n), \delta(x_1, y_2, n), \dots, \delta(x_h, y_w, n))$$

w = Number of pixels along the width (B) of the image

h = Number of pixels along the height (L) of the image

Then the pixels having greater difference value than the corresponding threshold are classified as foreground using equation (4.6). This results in a binary image (b).

$$b(x, y, n) = \begin{cases} 1, & \text{if } \delta(x, y, n) \geq \tau(x, y, n) \\ 0, & \text{otherwise} \end{cases} \quad (4.6)$$

Afterwards, disk shaped morphological operations (dilation and erosion) are used to improve the quality of the binary image. Connected component labeling algorithm is used to connect pixels marked as foreground in the binary image. This algorithm traverses the binary image, and merges the foreground pixels into objects based on the connectivity and relative values of their neighbors.

4.2.5. Shadow Removal Module

The presence of shadow causes false positive detection when it is alone on the road; whereas, it causes occlusion when any vehicle steps into it. Interestingly, shadow too is an object related to illumination. It has the same texture but lower intensity than the object/background on which it casts. In PARTS, Positive-Negative Segmentation (PNS) of differential image is adopted for removing shadow. The differential image has two components, one is positive and the other is negative. Equations (4.7) and (4.8) can be used for extracting the positive (δ_+) and negative (δ_-) images respectively. Hereafter, positive (b_+) and negative binary (b_-) images are obtained using equation (4.9). Areas of the positive and negative binary images are obtained using equation (10).

$$\delta_+ = \frac{\delta + |\delta|}{2} \quad (4.7)$$

$$\delta_- = \frac{|\delta - \delta|}{2} \quad (4.8)$$

$$b_{+/-} = b \cdot \delta_{+/-} \quad (4.9)$$

$$A_{+/-} = \sum_{i=1}^w \sum_{j=1}^h b_{+/-}(i, j) \quad (4.10)$$

Using these areas, the most probable equivalent width (w_B) and length (w_L) of the shadow are determined using equations (4.11) and (4.12) respectively.

$$w_B = \frac{A_-}{A_+ + A_-} \frac{B}{2} \quad (4.11)$$

$$w_L = \frac{A_-}{A_+ + A_-} \frac{L}{2} \quad (4.12)$$

Then, an evaluation is performed on the object detected by the “Vehicle Detection Module”. In this evaluation process, a total of four areas are identified by searching from each corner of the object inwards up to w_B and w_L . Within each of these areas, shadow is removed if $\frac{\sum b_-}{\sum b_+} > \tau_s$; where τ_s is the tolerance parameter for shadow removal. Finally, a binary image without any shadow is obtained.

4.2.6. Flow Measurement Module

Flow measurement module uses the binary image generated as above to compute traffic flow. In this regard, at first, impulse flow waves are generated using equation (4.13).

$$q_s(t) = \frac{[C_i(t) - C_i(t+1)]}{\Delta t} \quad (4.13)$$

Here, C_i is obtained from the geometric property (i.e. centroid) resulted from the connected component labeling of the binary image at any time t . It represents the number of vehicles existing within the strip defined by the user. Δt is the time difference between two consecutive frames. The impulse flow waves generated by equation (4.13) are shown in Figure 4.4, which is similar to alternating wave having non-homogenous wavelets. The time difference between two positive peaks represents the time headway between two consecutive incoming vehicles inside the strip. Thus, the relation between impulse flow waves and time headway is utilized to generate the actual flow wave. However, the wave generated from equation (4.13) does not provide flow measurement directly. It reforms in several folds. Positive part of the wave is named inflow impulse wave while the negative part is called the outflow impulse wave. While reforming, the impulse flow waves are rounded and rectified into inflow half waves using half-wave rectifier. The rectifier keeps only the positive portion of the wave; whereas rounding disregards any fractional value of the wave. Since inflow and outflow half waves are identical, only the inflow wave is considered in this research. Hereafter, cumulative sum of the half wave amplitudes is taken. The cumulative sum aggregates the discrete wave information into a single arrival wave which gives the volume at any time. After obtaining the volume, flow at any interval can be computed by performing finite 1st order differentiation of the arrival wave with respect to the user defined time interval. Figure 4.4 shows the detailed procedure of the mechanism.

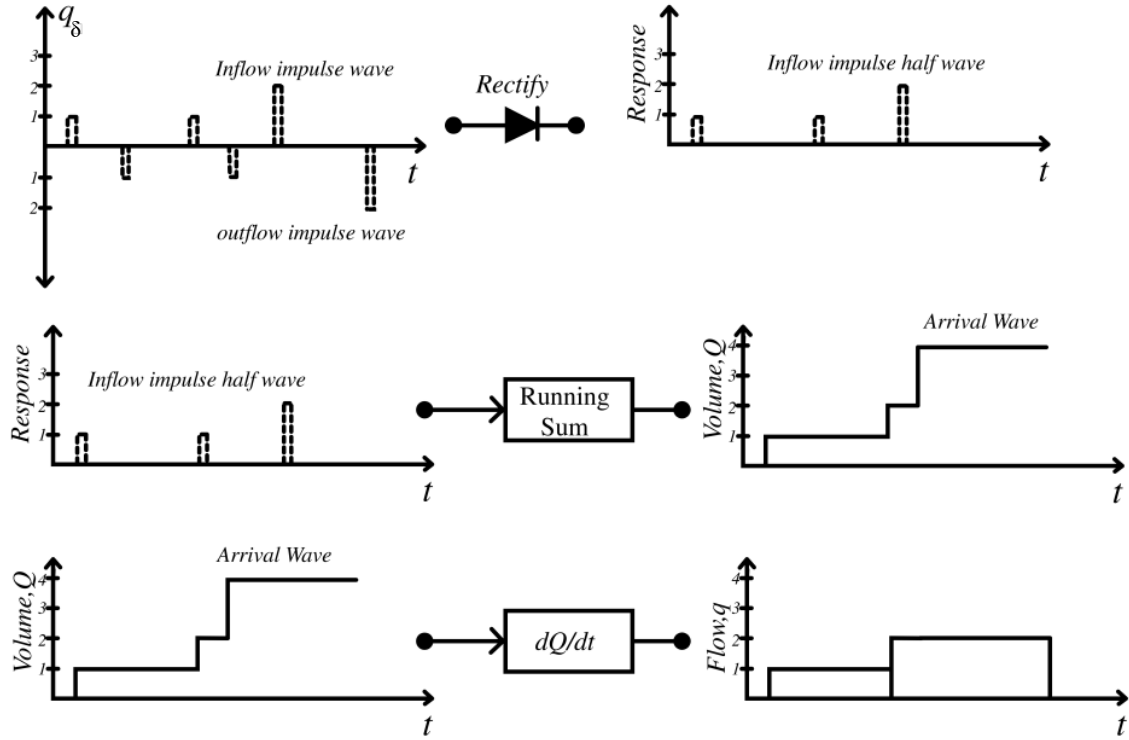


Figure 4.4 Rectifying and cumulating inflow waves for flow measurement.

4.2.7. Speed Measurement Module

In the speed measurement module, the frame is divided into different segments, each of which is too small to occupy vehicles having different speed. Hence, every segment represents the speed of the vehicle considering one segment occupies one vehicle only. The whole approach of speed measurement is illustrated in Figure 4.5. In this approach, speed of the vehicle is determined by measuring the change in center of area of vehicles within the segments using equation (4.14). The area of a vehicle and its centre are computed by equations (4.15) and (4.16) respectively. Afterwards, speeds of the segments are averaged to get aggregated speed of the vehicles within the frame at each instant; it gives the aggregated pictorial speed. Ultimately, aggregated pictorial speed is fed into a calibration equation. The calibration equation relates the actual distance with the pictorial distance considering perception error. The aggregate time mean speed is determined using equation (4.17).

$$d_{k,l} = \sqrt{(\bar{x}_{k,l} - \bar{x}_{k,l-1})^2 + (\bar{y}_{k,l} - \bar{y}_{k,l-1})^2} \quad (4.14)$$

$$A = \sum_{x=1}^w \sum_{y=1}^h b(x,y) \quad (4.15)$$

$$\left. \begin{aligned} \bar{x} &= \frac{\sum_{x=1}^{w-1} \sum_{y=1}^{h-1} xb(x,y)}{A} \\ \bar{y} &= \frac{\sum_{x=1}^{w-1} \sum_{y=1}^{h-1} yb(x,y)}{A} \end{aligned} \right\} \quad (4.16)$$

$$\bar{v}'_l = \frac{\sum_l^N \sum_k^{N_s} \frac{d_{k,l}}{\Delta t}}{N} \quad (4.17)$$

Where, N is the number of observations, N_s is the number of segments in the frame and Δt represents the time interval between frame l and $l-1$. Finally, actual speed is measured using a calibration equation as in equation (4.18),

$$\bar{V}_l = \theta(\bar{v}'_l, \rho(x, X)) \quad (4.18)$$

Where, $\rho(x, X)$ corrects the distorted distances due to decrease in length X when a vehicle moves away at a distance x from the camera (also known as error due to perspective view) and θ converts the pictorial distance into field distance.

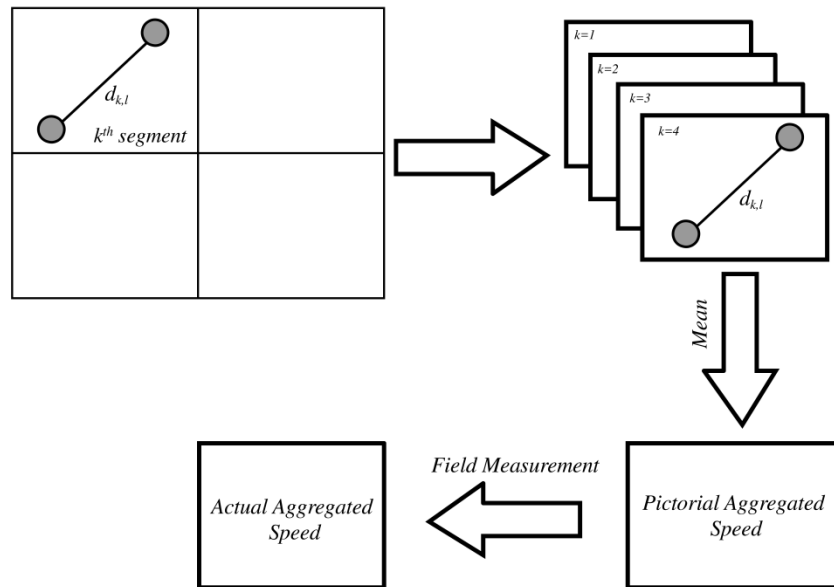


Figure 4.5 Measuring speed from displacement of object centroid at every segment.

4.3. Basic Features of PARTS

PARTS (Pixel Based Heterogeneous Traffic Measurement), a user friendly tool has been developed that holds all the modules as described above. It contains eight interactive panels (Figure 4.6): (1) File info; (2) Data; (3) Flow; (4) Speed; (5) Data info; (6) Shadow Removal;

(7) Preview; and (8) Plotting. The ‘File info’ panel provides information of the data source (i.e. folder location, total number of frames, frames per second). The ‘Data’ panel is the most important panel where the extracted real-time traffic data (flow and speed) are shown. PARTS estimates density from measured flow and speed. Furthermore, the ‘Flow’ and ‘Speed’ panels enable the user to select input parameters related to User Input Module. Most importantly, user defined interval can be set in the ‘Data info’ panel. In the ‘Shadow Removal’ panel, shadow removal parameters can be provided. The ‘Preview’ panel shows the frame that is being analyzed during run-time of the tool. Finally, the ‘Plotting’ panel shows the traffic volume verses time diagram.

Furthermore, several menus have been added to increase the tool’s usability. ‘File’ menu facilitates both off-line and on-line video loading. It also enables users to load the background file from a different source, if needed. The extracted data can be saved in three different formats: 1) .mat; 2) .xlsx; 3) .txt (formatted). ‘View’ menu facilitates users to see the estimated background, differential image and binary image for a particular video frame. ‘Play’ menu gives the facility to play video of dynamic background as well as that of differential and binary formats. ‘Tool’ menu includes the facility of combining images into videos. ‘Plot’ menu gives plotting facility of the extracted traffic data (i.e. Flow verses Time, Speed verses Flow and Speed verses Density). ‘Settings’ menu is kept for the changes in settings of the tool. Specifically, user can choose the type of speed measurement (instantaneous or time mean speed).

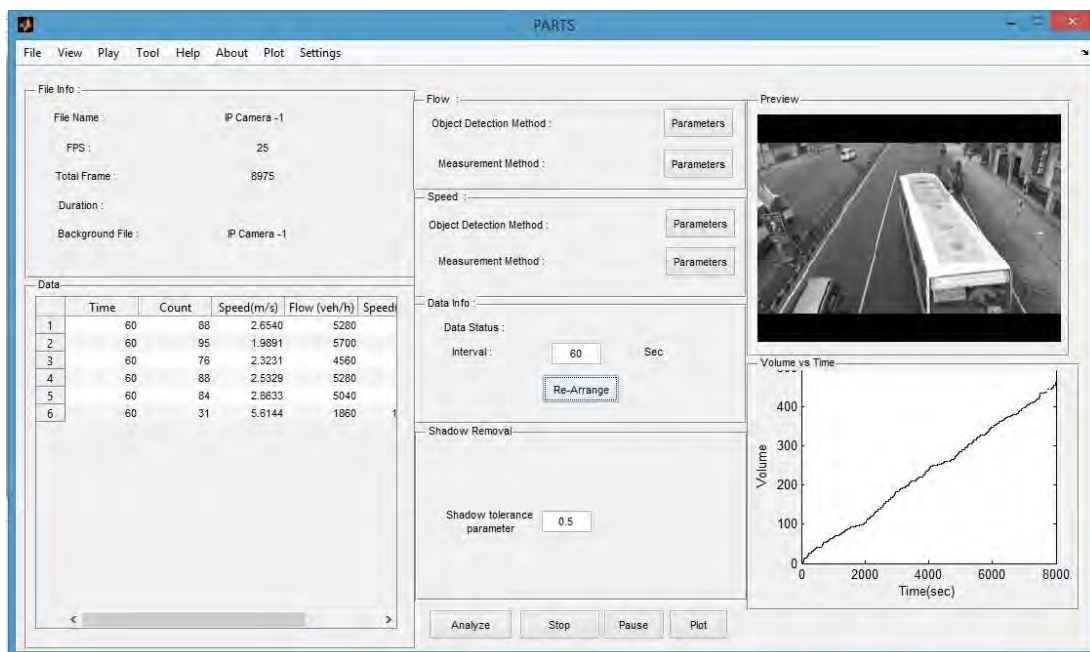


Figure 4.6 Interface of PARTS showing panels and menus.

4.4. Summary

This chapter presents a new methodology for traffic measurement using video sensors. PARTS (Pixel Based Heterogeneous Traffic Measurement), a user-friendly tool that incorporates the proposed methodology, has been developed to facilitate real-time traffic measurement. Particularly, the incorporation of luminance and pollution control parameters result in accurate background estimation. Whereas, the new heuristic dynamic threshold-difference function for determining per pixel threshold results in accurate foreground segmentation. Moreover, shadow is removed considering its physical characteristics by the PNS (Positive Negative Segmentation) technique. This chapter also includes the description of the software interface developed for extracting data using the methodology illustrated in this chapter.

Chapter 5

Crash Probability Estimation

5.1. Introduction

This chapter presents the methodology of estimating crash probability. It includes the video capturing method, vehicle detection method, tracking method, vehicle classification method. This chapter also describes the skeleton of the overtaking model and the formulation of time-to-collision. Ultimately, these two aspects are conjugated within the crash probability model. The methodology of this research is divided into five components: (1) Live Video Capture; (2) Detection; (3) Trajectory Estimation; (4) Vehicle Classification; and (5) Model Formulation. Figure 5.1 shows the flow chart illustrating the connectivity among the different components.

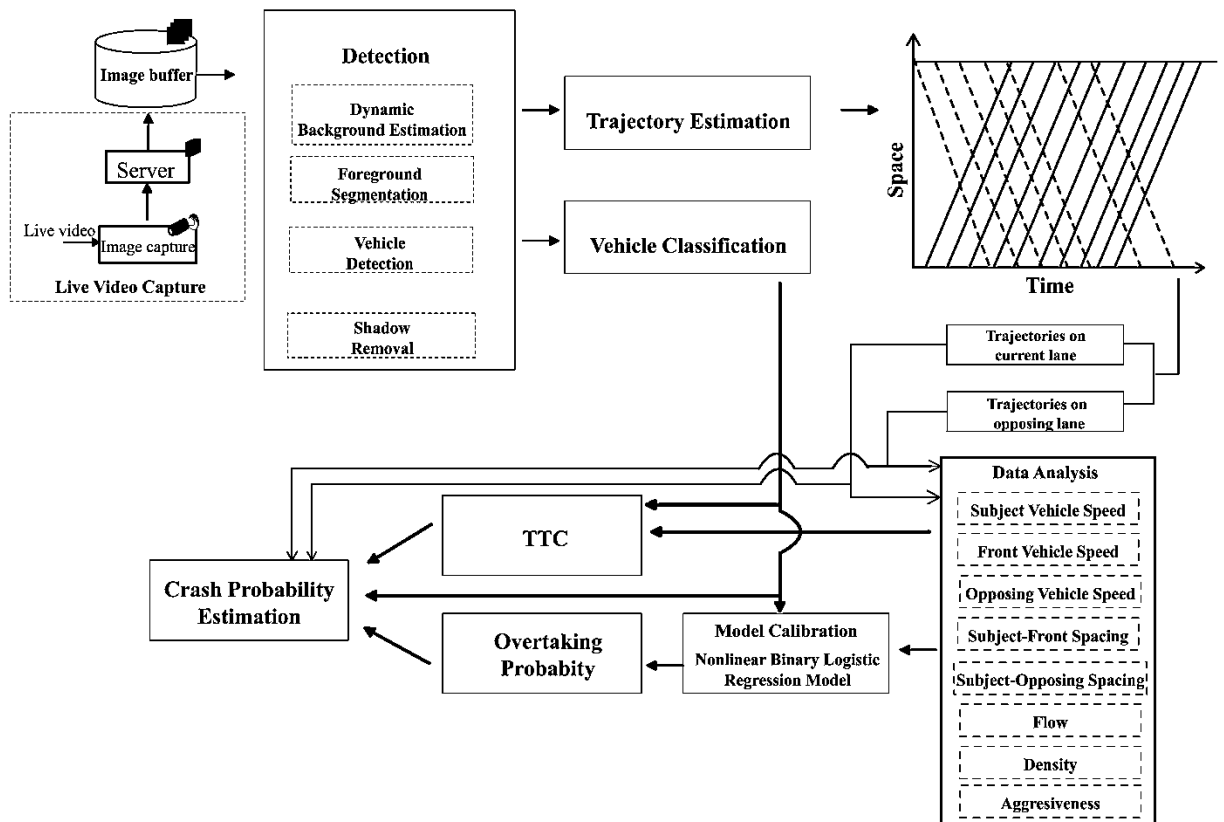


Figure 5.1 Flow chart showing the methodology of crash probability estimation

5.2. Live Video Capture

Real-time video signals were digitized into readable images from remote video sources, such as IP camera, and were sent to a server. The full streaming of the video was done with the

combination of ‘HIKVISION Network’ camera and ‘WLINK’ Industrial 4G router. 25 frames of 3MP (Megapixel) resolution can be recorded per second with this camera. Whereas, the router can operate over two 3G (third generation of mobile telecommunications technology) enabled mobile SIM (Subscriber Identity Module) alternatively, if one is out of order. It stored the captured video automatically in a buffer located in the server in MPEG4 (.mp4) format.

5.3. Detection

In this study, the vehicles were detected from the real-time captured video. It was done in two phases: (1) foreground segmentation; and (2) foreground detection. Foreground (vehicle in particular) segmentation was done incorporating background subtraction technique. It considered subtraction of a dynamic background from each frame within the video. Thus, a dynamic background was needed to be estimated for this process. This dynamically estimated background was then subtracted from corresponding corrected frame resulting in differential image. A set of threshold values were generated based on this differential image. Each pixel in the differential image was compared with the corresponding threshold value to obtain binary image. Any value, that was found to be greater than the threshold value, was considered to be 1, otherwise 0. Consequently, connected component analysis revealed the connectivity among the objects (vehicles) and labeled them as an individual vehicle. It identified the centroid of the vehicles within the binary image. Simultaneously, the shadow attached with each vehicle was also removed considering its physical characteristics. Later, this vehicle centroid was tracked to estimate the vehicle trajectory. For further information about the detection mechanism, readers are requested to see chapter 4.

5.4. Trajectory Estimation

In this component, the geometrical properties of the detected vehicles obtained from the previous subsection were tracked using Kalman filter (34). Kalman filter defined the geometrical properties of the objects as its states, motion model, and measurement equations matrix. Each moving object was defined by its centroid and tracking window. The horizontal and vertical centroid coordinates and the area of the i^{th} object in the k^{th} frame are x_k^i, y_k^i and S_k^i respectively. A cost function $V(i, j)$ was defined using the definition of centroid distance function $CD(i, j)$ and area difference function $AD(i, j)$ between the j^{th} object in the k^{th} and $k + 1^{th}$ frame. These definitions are as expressed in equation (5.1).

$$\left. \begin{aligned}
CD(i, j) &= \frac{\left| \sqrt{(x_k^i - x_{k+1}^j)^2 + (y_k^i - y_{k+1}^j)^2} \right|}{\text{Max} \left| \sqrt{(x_k^i - x_{k+1}^n)^2 + (y_k^i - y_{k+1}^n)^2} \right|} \\
AD(i, j) &= \frac{|S_k^i - S_{k+1}^i|}{\text{Max}_n |S_k^i - S_{k+1}^i|} \\
V(i, j) &= \alpha CD(i, j) + \beta AD(i, j)
\end{aligned} \right\} \quad (5.1)$$

Where, $S^i = A l_k^i h_k^i$ (represents the area of tracking window)

$$\alpha + \beta = 1$$

$\alpha = 0.8$ and $\beta = 0.2$ were considered, as the change in centroid difference function was greater than the area difference function during the lower vertical camera angle. The two objects were more likely to have correspondence for smaller cost function's value. This procedure was repeatedly done to finish the model updating until the moving objects were disappeared. Using the spatio-temporal signature obtained from tracked vehicle centroids, trajectory of each vehicle was obtained. However, the co-ordinates obtained from this process suffered from the perspective view due to camera angle. It was overcome using a calibration equation (5.2), which correlated the pictorial distances with field distance.

$$r_i = \theta(r_i', \rho(x, X)) \quad (5.2)$$

Where, $\rho(x, X)$ corrects the distorted distances due to decrease in length X , when a vehicle moves away at a distance X from the camera and θ converts the pictorial distance into field distance.

5.5. Vehicle Classification

Vehicle classification was done on the basis of geometrical properties of the detected objects revealed in component "Detection". The pictorial area for different vehicles was obtained by multiplying the mid length with mid width of the vehicle in the frame. Afterwards, the area of the vehicles was compared with these predefined areas.

5.6. Model Formulation and calibration procedure

This component illustrates three types of formulations: (1) Overtaking Decision (OD) Model; (2) Time-To-Collision (TTC); and (3) Crash Probability (CP) Estimation Model.

5.6.1. Overtaking Decision Model (OD)

Overtaking is considered as a choice decision taken by the driver to complete a passing maneuver. However, this decision making process is complex and dependent upon different variables. OD can be expressed as,

$$OD = f\left(X \mid \bigcup_{n \in \mathbb{Z}} \theta_n\right) \quad (5.3)$$

Where, θ is the corresponding parameter of the explanatory variable X . However, among these variables, some remain unobserved due to insufficient information. The unobserved variables are expressed as X_u and the observed variables are expressed as X_o . Incorporating these two types of variables, equation (5.3) can be rewritten as equation (5.4),

$$OD = f\left(X_o \mid \bigcup_{i \in \mathbb{Z}} \theta_{io}, X_u \mid \bigcup_{j \in \mathbb{Z}} \theta_{ju}\right) \quad (5.4)$$

When the explanatory variables are correlated with each other, the rate of change of the function $\left(\frac{df}{dX_m}\right)$ is also a function of itself and other explanatory variables. This rate of change can be expressed as equation (5.5).

$$\frac{df}{dX_m} = g_m\left(X_k \mid \bigcup_{k \in (i \cup j)} \theta'_k\right), \forall k \in i \quad (5.5)$$

However, if equation (5.5) is considered for all the variables, a system of differential equations is formed with m number of eigenvalue, which is expressed in equation (5.6).

$$X'_m = g_m\left(X_k \mid \bigcup_{k \in (i \cup j)} \theta'_k\right) \quad (5.6)$$

Conversely, this kind of system leads to m particular solutions. In order to get the exact solution, indefinite integral of equations (5.6) is found, which is presented in equation (5.7).

$$f\left(X_o \mid \bigcup_{i \in \mathbb{Z}} \theta_{io}, X_u \mid \bigcup_{j \in \mathbb{Z}} \theta_{ju}\right) = \sum_{m=1}^{N=|i \cup j|} G_m\left(X_k \mid \bigcup_{k \in (i \cup j)} \theta'_k\right) \quad (5.7)$$

Where,

$$G_m = \int g_m \left(X_k \mid \bigcup_{k \in (i \cup j)} \theta'_k \right) dX_m + \lambda_m$$

Where, λ_m = integral correlation constant.

Substantially, no correlation exists among mutually exclusive sample events. The variables capture the characteristics at the instant of overtaking, $k \in \{m, \emptyset\}$. Either, $k = m$, the gradient of the variables are dependent on its own value (multivariate polynomial equation) or $k = \emptyset$, the gradient is constant (multivariate linear equation). For capturing greater details of the explanatory variables, in this research, k is considered to be m . To understand the significance of the explanatory variables, the best way to represent f is multivariate polynomial form. Equation (5.7) can be written as,

$$OD = \sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \sum_{j \in \mathbb{Z}} \beta_{ju} X_{ju}^{\omega_{ju}} \quad (5.8)$$

Where, $\{\beta, \omega\} \in \theta$.

The unobserved variables cannot be modeled due to lack of information, thus those variables in equation (5.8) are replaced with a constant parameter β_0 stated in equation (5.9).

$$OD = \sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0 \quad (5.9)$$

To make equation (5.9) statistically estimable, a disturbance term is added with it. Then the equation (5.9) takes the form of equation (5.10).

$$OD = \sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0 + \xi \quad (5.10)$$

Where, ξ = disturbance term.

Considerably, explanatory variables are continuous. In contrast, the output variable has only two binary outcomes (0 and 1). Thus, the effect of increasing or decreasing of a certain variable on the OD model cannot be explained properly. Because, after crossing a certain threshold value, the model output will change suddenly zero to one or one to zero. This abrupt change may reduce the scope of explanatory variables inspection. Turning OD model into continuous

probability density function can solve this. It is considered that the disturbance term ξ follows normal distribution and the maximum of the randomly drawn values from that distribution has extreme value type I distribution. According to this, the probability function becomes a logistic regression function represented in equation (5.11),

$$\Pr(OD = 1 | X_{io}) = \frac{\exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0\right)}{1 + \exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0\right)} \quad (5.11)$$

The likelihood function for this model is given by equation (5.12),

$$\log L = f(y | (\beta_0, 1), (\beta_1, \omega_1), \dots, (\beta_n, \omega_n)) = \prod_{i=1}^n \left(\frac{e^{\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0}}{1 + e^{\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0}} \right)^{y_i} \left(\frac{1}{1 + e^{\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{\omega_{io}} + \beta_0}} \right) \quad (5.12)$$

A normal prior is frequently used for the parameters of the logistic regression model. Here, independent normal prior distribution (equation (5.13)) with zero mean and large variance is considered to express the prior ignorance.

$$\left. \begin{aligned} \beta_{io} &\sim N\left(\mu_{\beta_j}, \sigma_{\beta_j}^2\right) \text{ for } j = 0, 1, 2, \dots, n \\ \omega_{io} &\sim N\left(\mu_{\beta_j}, \sigma_{\beta_j}^2\right) \text{ for } j = 0, 1, 2, \dots, n \end{aligned} \right\} \quad (5.13)$$

This setup results in the posterior density function as shown by equation (5.14).

$$f((\beta_0, 1), (\beta_1, \omega_1), \dots, (\beta_n, \omega_n) | y) \propto f(y | (\beta_0, 1), (\beta_1, \omega_1)) f(y | (\beta_0, 1), (\beta_2, \omega_2)) \dots f(y | (\beta_0, 1), (\beta_n, \omega_n)) \quad (5.14)$$

5.6.2. Time-To-Collision (TTC)

In this fragment, analytical formulation of TTC for two-lane undivided highway is presented. Vehicles travelling on the road keep interacting with the neighboring vehicles on the current and opposing lane, which actually generate car-following and overtaking events. The subject vehicle (n) follows the leading vehicle ($n-1$) with the front spacing $S_{n/n-1}$. Thereupon, the vehicle n leaves the current lane and initiates overtaking. After reaching the opposing lane, to avoid crashes with the opposing vehicle and complete overtaking maneuver, the vehicle n should not decrease speed and should not stop overtaking. To avoid crashes, the subject vehicle has to complete the overtaking maneuver and take the gap in front of the vehicle $n-1$. Otherwise, a head-on crash will occur, if the subject vehicle and the opposing vehicle remain

at the same lane. Likewise, one of the most important factors affecting the crash between the subject and leading vehicle is time-to collision (TTC). TTC is calculated for the current time step t based on the equation (5.15). Equation (5.15) is derived from the laws of dynamics. Spacing between subject vehicle (n) and the opposing vehicle (k), speed and acceleration of the subject (V_n^t, a_n^t) and the opposing vehicle (V_k^t, a_k^t) are considered to calculate TTC. The crash potential is higher when the TTC is close to '0'. A larger TTC results in a lower likelihood of crash.

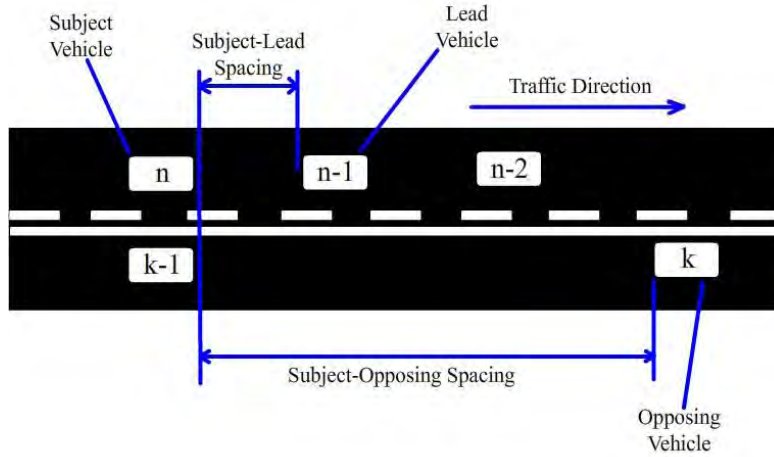


Figure 5.2 Definition of subject vehicle and surrounding vehicles

$$TTC_{nk}^t = \frac{\sqrt{(V_n^t + V_k^t)^2 + 2D_n^k (a_n + a_k)} - (V_n^t + V_k^t)}{(a_n^t + a_k^k)} \quad (5.15)$$

5.6.3. Crash Probability Estimation

Based on the above OD and TTC formulation, the potential that a head-on crash would occur at time step t can be estimated by equation (5.16).

$$\begin{aligned} \Pr^t(C_{nk}^t) &= \Pr^t(OD_n | X_{io}) \times [1 - \text{Max}(\Pr^t(OD_{n-1} | X_{io}), \Pr^t(OD_{n-2} | X_{io}), \dots)]^t \times \Pr^t(C_{nk}^t | TTC_{nk}^t), \text{ if } \Pr^t(OD_n | X_{io}) < 1 \\ &= \Pr^t(C_{nk}^t | TTC_{nk}^t), \text{ otherwise} \end{aligned} \quad (5.16)$$

Where,

$\Pr^t(C_{nk}^t)$: Probability of crash for vehicle n with opposing vehicle k .

$\Pr^t(OD_n | X)$: Probability of the vehicle n to overtake under an observed traffic condition X_{io} at time step t .

$\Pr^t(OD_{n-1} | X_{io})$: Probability of the vehicle $n-1$ to overtake under an observed traffic condition X_{io} at time step t .

$\Pr^t(C_{nk}^t | TTC_{nk}^t)$: Probability that the subject vehicle collides with the opposing vehicle.

$I =$ Overtaking Importance Factor, OIF (depends on type of overtaking: normal/flying/piggyback).

The probability of the subject vehicle to collide is considered as a function of TTC_n^k and is determined from,

$$\Pr^t(C_{nk}^t | TTC_{nk}^t) = \frac{2(TTC_{nk}^t)^{-w}}{1 + (TTC_{nk}^t)^{-w}} \quad (5.17)$$

Where, w is crash frequency parameter (CFP).

Thus, the ultimate crash probability estimation equation becomes,

$$\Pr^t(C_{nk}^t) = \left(\frac{\exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{oio} + \beta_0\right)}{1 + \exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{oio} + \beta_0\right)} \right)_n \times \left[1 - \text{Max} \left(\left(\frac{\exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{oio} + \beta_0\right)}{1 + \exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{oio} + \beta_0\right)} \right)_{n-1}, \left(\frac{\exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{oio} + \beta_0\right)}{1 + \exp\left(\sum_{i \in \mathbb{Z}} \beta_{io} X_{io}^{oio} + \beta_0\right)} \right)_{n-2}, \dots \right) \right]^I \times \left(\frac{2(TTC_{nk}^t)^{-w}}{1 + (TTC_{nk}^t)^{-w}} \right) \quad (5.18)$$

The parameter w depends on the frequency of crash occurrence. High frequency of crash occurrence indicates low w and vice versa. This parameter facilitates a marginal crash value at the highest TTC ensuring higher safety margin.

5.7. Summary

In this chapter, the methodology of estimating crash probability from vehicle trajectory has been presented. A model is developed that estimates head-on crash probability for classified vehicle trajectory. Nonlinear random parameter multivariate binary logistic regression is used to model drivers' overtaking decision. On the other hand, a new formulation of TTC is proposed that considered the dynamic acceleration of the subject vehicle. Finally, the conjugation of OD model and TTC formulation gave the classified crash probability.

Chapter 6

WINBUGS: Markov Chain Monte Carlo

6.1. Introduction

This chapter presents the step by step procedure of Bayesian analysis using Markov chain Monte Carlo (MCMC) methods in WinBUGS. Special characteristics, along with the massive development of computing facilities, have made Markov chain Monte Carlo (MCMC) techniques popular since the early 1990's. MCMC methods are not new, as they were introduced into physics in 1953 in a simplified version by Metropolis and his associates. Nevertheless, it took about 35 years until MCMC methods were rediscovered by Bayesian scientists and became one of the main computational tools in modern statistical inference. Markov chain Monte Carlo techniques enabled quantitative researchers to use highly complicated models and estimate the corresponding posterior distributions with accuracy. In this way, MCMC methods have greatly contributed to the development and propagation of Bayesian theory. MCMC techniques are based on the construction of a Markov chain that eventually "converges" to the target posterior distribution (called stationary or equilibrium). This is the main way to distinguish MCMC algorithms from "direct" simulation methods, which provide samples directly from the target posterior distribution. Moreover, the MCMC output is a dependent sample since it is generated from a Markov chain, in contrast to the output of "direct" methods, which is an independent sample. Finally, MCMC methods incorporate the notion of an iterative procedure (for this reason they are frequently called iterative methods) since in every step they produce values depending on the previous one.

6.2 MCMC Algorithm

A Markov chain is a stochastic process $\{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(t)}\}$ such that

$$f(\theta^{(t+1)} | \theta^{(t)}, \dots, \theta^{(1)}) = f(\theta^{(t+1)} | \theta^{(t)})$$

that is the distribution of θ at sequence $t+1$ given all the preceding θ values depends only on the value $\theta^{(t)}$ of the previous sequence t . Moreover, $f(\theta^{(t+1)} | \theta^{(t)})$ is independent of time t .

Finally, when the Markov chain is irreducible, aperiodic, positive-recurrent, as $t \rightarrow \infty$ the distribution of $\theta^{(t)}$ converges to its equilibrium distribution, which is independent of the initial values of the chain $\theta^{(0)}$.

In order to generate a sample from $f(\theta|y)$, we must construct a Markov chain with two desired properties: (1) $f(\theta^{(t+1)} | \theta^{(t)})$ should be “easy to generate from”, and (2) the equilibrium distribution of interest $f(\theta|y)$.

Assuming that we have considered a Markov chain with these requirements, we then :

1. Select an initial value $\theta^{(0)}$.
2. Generate T values until the equilibrium distribution is reached.
3. Monitor the converge of the algorithm using convergence diagnostics (statistical tests). If diagnostics fail, we then generate more observations.
4. Cut off the first B observations.
5. Consider $\{\theta^{(B+1)}, \theta^{(B+2)}, \dots, \theta^{(T)}\}$ as the sample for the posterior analysis.
6. Plot the posterior distribution.
7. Finally obtain summaries for posterior distribution (mean, median, standard deviation, quantiles, correlations).

6.3. Markov Chain Monte Carlo Simulation using WinBUGS

In this study, the random parameter optimization is performed using WinBUGS, a well-known statistical modeling software. WinBUGS is a statistical software for Bayesian analysis using Markov chain Monte Carlo (MCMC) methods. WinBUGS is a programming language based software that is used to generate a random sample from the posterior distribution of the parameters of a Bayesian model. The user only has to specify the data, the structure of the model under consideration, and some initial values for the model parameters.

After writing the full model code, the data and the initial values in an odc file we need to compile and run the model. This procedure is described using the following steps:

1. Open model specification tool.
2. Check the model’s syntax.
3. Load data.
4. Compile model.
5. Set initial values.
6. Run the MCMC algorithm.

The description of these steps are elaborated as follows:

1. **Open the Model Specification Tool.** Follow the path: *Model*> *Specification* to open the (model) specification tool.

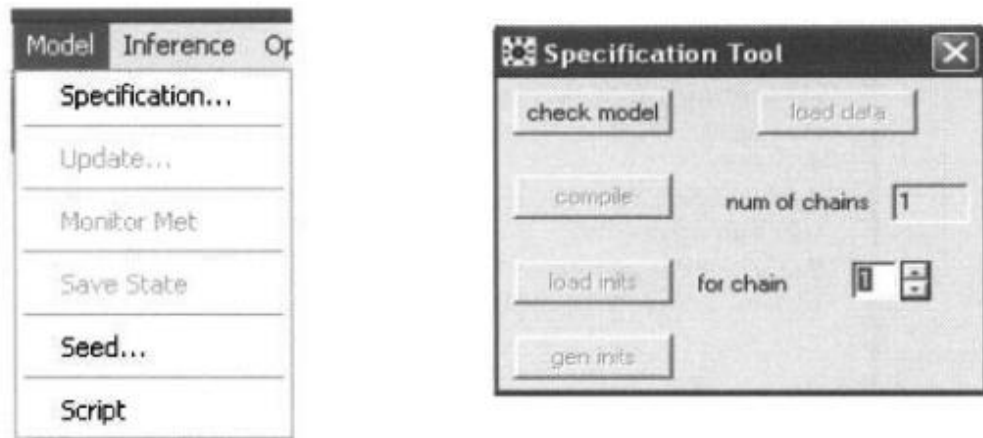


Figure 6.1 Specification Tool.

In this way, the model specification tool appears on the screen. This tool includes all the basic operations needed to initialize the MCMC algorithm (checking the model’s code syntax, loading the data, compiling the model, and setting the initial values) and specify the number of chains that we wish to generate.

2. **Check the syntax of the model:** Highlight the command model and press the check model box of the model specification tool.

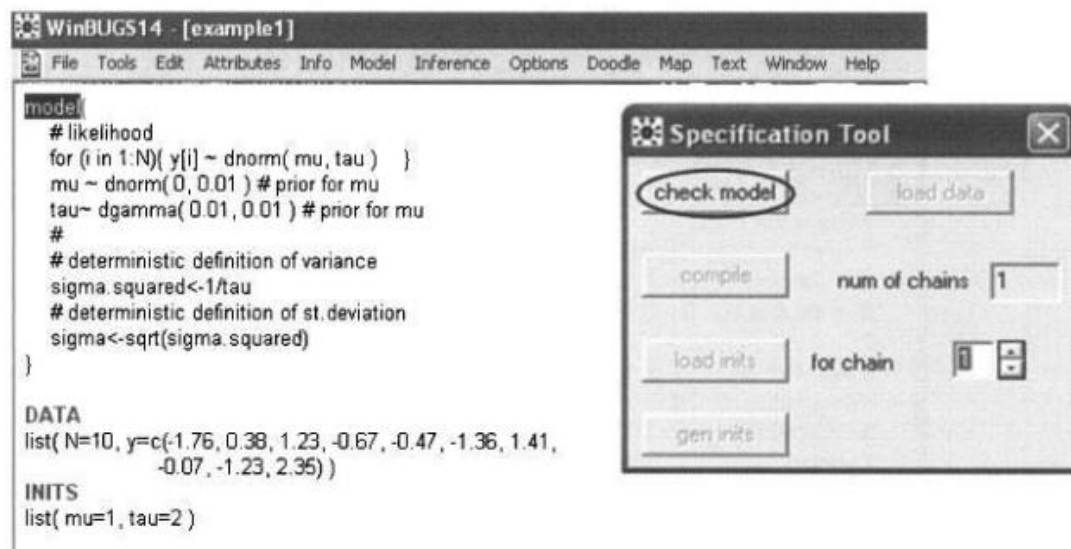


Figure 6.2 Checking the syntax.

WinBUGS checks the model's syntax starting from the first character of the highlighted command. If no command is highlighted, then the check begins at the top of the opened file or window. We recommend always highlighting the desired model command to avoid problems, especially if multiple model codes are included in the same odc file or window. If a problem in the syntax exists, then an indication is given in the lower left of the WinBUGS window while the cursor is placed at the location where the error was detected.

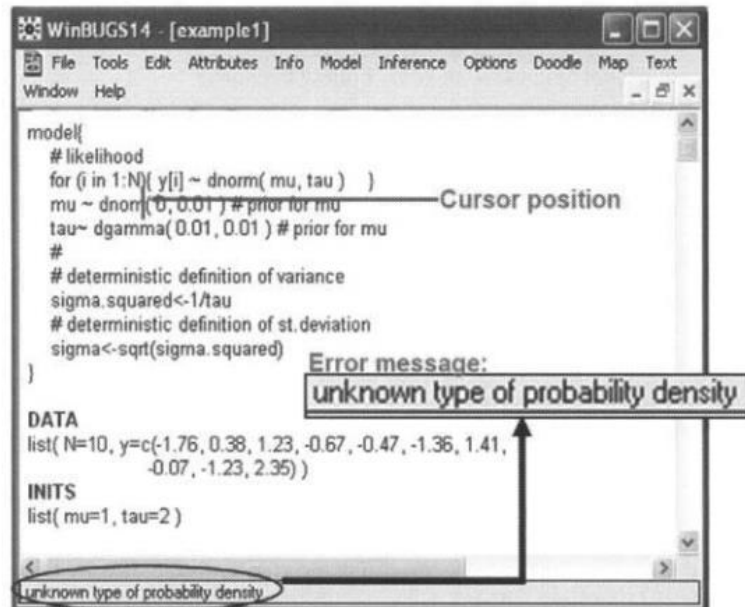


Figure 6.3 Debugging the model.

Otherwise, the message model is syntactically correct appears in the same position.

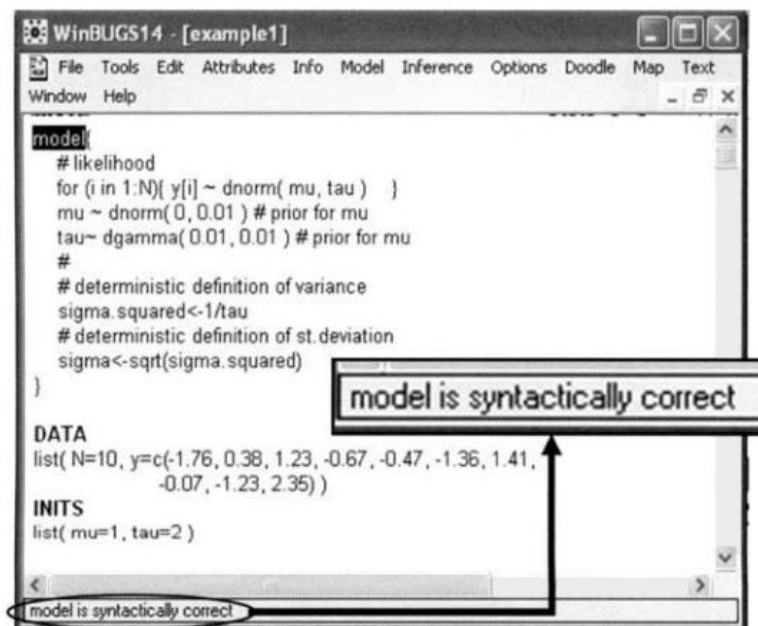


Figure 6.4 Model Correctness.

3. Load the data. Highlight the word list in the data list format and press the load data box of the model specification tool.

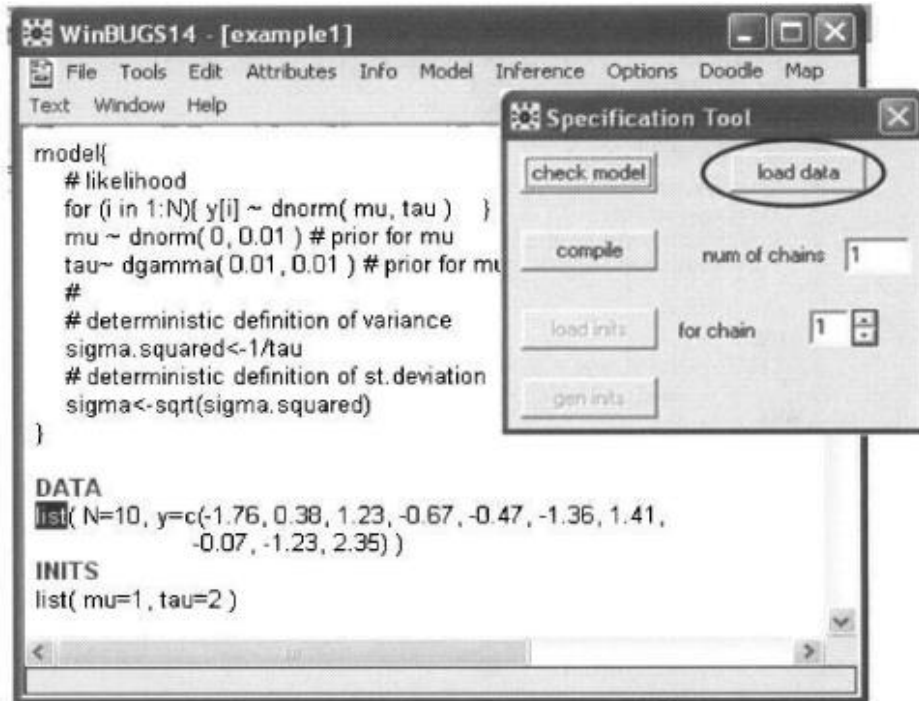


Figure 6.5 Data input.

When the data are defined in a rectangular format then we highlight the first row of the data where the names are declared.

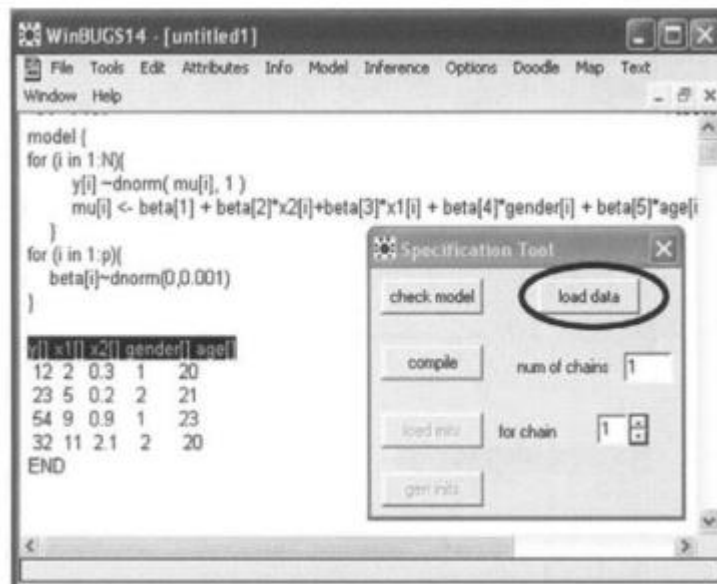


Figure 6.6 Tabular Data Input.

If a set of data is loaded successfully, then the message data loaded will appear at the bottom left of the WinBUGS window (status bar). Otherwise, an error message will appear in the same position.

```
list( aa=c(1,2,3,4,5,6,7,8,9,10))
list( n=4, p=5)
y[] x1[] x2[] gender[] age[]
12 2 0.3 1 20
23 5 0.2 2 21
54 9 0.9 1 23
32 11 2.1 2 20
END
```

Figure 6.7 Sample Data structure.

- 4. Compile Model.** After all data are loaded, press the compile box in the model specification tool.

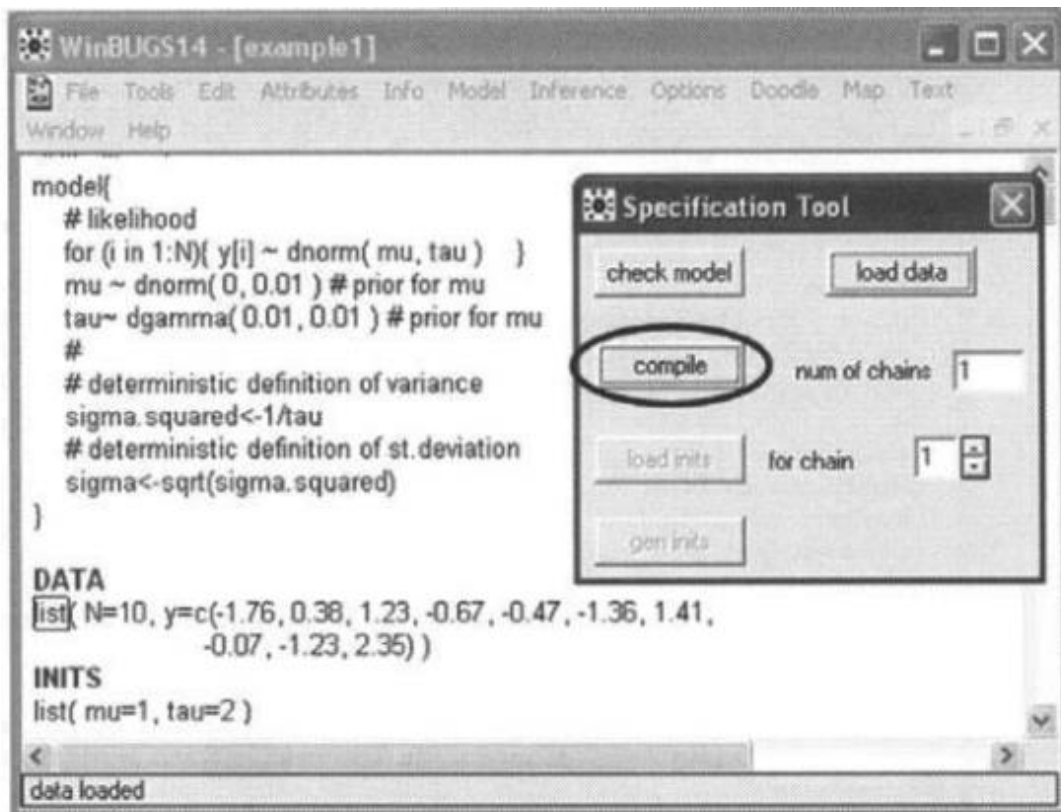


Figure 6.8 Data Loading.

If the compilation is successful then the message model compiled will appear in the status bar otherwise an error message will appear in the same position.

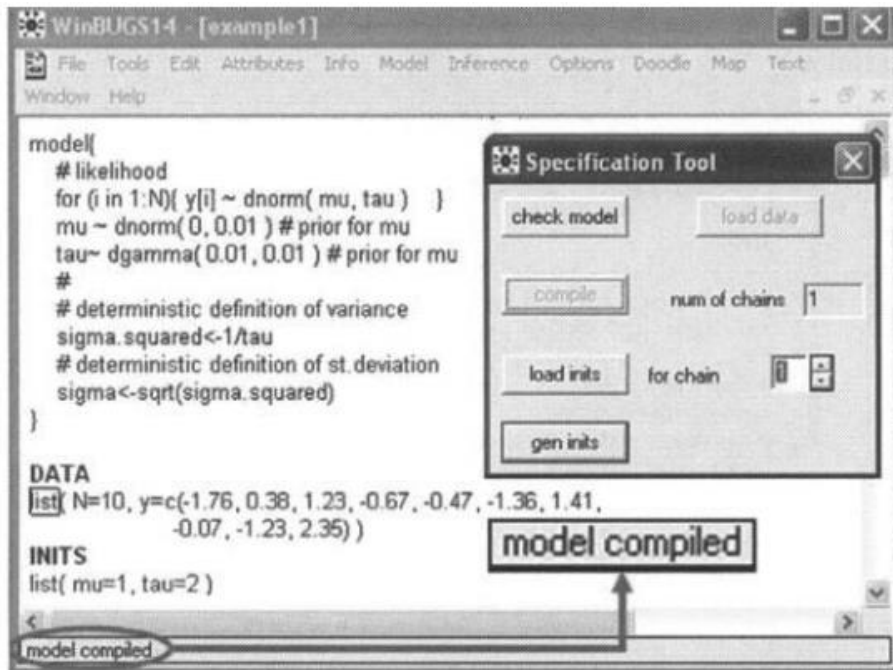


Figure 6.9 Model Compilation.

5. Set initial values. Initial values are set by following a procedure similar to the one used for data. We highlight the word list and then press the load inits box at the model specification tool.

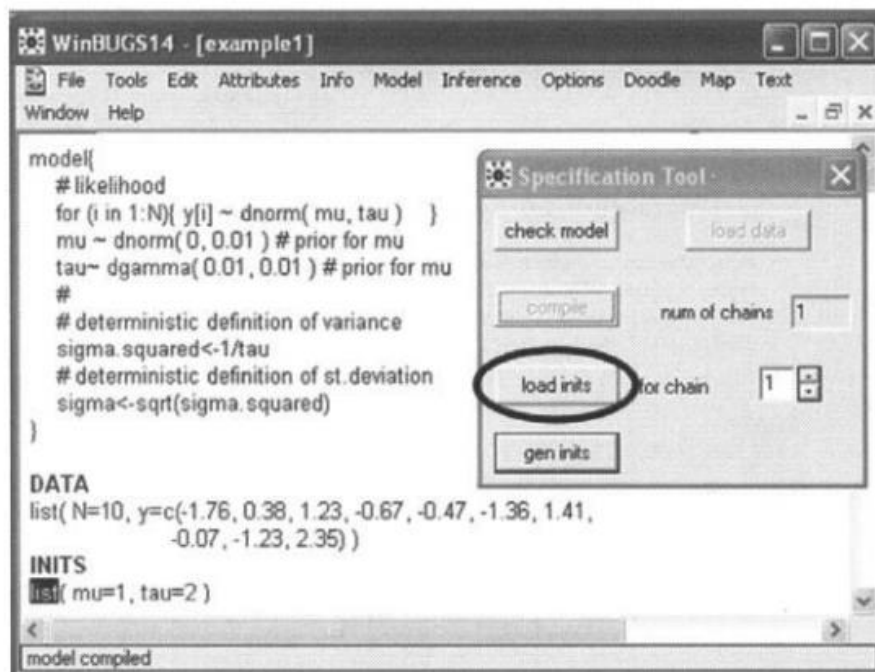


Figure 6.10 Loading Initial values.

If all initial values are set successfully, the message model is initialized will appear in the status bar; otherwise an error message will be generated in the same position.

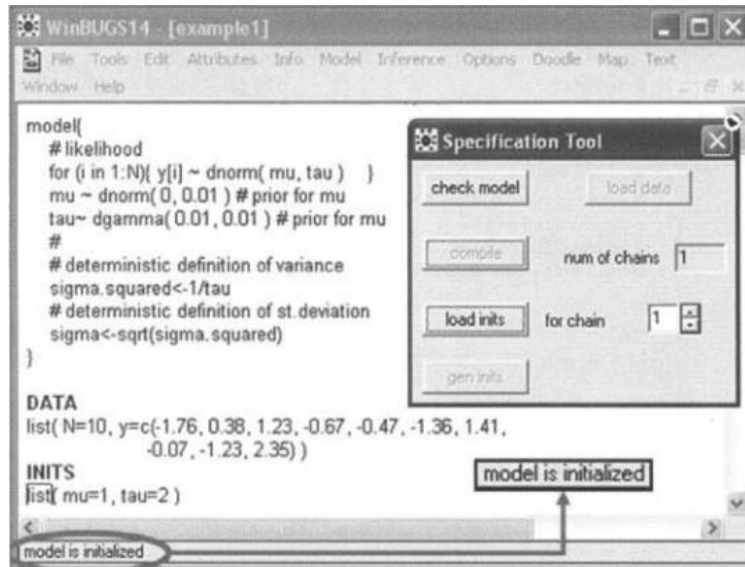


Figure 6.11 Model Initialization.

6. **Run the MCMC algorithm:** Generate random variables (burning period). Follow the path

Model> Update to open the update tool.



Figure 6.12 Updating the model.

7. Set the parameters we wish to monitor: Follow the path Inference> Samples to open the sample monitor tool.

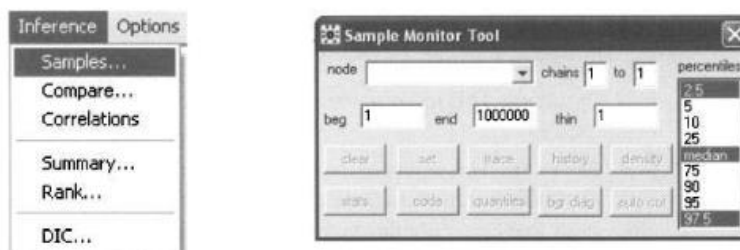


Figure 6.13 Setting number of iterations.

Write the name of the parameter that we wish to monitor in the node box and then press the set button.

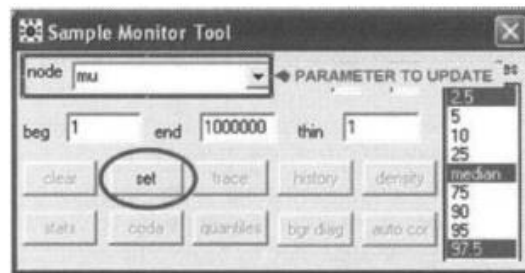


Figure 6.14 Setting parameter observer.

Using the procedure described above, we specify the parameters whose posterior distributions we wish to estimate via the MCMC generated values. The simulated values of these parameters will be now stored in order to produce a detailed posterior analysis. In WinBUGS terminology, this procedure is “setting the monitored parameters”.

8. Update the MCMC algorithm: generate and store random variables. After setting the parameter we wish to monitor, we update the MCMC sampler by repeating the procedure described in step 6. After setting the parameters of interest and generating additional random values, we can monitor the posterior distribution by extracting posterior summary statistics and plots. Analysis of the MCMC output is made via the Inference menu and mainly by the sample monitor tool.

6.4. Summary

In this chapter, the step-by-step procedure of using WinBUGS for using the Markov chain Monte Carlo (MCMC) method. Defining and compiling a model, importing the data, initializing the model, simulating the model, extracting the parameter statistics are included in this chapter. Illustration of each step is also provided for ensuring further insight of the software.

Chapter 7

Data Collection and Analysis

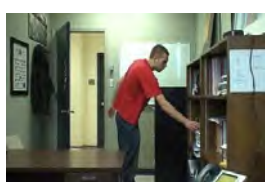
7.1. Introduction

This chapter includes a detailed description of data collection and data analysis methodology. Different models discussed in the previous chapter requires different type of data for calibration and validation. This chapter gives the complete description of the data collection methodology and the way of analysis. Particularly, the data required for calibrating the overtaking model is very sophisticated and requires complex formulation and computer algorithm. This chapter gives an extended illustration of the OD model data analysis procedure. Furthermore, the novelty of the temporal overlapping matrix and adjacency matrix are explained in this chapter. The method of extracting all the required variables for calibrating the model is also revealed in this chapter.

7.2. Data Collection

7.2.1. Data Collection for Background Modelling

As part of this thesis, six video dataset are collected from [113]: (1) ‘Office’; (2) ‘Pedestrians’; (3) ‘PETS2006’; (4) ‘Highway’; (5) ‘Boat’; and (6) ‘Blizzard’. These videos contain a mixture of mild to hard challenges such as gradual to sudden illumination variation, stop and go traffic situation, which are similar to the problems that can be addressed through M3 as per chapter 3. Among these videos, the first three have been used for model calibration; whereas, the rest have been used for validation purpose.



(a) ‘Office’



(b) ‘Pedestrians’



(c) ‘PETS2006’



(d) ‘Highway’



(e) ‘Boat’



(f) ‘Blizzard’

Figure 7.1 Traffic dataset.

7.2.2. Data Collection for Real-Time Traffic Measurement

Three different locations in Dhaka city are chosen to validate the flow measurement methodology incorporated within PARTS. These locations comprise of different roadway and traffic characteristics. Specifically, the first location is an urban signalized intersection containing heterogeneous motorized and non-motorized vehicles. Moreover, due to traffic signal, vehicles need to stop periodically causing background pollution. This problem is handled efficiently by incorporating pollution controlling parameters as illustrated in the Background Estimation Module. The next location is a 4-lane urban arterial link. It contains high speed uninterrupted motorized vehicles, where passenger cars are dominant. The video captured for this location contains sudden and gradual illumination variation. This allows analyzing the accuracy of incorporating luminance controlling parameters in background modeling. The third location is a rural highway containing high speed uninterrupted vehicles. However, in contrast to the other two locations, it comprises of mostly large bus and trucks. This imposes another challenge, i.e., occlusion problem due to the presence of large vehicles in the field of vision (FOV). Among these locations, traffic movements in the urban intersection and arterial link shows poor lane discipline.



Figure 7.2 Traffic measurement dataset.

Two hours videos have been captured in each of the locations. The mounting height of the cameras is at least 20ft and their angle is less than 45 degrees to reduce detecting the object details. For the same period, ground truth data (speed and flow) has also been collected from the video through manual post-processing. For ground truth and PARTS-based speed measurement, a strip of 88ft is chosen within the FOV. For camera calibration, distances of field lane markings are collected and provided into the User Input Module.

7.2.3. Data Collection for Crash Probability Estimation

In this study, a 2.5 hour video was captured from a two-lane undivided rural highway containing high speed uninterrupted vehicles. To avoid detailed object detection, the mounting

height of the cameras was kept at 20ft and their angle was less than 45 degrees. The video contained gradual to sudden illumination variation, vehicular shadows and small camera jitter due to wind.



Figure 7.3 Dataset for crash probability estimation.

7.4. Data Analysis

The video was analyzed following the methodology described in subsection 5.3 to detect the vehicles. Accordingly, the vehicle trajectories of the detected vehicles were extracted following the methodology described in subsection 5.4. Furthermore, vehicle classification was also done as per subsection 5.5. It was found that the video contained trajectory information of 2854 vehicles. Among these trajectories, 771 bus, 428 microbus, 485 car and 1170 freight vehicles were detected. This study location had a high crash record of 8922 crashes over the last fifteen years. Among these, bus, car, microbus, freight vehicles (truck, semi-trailer and oil tanker) accounted for 4054, 652, 747 and 3469 crashes, respectively. It represents that bus and freight vehicles are at the top in number of recorded crashes.

The overtaking decisions were obtained automatically for each trajectory considering that a driver began to think whether to overtake or not after noticing the lead vehicle. Afterwards, the driver entered the opposing traffic lane, when he/she took decision to overtake. Consequently, there should be adequate gap between vehicles of the opposing traffic during overtaking. For the specific experiment, each opposing traffic gap represented overtaking chance, which ended by the time the opposing vehicle passed by the subject vehicle, while a new chance for overtaking began. Each time, the decision for accepting gap and overtaking depended on driver. Considering this, successful overtaking attempts were codified with 1 and otherwise 0. Total

720 observations were found to be compliant with this consideration and total 82 successful overtaking maneuvers were recorded.

In OD model, eight variables were considered. The variables are: (1) Subject Vehicle Speed (V_n); (2) Lead vehicle speed (V_{n-1}); (3) Opposing vehicle speed (V_k); (4) Subject-lead vehicle spacing ($S_{n|n-1}$); (5) Subject-opposing vehicle spacing (S_{nk}); (6) Flow of the stream (q_n); (7) Density of the stream (ρ_n); and (8) Aggressiveness of the Driver (γ_n).

Tactlessly, extracted trajectories from the ‘Data collection’ (subsection 7.2.3) possessed only time-space signature of individual vehicle. To calibrate the model following ‘Model calibration procedure’, a set of data corresponding to the observed variables were estimated. In this purpose, firstly, the trajectories of different vehicles were sorted out into two different types of trajectories: (1) Trajectories in current lane; and (2) Trajectories in opposing lane. It was sorted considering equation (7.1).

$$\left. \begin{aligned} T_E^{con} &= T_i, y_i(0) < \frac{H}{2} \\ T_F^{op} &= T_i, y_i(0) \geq \frac{H}{2} \end{aligned} \right\} \quad (7.1)$$

Where,

T_i = Trajectory of the i^{th} vehicle

T_E^{con} = Trajectory of the E^{th} vehicle on the current lane

T_F^{op} = Trajectory of the F^{th} vehicle on the opposing lane

H = Height of the field of vision

$i \in N$, $N = \{1, 2, 3, \dots\}$, $|i| = |E| + |F|$

Later, each of the vehicles on the current lane was considered as subject vehicle and its corresponding front vehicles were determined. Thus, the temporal overlap matrix (T_{ij}^{overlap}) among the trajectories on current lane (T_E^{con}) was obtained from equation (7.2).

$$T^{\text{overlap}} = \begin{pmatrix} 100 & a_{12} & \cdots & a_{1N} \\ a_{21} & 100 & \cdots & a_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N1} & \cdots & a_{NN-1} & 100 \end{pmatrix}, a_{ij} = \frac{\text{Max}(t_{r_i}) - \text{Min}(t_{r_j})}{\text{Max}(t_{r_i}) - \text{Min}(t_{r_i})} \times 100 \quad (7.2)$$

From this, the Subject-Front adjacency matrix (A_{SF}) was obtained using equation (7.3).

Afterwards, Subject-Leader adjacency matrix (A_{SL}) was determined using equation (7.4).

$$A_{SF} = T^{overlap} \cdot (1 - I) \quad (7.3)$$

Where, I = Identity matrix.

$$A_{SL} = \begin{pmatrix} 0 & t_{12} & \cdots & t_{1N} \\ t_{21} & 0 & \cdots & t_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ t_{N1} & \cdots & t_{NN-1} & 0 \end{pmatrix} \cdot A_{SF} \quad (7.4)$$

Where,

$$t_{ij} = 1, \text{ if } (t_j - t_i) \leq \tau_h \\ = 0, \text{ otherwise}$$

t_i = Entry time of vehicle i within the scenario

t_j = Entry time of vehicle j within the scenario

τ_h = Threshold headway, 3 sec (9)

Similarly, the Subject-Opposing adjacency matrix (A_{SO}) was determined. With the above two matrices, the space mean speed matrix for both considered lane (V_{con}) and opposing lane (V_{op}) were determined. Additionally, Subject – Leader spacing matrix (S_{SL}) and Subject-Opposing spacing matrix (S_{SO}) were determined from the spatial signature of the trajectories. All the necessary traffic characteristics were obtained from these matrices. Particularly, vectors containing the variables V_n , V_{n-1} , V_k , $S_{n/n-1}$ and S_{nk} were obtained from these matrices using equations (7.5).

$$\left. \begin{aligned} V_n &= V_{con} \\ V_{n-1} &= A_{SL} \times V_{con} \\ V_k &= A_{SO} \times V_{OP} \\ S_{n/n-1} &= S_{SL} \times A_{SL} \\ S_{nk} &= S_{SO} \times A_{SO} \end{aligned} \right\} \quad (7.5)$$

Moreover, the flow q_n and density ρ_n were also determined from the trajectory data for time at the entry of each subject vehicle. Additionally, the aggressiveness (41) of the vehicle was determined from the following equation (7.6). The equation was calibrated using extracted

traffic flow data (q_n, ρ_n). Negative value of aggressiveness indicated aggressive driver and vice-versa.

$$\gamma_n = \frac{1}{V_n^2} (S_{n|n-1} - \tau V_n - L_n) \quad (7.6)$$

Where, τ = Perception-reaction time, (2.482 sec)

L_n = Average length of vehicle, (12.576m)

Table 7.1 Traffic and driver characteristics while overtaking

Variables	Car		Freight		Bus		Microbus	
	μ	sd	μ	sd	μ	sd	μ	sd
V_n (m/s)	27.7	21.632	20.55	19.76	22.14	4.17	25.30	9.02
V_{n-1} (m/s)	12.12	13.286	13.26	10.38	13.98	2.05	9.03	3.50
V_k (m/s)	8.14	3.6313	8.29	6.22	7.02	4.80	12.79	0.97
$S_{n n-1}$ (m)	62.33	18.997	58.35	23.67	37.16	16.15	52.94	24.25
$S_{n k}$ (m)	92.09	15.849	85.97	24.46	61.71	13.11	83.88	21.27
q_n (veh/h)	180	20.56	198	21.21	294.5	37.26	103	43.5
ρ_n (veh/mile)	27.76	6.465	33.61	15.52	62.40	56.87	38.35	19.33
γ_n	-0.071	0.005	-0.016	0.004	-0.099	0.003	-0.067	0.001

Table 7.1 shows the traffic and driver characteristics while overtaking extracted from the vehicle trajectories. The mean (μ) and standard deviation (sd) value of the considered variables are represented in this table. It revealed that car adopted highest speed while overtaking; whereas heavy vehicles adopted the lowest one. Moreover, in each case, leader and opposing vehicle had less speed than the subject vehicle. The spacing between leader and follower also differed with vehicle type. Interestingly, the highest spacing was required for car and the lowest was for bus. While overtaking, the gap required between the subject and opposing vehicle was in between 60 to 95m. Table 7.1 also illustrates that the overtaking occurs

in low traffic density. Ultimately, it can be seen that car and bus drivers were aggressive compared to the other two types of vehicles for the study location.

7.5. Summary

In this chapter, different types of data for different model is presented. Some data are pictorial and some data are traffic measurements obtained from the field as ground truth. The pictorial data are mostly used for background model calibration and overtaking model calibration. On the other hand, the traffic measurement data is collected to validate the traffic measurement module. This measurement data has also been used to see the counting performance of the detection module. Particularly, the videos from the field are converted into pictorial data and afterwards the object is detected from it. The trajectory of the detected vehicles have been extracted accordingly. Using this trajectory, the required data for overtaking modeling is obtained.

Chapter 8

Calibration and Validation

8.1 Introduction

This chapter presents the calibration and validation of the background model and the overtaking model. For calibrating background model the pictorial data from subsection 7.2.1 are used. The validation is done using the same data set. First three videos out of six are used for calibration and the rest are used validation of the background model. Both graphical optimization and constrained optimization is adopted for calibrating the background model. Field testing of this models is also included in this chapter. For validating the performance of the measurement tools the data described in subsection 7.2.2 are used. Three videos from three different places are used to check the performance of the traffic measurement tool (PARTS) in measuring both flow and speed. Ultimately, the final data set as discussed in 7.2.3 are used in calibrating the OD mode. This chapter also discusses the calibration procedure of the OD model and estimating the optimum parameter values.

8.2. Calibration and Validation of Background Model

Model M3 consists of 4 parameters. Among them, two are luminance controlling parameters (N_l, N_r) and the other two are pollution controlling parameters (δ_c, r_i). However, while traffic segmentation, binary decision needs to be made using Equation (5.27) on the basis of threshold which mainly depends on maximum threshold τ_{max} . τ_{min} is considered zero, since the higher difference does not need any suppression. Hence, only τ_{max} is considered as threshold parameter. Therefore, the segmentation needs a total of 5 parameters. Among these, four are for background modeling and one for binary decision making. These 5 parameters need to be calibrated first to make the model implementable in segmentation. In this context, we have collected several video dataset and determined the feasible region graphically for optimization. Using these feasible regions, a constrained optimization is conducted. Finally, with the calibrated parameters, both qualitative and quantitative analyses are conducted for model validation.

8.1.1. Performance Indicators

Generally, an algorithm labels a pixel as either positive or negative in a binary decision problem, where ‘positive’ and ‘negative’ respectively represent foreground pixel and

background pixel. For a given frame in a video sequence, a comparison can be drawn between the resultant image and the ground truth image. A pixel is denoted as white when it is a part of an object in the foreground, and black when it actually belongs to the background. To quantify the classification performance with respect to ground-truth, four basic measures are used, such as, (1) true positives (TP): correctly classified foreground pixels; (2) true negatives (TN): correctly classified background pixels; (3) false positives (FP): incorrectly classified foreground pixels; and (4) false negatives (FN): incorrectly classified background pixels. In addition, Precision-Recall curves are good performance indicator providing an optimistic appraisal of the classifier's performance when there is a significant skewness in the class distribution [34]. Precision-Recall curves are assembled from the following formulas:

$$\text{Precision (PR)} = \frac{TP}{TP + FP} \text{ and } \text{Recall (RE)} = \frac{TP}{TP + FN}$$

Other measures for fitness quantification, in the context of background subtraction techniques, were proposed in the literature [36-38], such as, Percentage of Correct Classification (PCC).

$$PCC = \frac{TP + TN}{TP + FN + FP + TN}$$

8.1.2. Parameter Calibration

For calibration of the parameters, optimum range is needed to be identified first. In this regard, the most feasible regions of the parameters are obtained using graphical optimization. Afterwards, using these feasible regions, a constrained optimization is conducted as follows.

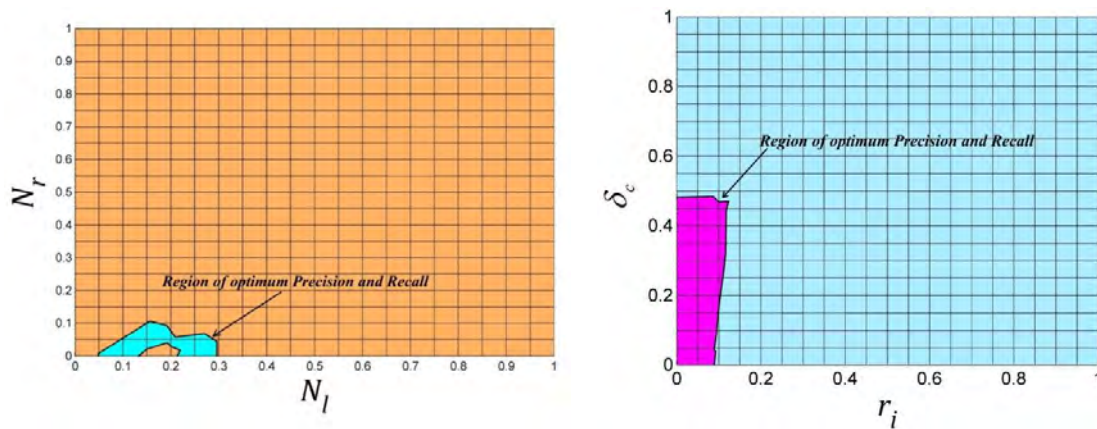
8.1.2.1. Graphical optimization

This section illustrates the methodology of the graphical optimization. It is done by plotting the indicator values against the parameter values. It is a 3-dimensional plot from which the feasible region of the corresponding parameters can be obtained. The main criterion for this optimization is the number of parameters in single plot cannot be greater than 2. Thus, the five parameters need to be breakdown into smaller groups. This breakdown is done according to the properties of parameters. Among the parameters, luminance controlling parameters (N_i and N_r) are for adopting illumination variation. Thus, these parameters are interdependent and needs to be grouped into one and optimized in same plot. In contrast, the pollution controlling parameters (δ_c and r_i) are for controlling accumulation of static object into the background model. Hence, these parameters are interdependent and needs to be grouped into one and optimized into same plot other than above. It should be noted that, illumination variation and accumulation of static object are mutually independent phenomena. However, the threshold

parameter copes up with the resulted background using both luminance and pollution controlling parameters. Thus, the threshold parameter (τ_{\max}) is dependent on those parameters. Considering this, the feasible regions of interdependent parameters are obtained initially. Using the feasible region of the interdependent parameters, the feasible region of dependent parameter is obtained afterward.

Using the first three videos (Figure 7.1(a-c)) along with their corresponding ground truth, performance indicators (Precision and Recall) are obtained for different combination of luminance controlling parameters (N_l and N_r). For convenience, normalized values of N_l and N_r are used in Figure 8.1(a) which shows the feasible region of luminance controlling parameters. The indicator values are plotted against N_l and N_r to find the feasible region. The precision and recall values are being maximized within the feasible region at the same time. The feasible region entails that lower value of both N_r and N_l are required for accurate traffic segmentation. Particularly, the lower value of N_r and N_l are needed to capture the background dynamics such as gradual and sudden illumination variation.

On the other hand, the performance indicators are also obtained for different combination of pollution controlling parameters (δ_c and r_i). For convenience, normalized values of δ_c and r_i are used in Figure 8.1(b) which shows the feasible region of pollution controlling parameters. Likewise, the indicator values are plotted against δ_c and r_i to find the feasible region. And similarly, the precision and recall values are being maximized within the feasible region at the same time. The feasible region entails that lower value of both δ_c and r_i are required for reducing pollution during background estimation. Particularly, the lower value of δ_c and r_i are needed to retard the accumulation of traffic pixels into the estimated background.



(a) Luminance controlling parameters

(b) Pollution controlling parameters

Figure 8.1 Parameters optimization.

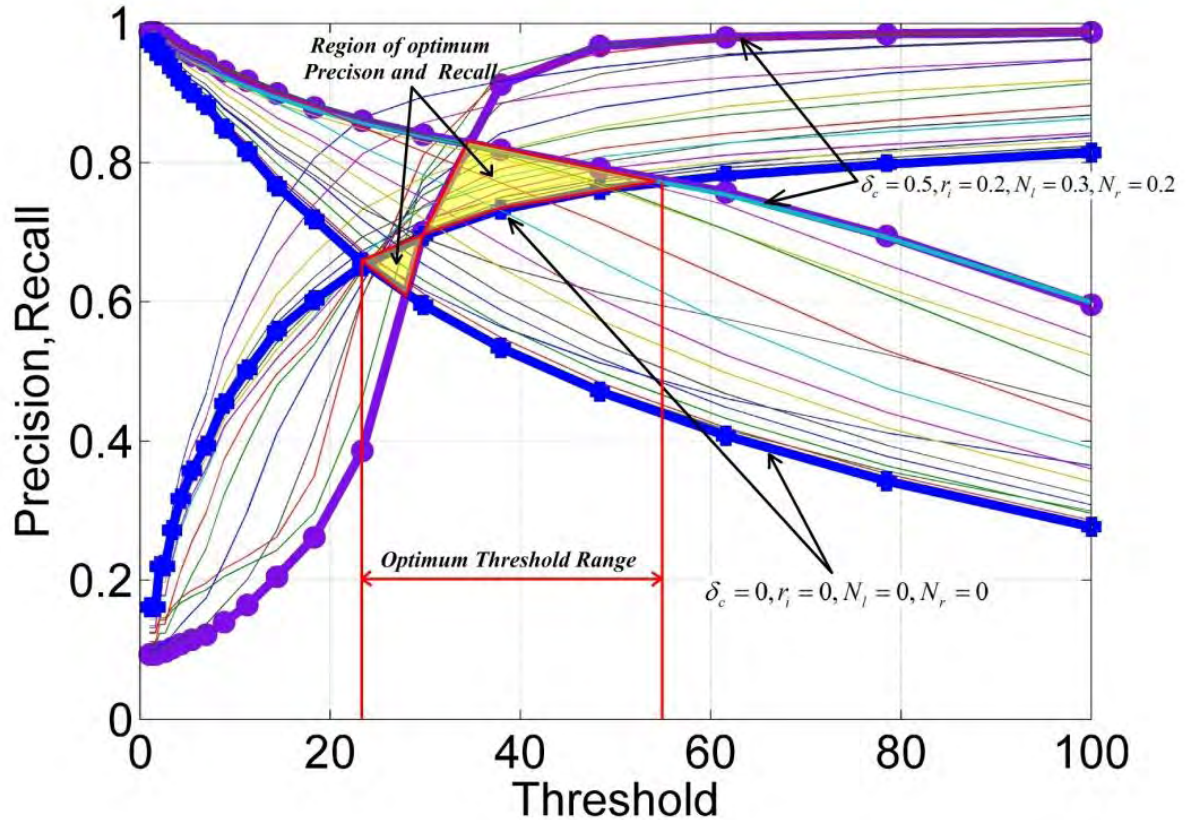


Figure 8.2 Optimum threshold for optimum luminance and pollution controlling parameters.

The region of optimum precision and recall for threshold parameter can be obtained by using various combinations of the boundary values from these two figures. Using the boundary values of Figure 8.1(a) and Figure 8.1(b), 24 parameter sets have been established having values of $\delta_c = \{0, 0.5\}$, $r_i = \{0, 0.2\}$, $N_l = \{0.1, 0.2, 0.3\}$, $N_r = \{0.1, 0.2\}$ covering all possible combinations.

Ultimately, using these values, graphical optimization to find the feasible region for the threshold is conducted. This optimization has been done by precision recall analysis. In total 48 curves (24 precision-threshold and 24 recall-threshold curves) have been plotted in Figure 8.2 to obtain the feasible region where lies the optimum threshold value. The upper and lower boundary of the parameter set has been marked with darker thick curves. The figure shows that the lower boundaries cause minimum precision and recall value; whereas the upper boundary causes maximum precision and recall value. On the other hand, if the model parameters are changed as per the boundary values which are maximum, precision and recall both values maximizes. The shaded region in Figure 8.2 entails the optimum range of threshold parameter which is in between 24 and 56. Moreover, it has two triangular regions within it. Although, the first one is smaller and one of its corner defines the lowest value for the threshold, the second

one is biggest and one of its corner defines the maximum threshold can be used for vehicle segmentation. If very small threshold value is used, it makes the detection anomalous by considering noises as foreground by increasing the false positive rate. On the other hand, very higher value of threshold makes the detection anomalous by suppressing the necessary objects. Thus it increases false negative rate. Finally, the range of optimum threshold is obtained.

8.1.2.2. Constrained optimization

The feasible regions obtained from the previous section entails the upper and lower boundary of the constrained optimization. This constrained optimization is done to find the precise value of the parameters. To find the optimum parameters, Equation (8.1) is used,

$$\min \frac{1}{N} \sum_{i \in n} (PR)^{-1} (RE)^{-1} \quad (8.1)$$

Such that,

$$0 \leq \delta_c \leq 0.5$$

$$0 \leq r_i \leq 0.2$$

$$0.1 \leq N_l \leq 0.3$$

$$0.1 \leq N_r \leq 0.2$$

$$24 \leq \tau \leq 56$$

Optimizing Equation (8.1), the calibrated values has been obtained. The value: $N_l = 425$, $N_r = 255$, $\delta_c = 120$, $r_i = 0.1$ and $\tau_{\max} = 25.56$. Apart from the optimized values, it is necessary to evaluate the sensitivity of the parameters deviating from calibrated values. The sensitivity is dependent of the rate of change of the performance measure with changing parameter values. To observe the sensitivity, Figure 8.3 is generated using parameters. Both precision and recall is considered as performance indicator. It can be seen from the figure that the most sensitive parameter of the model is the threshold parameter τ_{\max} . It concludes that while calibrating the threshold parameter have to be chosen very carefully.

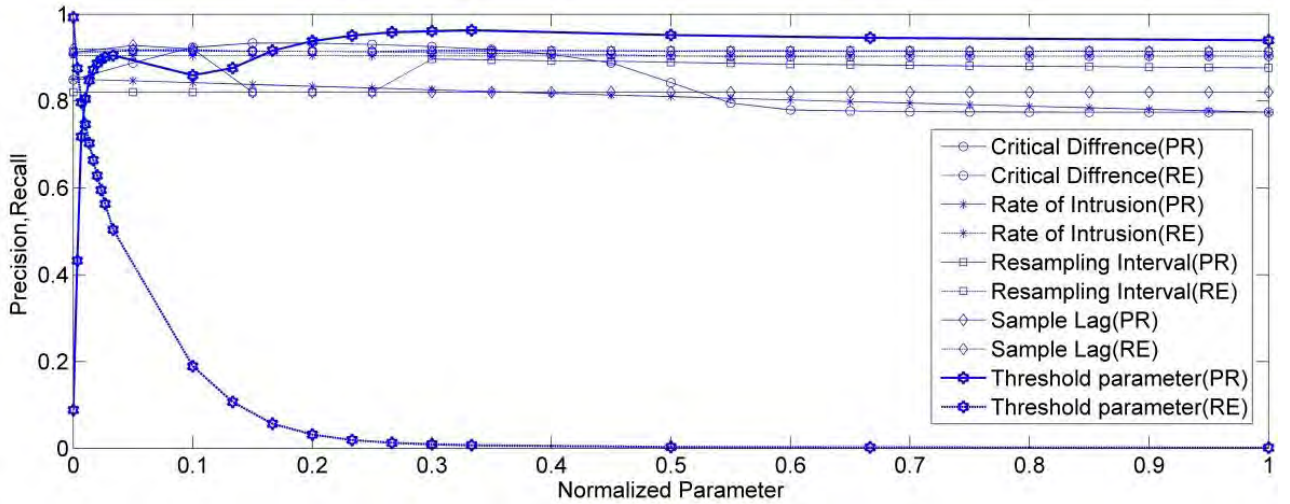








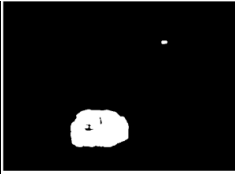


Figure 8.3 The effect of normalized parameter on performance indicator considering $\delta_c \in [0, 255]$, $r_i \in [0, 1]$, $N_r = [0, 1700]$, $N_l = [0, 1700]$ and $\tau_{\max} = [0, 3000]$.

8.1.3. Validation

Using calibrated parameter values, the binary images of the validation dataset have been obtained and presented in Table 8.1. The detected objects have been marked white pixels. The table shows the qualitative analysis result for a single frame and quantitative analysis result shows the average performance indicator of all the frames for a certain video.

Table 8.1 Foreground segmentation using optimized parameters.

	Qualitative analysis			Quantitative analysis		
	Actual Image	Ground Truth	Model Output	Precision	Recall	PCC
Highway				0.9269	0.9575	0.9891
Boats				0.8335	0.9928	0.9567
Blizzard				0.9651	0.9275	0.9820

The qualitative analysis consists of the actual, ground truth and model output binary image. The model output is generated by using M3 and the calibrated parameters. It shows that, model M3 outperforms in vehicle detection in all the situation. However, in ‘Boat’ sequence there are some false positive pixels. It is due to the river dynamics which is not properly captured by the model. On the other hand, ‘Blizzard’ shows some false negative pixels. Camouflage is the main reason behind this error. The quantitative analysis shows that it has achieved 90% precision, 96% recall and 98% PCC value. It also shows that the precision value decreases due to increase in false positive pixels for ‘Boats’ video and recall value decreases due to false negative pixels in ‘Blizzard’ video.

8.2. Field Testing of Model M3

Three different locations in Dhaka city are chosen to assess the performance of model M3. The data collection time at each location was chosen to cover the peak and off-peak period characteristics of the traffic stream. These locations comprise of different roadway and traffic characteristics. Specifically, the first location is an urban signalized intersection containing heterogeneous motorized and non-motorized vehicles. Moreover, due to traffic signal, vehicles need to stop periodically causing background pollution. This problem is handled efficiently by incorporating pollution controlling parameters as illustrated in the M3. The next location is a 4-lane urban arterial link. It contains high speed uninterrupted motorized vehicles only, where passenger cars are dominant. The video captured for this location contains sudden and gradual illumination variation. This allows analyzing the accuracy of incorporating luminance controlling parameters in background modeling. The third location is a two-lane rural highway containing high speed uninterrupted vehicles. However, in contrast to the other two locations, it comprises of mostly large bus and trucks. This imposes another challenge, i.e., occlusion problem due to the presence of large vehicles in the field of vision (FOV). Among these locations, traffic movements in the urban intersection and arterial link shows poor lane discipline. The mounting height of the cameras is at least 20ft and their angle is less than 45 degrees to reduce detecting the object details. For the same period, ground truth count data has also been collected from the video through manual post-processing.

Using these collected videos, model M3 has been utilized to produce binary image incorporating the calibrated parameter values optimized in subsection 8.1.2.2. Furthermore, disk shaped morphological operations (dilation and erosion) are used to improve the quality of the binary image. Afterwards, blob analysis has been applied on the improved binary image to

obtain the estimated count. Using this estimated count, three measure of performance such as Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Average Accuracy of Object Count (AAOC) is determined applying the following formulae.

$$MAE = \frac{1}{N} \sum_i^N |N_i^G - N_i^{M3}|$$

$$MAPE = \frac{100}{N} \sum_i^N \frac{|N_i^G - N_i^{M3}|}{N_i^G}$$

$$AAOC = \frac{1}{N} \sum_i^N \frac{\text{Min}(N_i^G, N_i^{M3})}{\frac{N_i^G + N_i^{M3}}{2}}$$

Where,

N_i^G = Ground truth count for *ith* sample

N_i^{M3} = Estimated count obtained using M3 for *ith* sample

N = Number of total sample

$N_i^G + N_i^{M3} > 0$

Small values of MAE and MAPE and big values of AAOC are considered as better result.

Figure 8.4 shows that for all dataset, result obtained using M3 is in harmony with the ground truth. The MAE values are 2.48, 2.01, 9.77 vehicle/minute and MAPE values are 13%, 18% and 25% for urban arterial, rural highway and urban intersection dataset respectively. Both MAE and MAPE value are minimum for urban arterial dataset and maximum for urban intersection dataset.

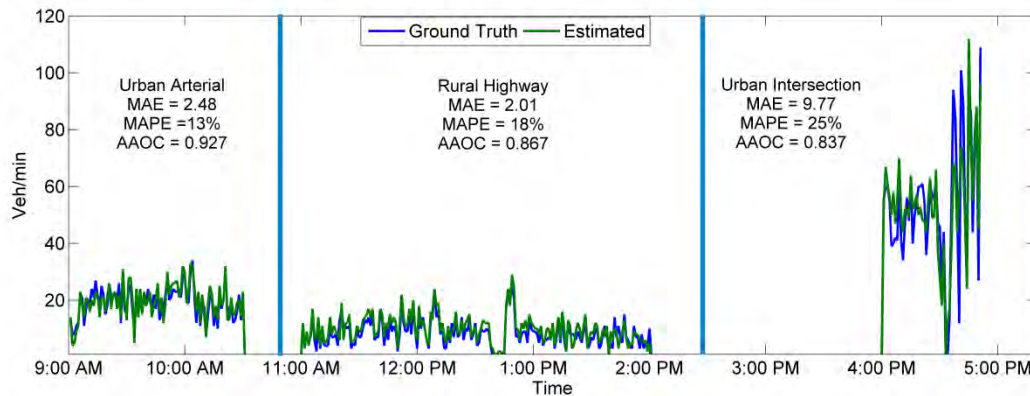


Figure 8.4 Field testing of M3 in different locations.

The main reason behind it is the presence of large and small vehicles in the urban intersection dataset. Specifically, small vehicles like motorcycles and bicycles are eroded by the morphological operations, which ultimately reduces the estimated count. Whereas, the other two locations contain fewer number of small vehicles. The analysis also shows that the model achieves 93% AAOC in morning peak period, which represent its robustness in traffic detection.

8.3. Field Testing of PARTS

Global parameters set is obtained by precision-recall analysis over different videos available in the Change Detection Challenge dataset (74). Specifically, $N_i=100$, $\delta_c=0.1$, $r_i=0.01$, $N_l=0.05$, $N_r=0.2$, $\tau_{\min}=1$, $\tau_{\max}=25$, $\tau_s=0.5$ yield maximum precision-recall values and are used as required parameters for different modules of PARTS. The above dataset contains all the challenges that the vehicle detection methodology of this paper addresses and provides ground truth binary images necessary for precision-recall analysis. Note that, the above data set has also been used to estimate optimal parameters required for vehicle detection methodology as proposed in (114, 115, 116, 117 and 118).

In addition to the above parameters, two segments for speed measurement is considered in the User Input module (see Figure 4.3(b)). Figure 8.5 shows the performance of PARTS for measuring flow and speed at different locations considered in this study. In that figure, ± 10 vehicles and ± 3 km/h error lines are established for vehicle count and speed validation respectively.

Figures 8.5 (a), (c) and (e) show qualitative evaluation of PARTS-based flow measurement and ground truth. These figures illustrate that most of the points are within the area enclosed by the error lines. However, some data points fall outside the above area representing differences larger than 10 vehicles. Among them, Figure 8.5(a) shows more points outside the area compared to the other two. In contrast, Figures 8.5 (b), (d) and (f) show the qualitative evaluation of the PARTS-based speed measurement and ground truth. These figures illustrate that nearly all the points are within the area enclosed by the error lines. This represents the closeness of the PARTS-based speed measurement to the ground truth differing by less than 3 km/h.

Furthermore, quantitative analysis, i.e., two tailed t- test is performed to confirm whether there is any significant difference between the ground truth and PARTS-based measurements. In this context, the null hypothesis assumes that the mean of ground truth and PARTS-based measurement is same as presented in Table 8.2 (a) and (b). Both the tables show that the t-values for PARTS-based speed and flow measurements in all the locations are less than t-critical value at 95% confidence level. It results in accepting the null hypothesis. This in turn indicates that the differences in the mean of ground truth and PARTS-based measurements are not significant.

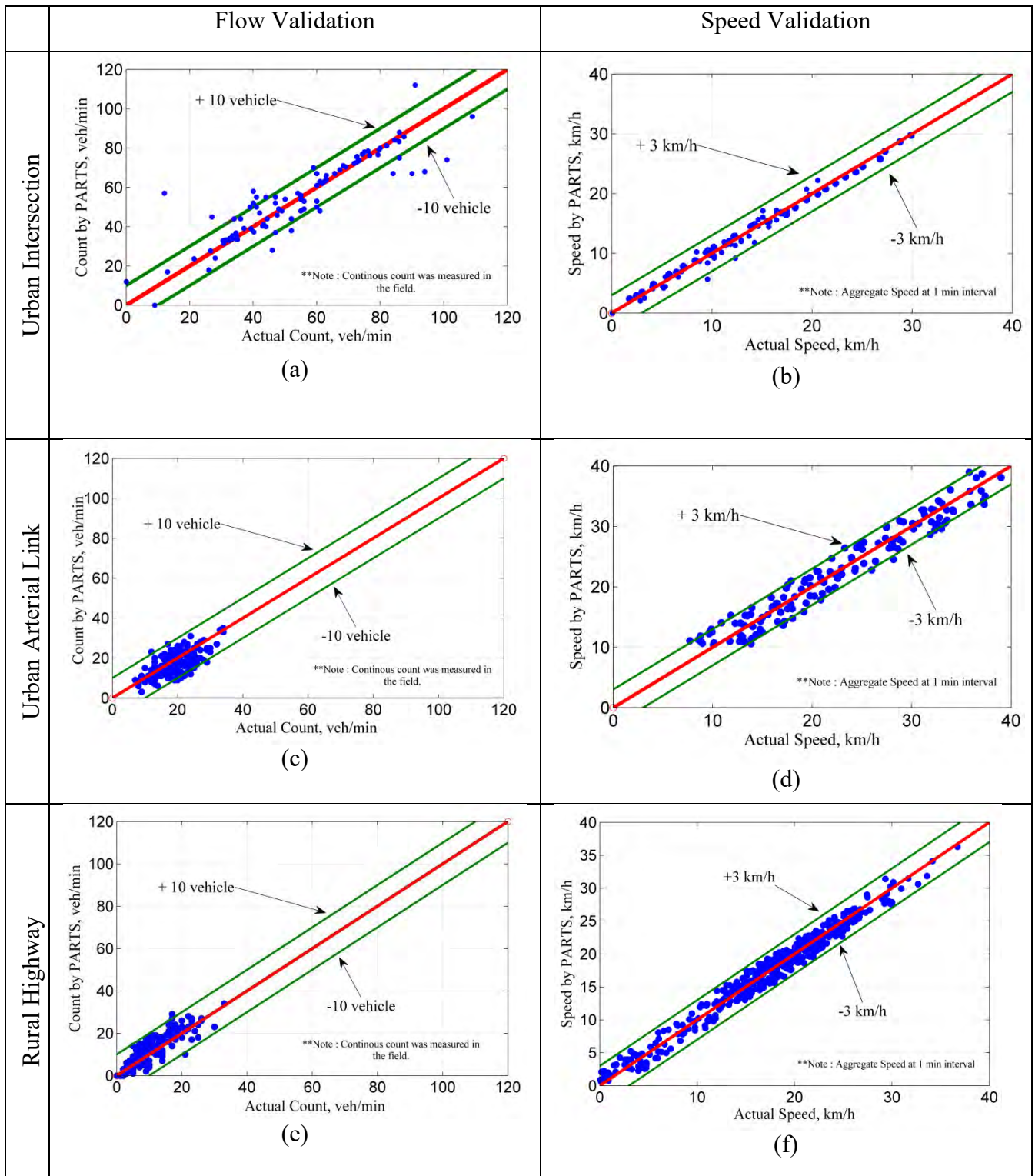


Figure 8.5 Performance evaluation of PARTS-based flow and speed measurement at different locations; (a)-(b) Urban intersection; (c)-(d) Urban arterial link; and (e)-(f) Rural highway.

Table 8.2(a) and (b) show that Pearson Correlations for the three study locations are 0.89, 0.96, and 0.83 in measuring flow; whereas 0.97, 0.97, 0.98 in measuring speed respectively. Hence, graphical and statistical analyses show that PARTS produces better results for the rural highway and urban arterial link compared to the urban intersection. The main reason behind it

is the presence of large and small vehicles in the intersection. Specifically, small vehicles like motorcycles and bicycles are eroded by the morphological operations, which ultimately produce small inflow waves in Flow Measurement Module. Small inflow waves are rounded towards zero for obtaining integer number of vehicles. Whereas, the other two locations contain fewer number of small vehicles. Interestingly, the analyses show that PARTS provide similar accuracy in speed measurement regardless of the locations. Speed measurement only considers aggregated differential centroid movement; hence, error due to erosion of small vehicles is minimized.

Table 8.2 Statistical evaluation of PARTS-based measurement with respect to ground truth.

<i>(a) Flow (vehicle/min)</i>						
	<i>Urban Intersection</i>		<i>Urban Arterial Link</i>		<i>Rural Highway</i>	
	<i>PARTS</i>	<i>Ground truth</i>	<i>PARTS</i>	<i>Ground truth</i>	<i>PARTS</i>	<i>Ground truth</i>
Mean	53.059	53.019	19.235	18.977	10.103	10.517
Variance	387.256	523.539	30.136	31.022	22.396	25.903
Observations	120	120	120	120	120	120
Pearson Correlation	0.827		0.955		0.888	
Hypothesized Mean Difference	0		0		0	
df	119		119		119	
t	0.021		1.479		-1.649	
P(T<=t) one-tail	0.491		0.071		0.051	
t Critical one-tail	1.657		1.657		1.657	
P(T<=t) two-tail	0.982		0.142		0.102	
t Critical two-tail	1.980		1.980		1.980	

<i>(b) Speed(Km/h)</i>						
	<i>Urban Intersection</i>		<i>Urban Arterial Link</i>		<i>Rural Highway</i>	
	<i>PARTS</i>	<i>Ground truth</i>	<i>PARTS</i>	<i>Ground truth</i>	<i>PARTS</i>	<i>Ground truth</i>
Mean	10.590	10.332	22.351	22.441	24.740	24.841
Variance	27.873	30.212	62.870	68.041	17.233	18.844
Observations	120	120	120	120	120	120
Pearson Correlation	0.980		0.971		0.970	
Hypothesized Mean Difference	0		0		0	
df	119		119		119	
t	1.720		-0.510		-1.246	
P(T<=t) one-tail	0.045		0.305		0.107	
t Critical one-tail	1.657		1.657		1.657	
P(T<=t) two-tail	0.091		0.610		0.214	
t Critical two-tail	1.980		1.980		1.980	

8.4. Model Calibration Procedure

The parameter values in equation (5.18) should be known prior to the crash estimation. Thus, calibration of the parameters is required. The value of l depends on the type of overtaking and can be determined from equation (8.2),

$$I = \varepsilon \exp \left(-2 \ln \frac{1}{\varepsilon} \left[1 - \text{Max} \left(\left(\frac{\exp \left(\sum_{i \in \mathbb{Z}}^n \beta_{io} X_{io}^{\omega_{io}} + \beta_0 \right)}{1 + \exp \left(\sum_{i \in \mathbb{Z}}^n \beta_{io} X_{io}^{\omega_{io}} + \beta_0 \right)} \right)_{n-1}, \left(\frac{\exp \left(\sum_{i \in \mathbb{Z}}^n \beta_{io} X_{io}^{\omega_{io}} + \beta_0 \right)}{1 + \exp \left(\sum_{i \in \mathbb{Z}}^n \beta_{io} X_{io}^{\omega_{io}} + \beta_0 \right)} \right)_{n-2}, \dots \right] \right) \quad (8.2)$$

The value of W can be obtained from the following equation,

$$w = e^{-\frac{N}{N_s}} \quad (8.3)$$

Where,

N = Number of crashes occurred within a certain period on the considered road.

N_s = Number of crashes occurred within the same period on the highest accident-prone road.

However, the calibration of the OD model parameters (β, ω) needs different procedure. The model parameters $(\theta | \theta \in \{\beta, \omega\})$ are calibrated using Metropolis-Hastings algorithm. The algorithm is summarized by the following steps:

1. Set initials values $\theta^{(0)}$
2. For $t = 1, \dots, T$, repeat the following steps
 - a. Set $\theta = \theta^{(t-1)}$
 - b. For $j = 1, \dots, d$
 - i. Generate new candidate parameter values from a proposal distribution.
 - ii. Calculate $\alpha = \min \left(1, \frac{f(\theta'_j | \theta_j, y) q(\theta_j | \theta'_j, \theta_j)}{f(\theta_j | \theta_j, y) q(\theta'_j | \theta_j, \theta_j)} \right)$

Where, θ_j is the vector θ excluding its j^{th} component θ_j

 - iii. Update $\theta_j = \theta'_j$ with probability α
- c. Set $\theta^{(t)} = \theta$

When the iterations are completed, the calibrated parameters are obtained.

8.5. Summary

In this chapter, calibration and validation of all the models using different data set are presented. Calibration of the background model is done using both graphical and constrained optimization of the parameters. However, the boundary values are calculated from the graphical calibration. Then the background model is tested for object detection and taking traffic measurements. The field testing of PARTS is also included in this chapter. Ultimately, this chapter presents the model calibration procedure using Metropolis-Hastings algorithm.

Chapter 9

Results and Discussion

9.1. Introduction

This chapter includes a detailed analysis over the compiled data. Initially the correlation among the variables are revealed. Afterwards, the data has been fitted into newly proposed model and the performance of the models have been compared. Ultimately, this chapter ends with providing nomographs for estimating the crash probability of an overtaking vehicle.

9.2. Correlation among the parameters

The correlation matrix of the concerned parameters is presented in Table 9.1. Table 9.1 shows the interrelation among the variables. The absolute correlation value which are greater than 0.1 are marked with bold font. This analysis shows that leader vehicle speed invokes the subject vehicle to accelerate. Leader-subject vehicle spacing also affects the opposing-subject vehicle spacing. Ultimately, the table shows that overtaking decision is correlated with almost all the vehicles except the opposing vehicle spacing.

Table 9.1 Correlation among the considered variable

	V_n (m/s)	V_{n-1} (m/s)	V_k (m/s)	$S_{n n-1}$ (m)	$S_{n k}$ (m)	q_n (veh/h)	ρ_n (veh/mile)	γ_n	OD
V_n (m/s)	1.000								
V_{n-1} (m/s)	0.179	1.000							
V_k (m/s)	-0.055	-0.010	1.000						
$S_{n n-1}$ (m)	0.019	0.069	0.006	1.000					
$S_{n k}$ (m)	0.001	0.051	-0.097	0.901	1.000				
q_n (veh/h)	-0.046	-0.112	0.087	-0.064	-0.046	1.000			
ρ_n (veh/mile)	0.086	-0.097	-0.061	-0.607	-0.525	0.045	1.000		
γ_n	-0.123	-0.073	0.037	0.488	0.438	-0.039	-0.307	1.000	
OD	0.207	-0.143	-0.055	0.262	0.260	-0.185	-0.202	-0.012	1.000

Table 9.2 shows the calibrated values of the parameters for all vehicles using Metropolis-Hastings algorithm. The results of the model showed very interesting and significant conclusions. It was found that the increase in subject vehicle speed also increased the probability of overtaking due to increasing speed difference with the leading vehicle. Conversely, if the speed of the leading vehicle speed increased, the probability of overtaking decreased. Because, increase in speed of the leading vehicle caused more difficulty to complete the overtaking maneuver. Thus, the overtaking probability increased with the speed difference between the subject vehicle speed and lead vehicle speed. It also comprised with the findings in (18). On the contrary, the probability of overtaking decreased for increasing speed of the opposing vehicle. Higher speed of the opposing vehicle caused the subject-opposing gap to decrease more quickly making overtaking difficult. Moreover, the higher the spacing between the leading and subject vehicle, the lower the overtaking tendency could be found. Studies (18, 21) also shown same tendency, when following gap decreased. It revealed that the subject vehicle usually overtook, when it came closer to leading vehicle. Oppositely, if the spacing between subject and opposing vehicle increased the overtaking probability increased, as larger passing gaps increased the chances of overtaking. With the increment of traffic density, the probability of overtake decreased. This is because of larger passing gaps that were available in lower traffic volumes and consequently, the overtaking chances were larger. Finally, increase in aggressiveness also increased the probability of overtaking.

The two-tailed t-stat of the coefficients revealed that all the variables were significant at 95% confidence interval except the constant term, as those were greater than the t-critical (1.96) value. Since the constant term was insignificant, it indicated that the considered variables were well enough to describe the overtaking phenomena. Thus, the effects of the unobserved variables were very small.

In Table 3, the variables are also ranked on the basis of their magnitude of the exponent. The higher the exponent, the higher the effect of the variable. If the exponent of a variable was close to zero, the change in variable would affect the overtaking probability slightly. On the other hand, if the value was away from zero, the effect of the variable increased. It could be seen that the subject vehicle speed and subject-opposing spacing had the higher exponent value. It demonstrated that they have greater effect than the other variables considered in this study. In contrast, the opposing vehicle speed had very low impact on the model. It was due to the fact

that the driver could not accurately estimate the opposing vehicle speed using his/her vision only.

Table 9.2 Values of the calibrated parameters (β, ω).

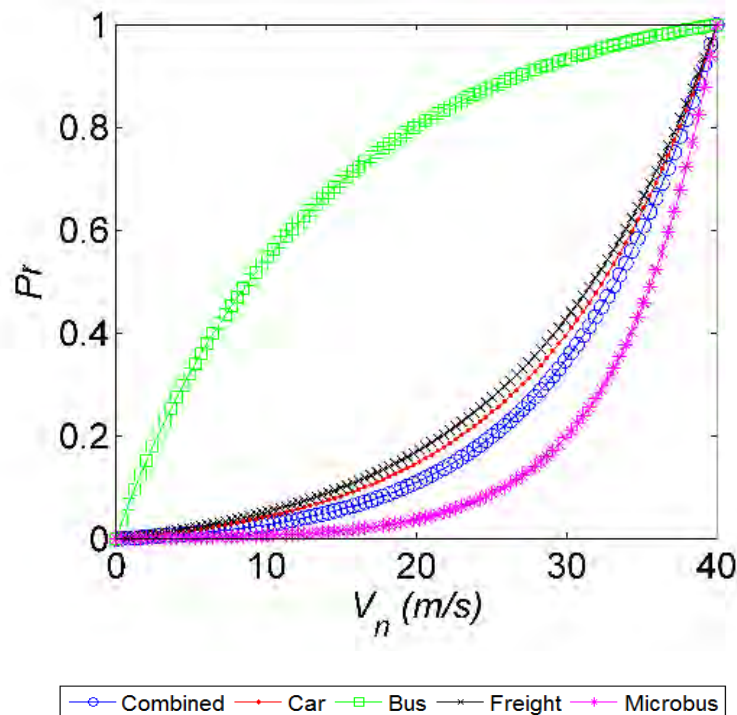
Variables	β			ω			
	μ	sd	t-stat	μ	sd	t-stat	Rank
Const.	-0.820	5.599	-1.562	1			
V_n	0.122	1.101	1.958	0.95	0.072	230.31	2
V_{n-1}	-0.313	2.474	-2.214	0.8812	0.079	194.72	3
V_k	-0.011	0.0963	1.998	0.189	0.163	20.17	8
$S_{n n-1}$	-0.527	1.404	-6.570	0.7145	0.126	98.80	5
$S_{n k}$	0.046	0.1631	5.036	0.9667	0.290	58.21	1
q_n	-0.144	0.0699	-36.184	0.313	0.166	32.97	6
ρ_n	-8.572	6.142	-24.415	0.256	0.207	21.56	7
γ_n	39.271	17.06	40.268	0.7259	0.069	183.40	4
$\log L$	-43.031						
ρ^2	0.662						

Figure 9.1 shows the overtaking probability for different variables and different vehicle types. It illustrated the change in overtaking probability with the increase in different explanatory variables. The probability for each vehicle types was plotted. It revealed that the curves were following the sign of the coefficients presented in Table 9.2. The interesting finding was that among different types of vehicle, bus possessed greater potential of overtaking. Furthermore, bus and freight vehicles were top in terms of aggressiveness. This information explained behind the highest accidents records for bus and freight vehicles in the study location.

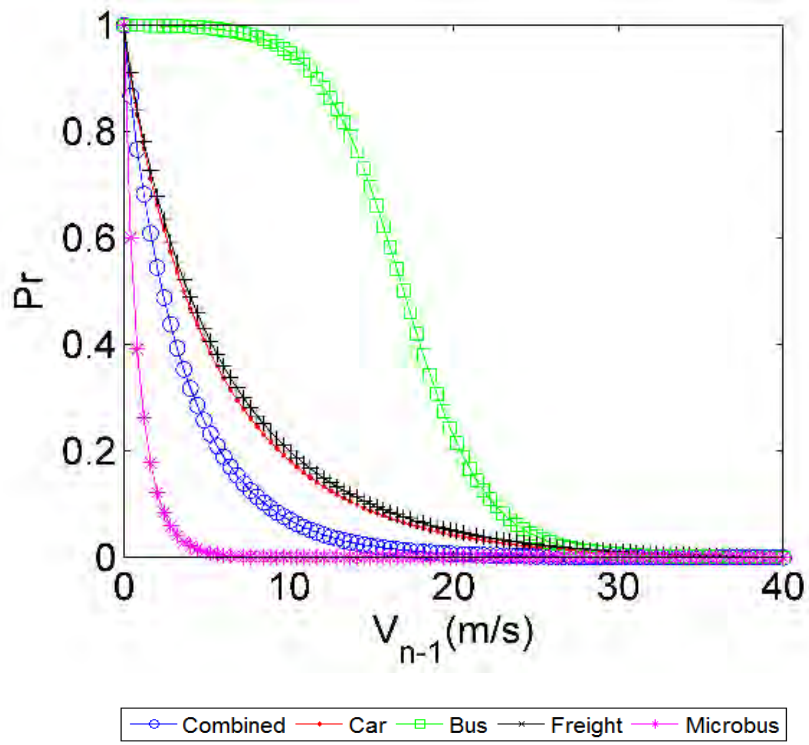
Furthermore, trajectories of different vehicles types were chosen to analyze their individual crash probability. The value of ε was considered to be unity, which balanced the importance

on the overtaking probability of the front vehicles. Value greater than unity emphasized on the probability of the front vehicles and value less than unity underestimated it. Using these trajectories and incorporating equation (5.18), the crash probability for different vehicle types were estimated. The values of maximum crash probability for different vehicles were 0.57, 0.43, 0.35 and 0.25 for bus, freight, car and microbus, respectively, which matched with the accident records presented in ‘Data collection’ section.

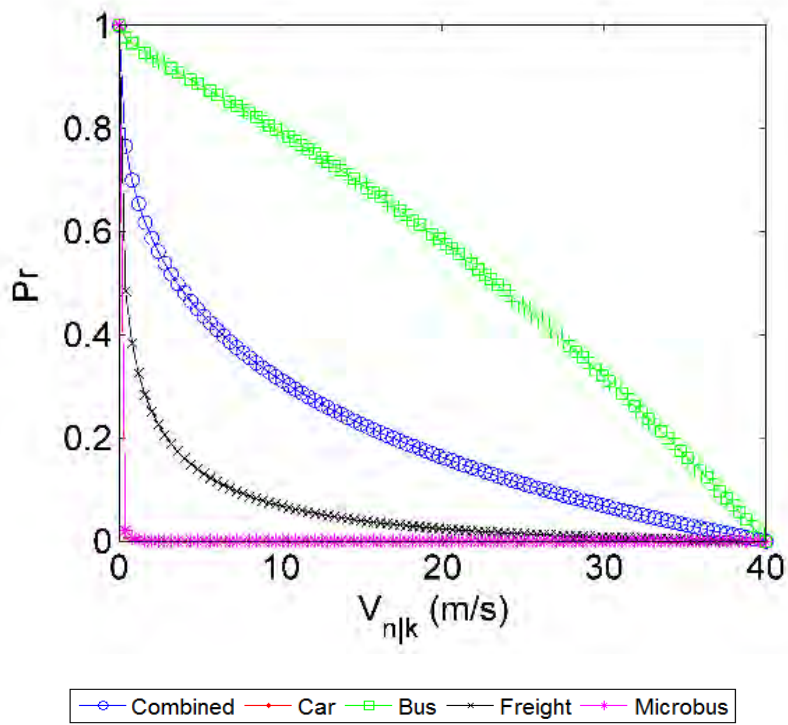
For greater understanding, the change in crash probability with time is illustrated in Figure 9.1(i). It showed that during pre-overtaking period, the crash probability was low, as the TTC was not activated yet. The left regime was showing the crash probability if and only if the vehicle overtook. However, in the mid regime, when the vehicle overtook, the TTC activated and it was clear that the crash probability increased with the shortening of TTC. Ultimately, the vehicle came back to its destined lane, when the overtaking maneuver was finished and the probability decreased. It should be noted that the vehicle came back to the previous lane keeping a safe distance between the overtaking vehicles. Nearer than this safe distance, further overtaking was difficult. Additionally, the gap between the subject and opposing vehicle decreased. According to Table 9.2, the probability of overtaking also decreased as the gap decreased. Thus, there was very low chance to overtake.



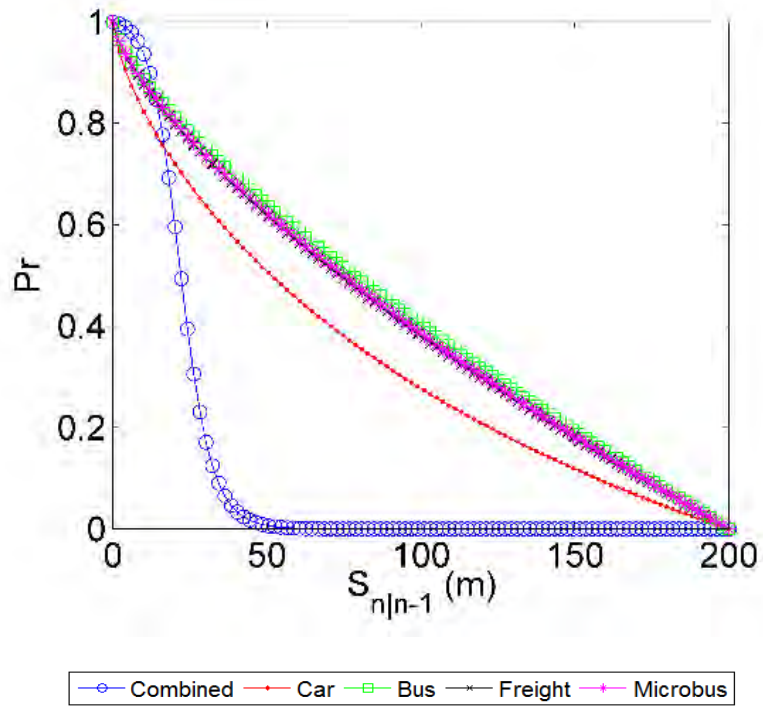
(a) Subject Vehicle



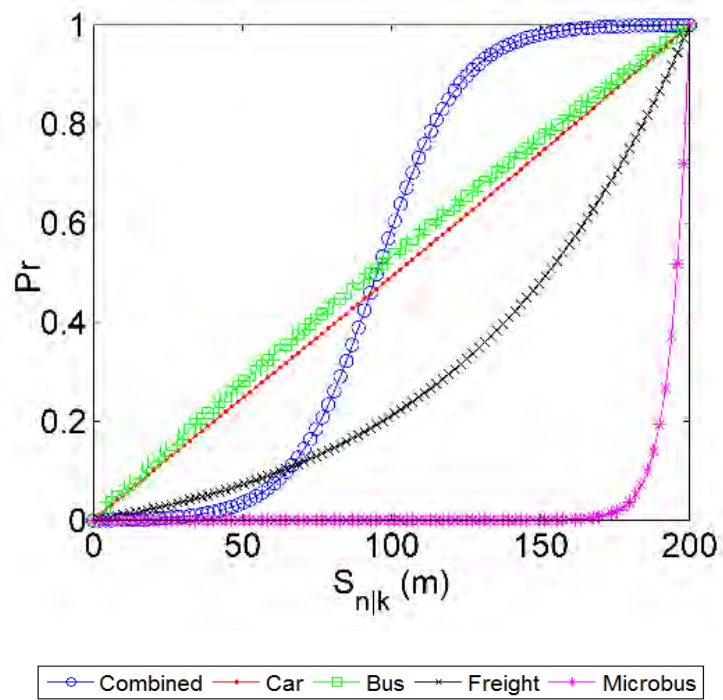
(b) Leading vehicle



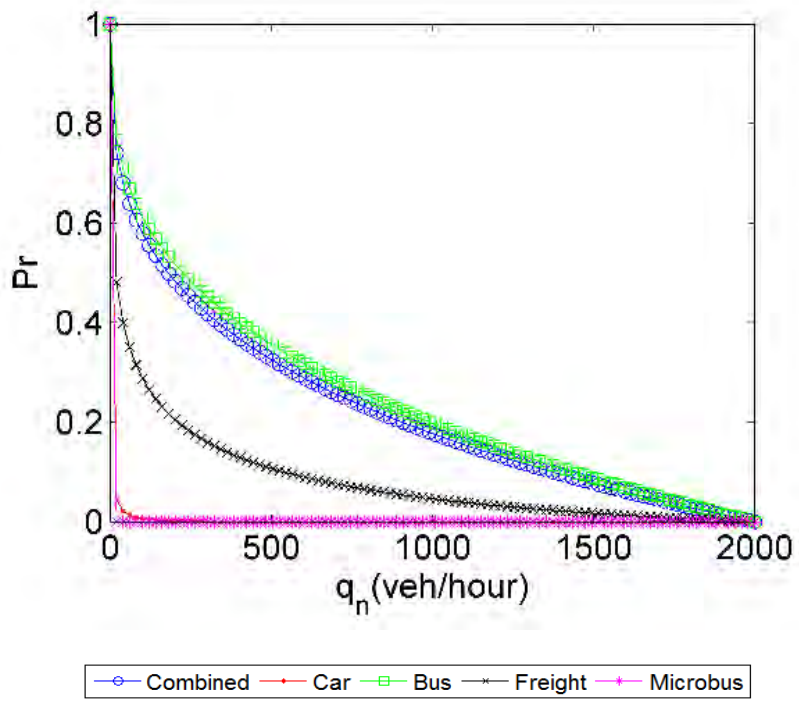
(c) Opposing Vehicle



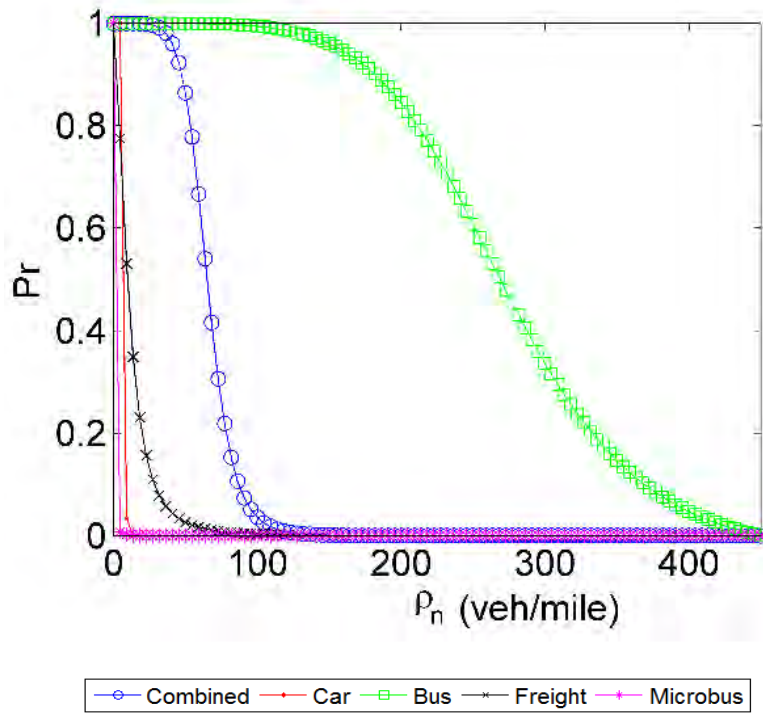
(d) Subject-Leading spacing



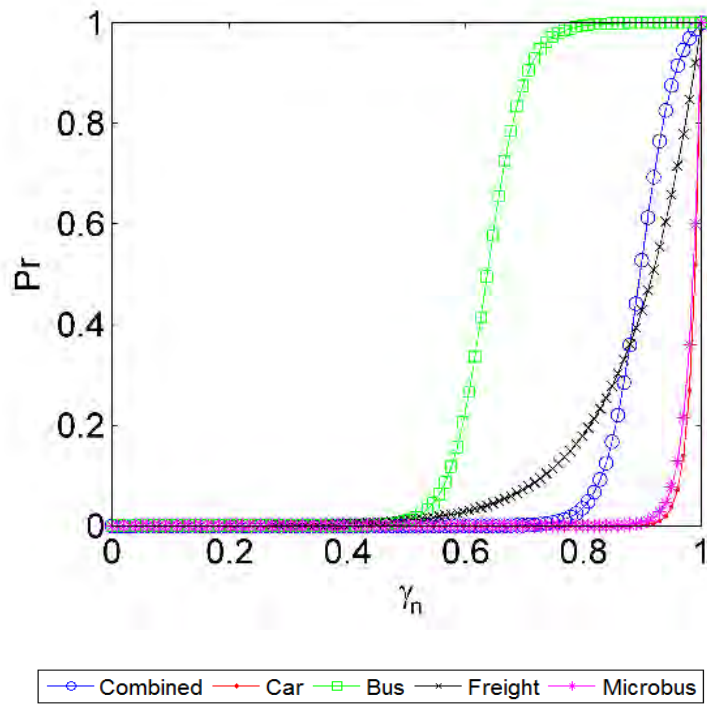
(e) Subject-Opposing spacing



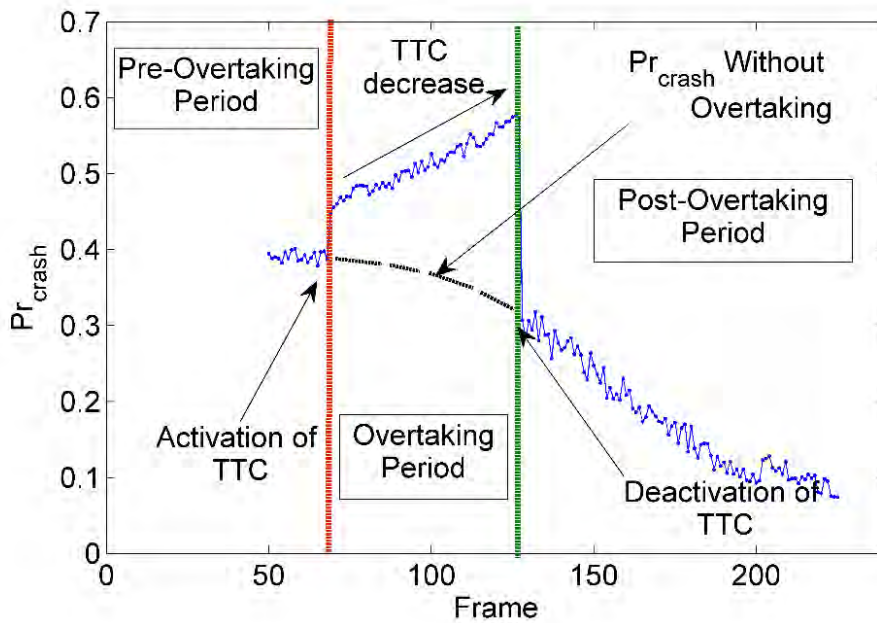
(f) Flow



(g) Density



(h) Aggressiveness



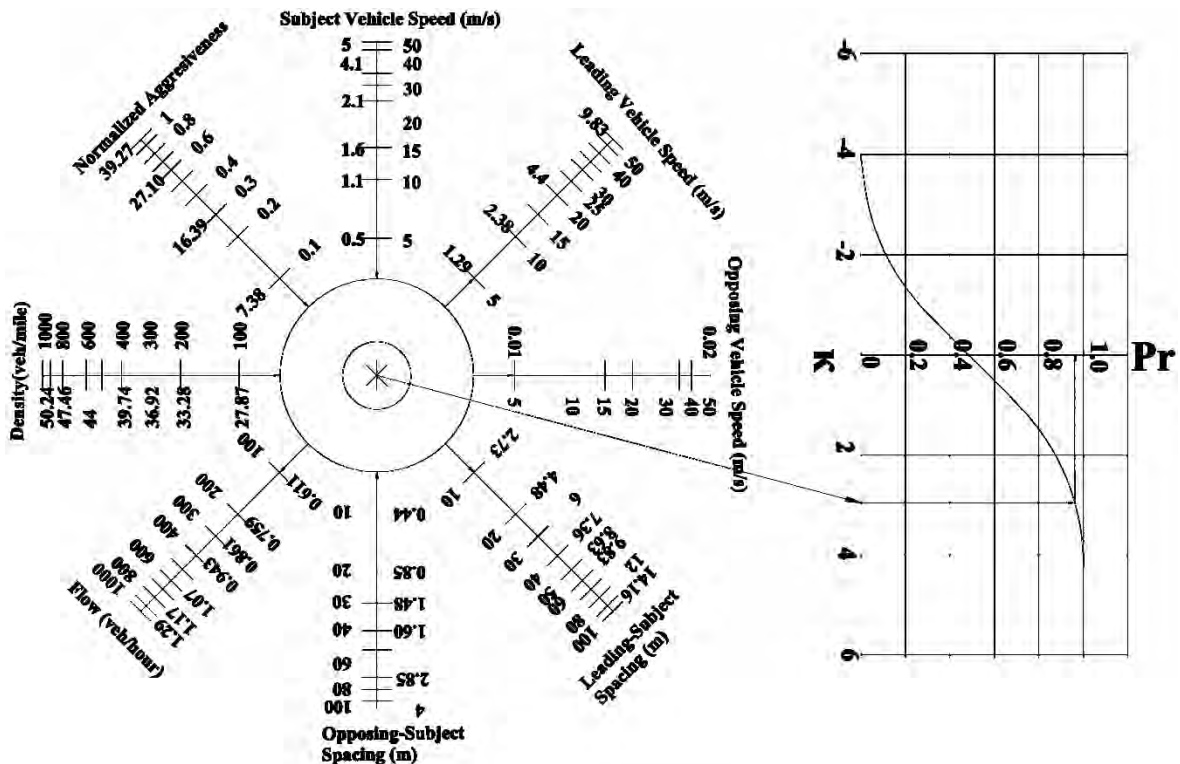
(i) Temporal change in crash probability while overtaking

Figure 9.1 Change in probability with explanatory variables.

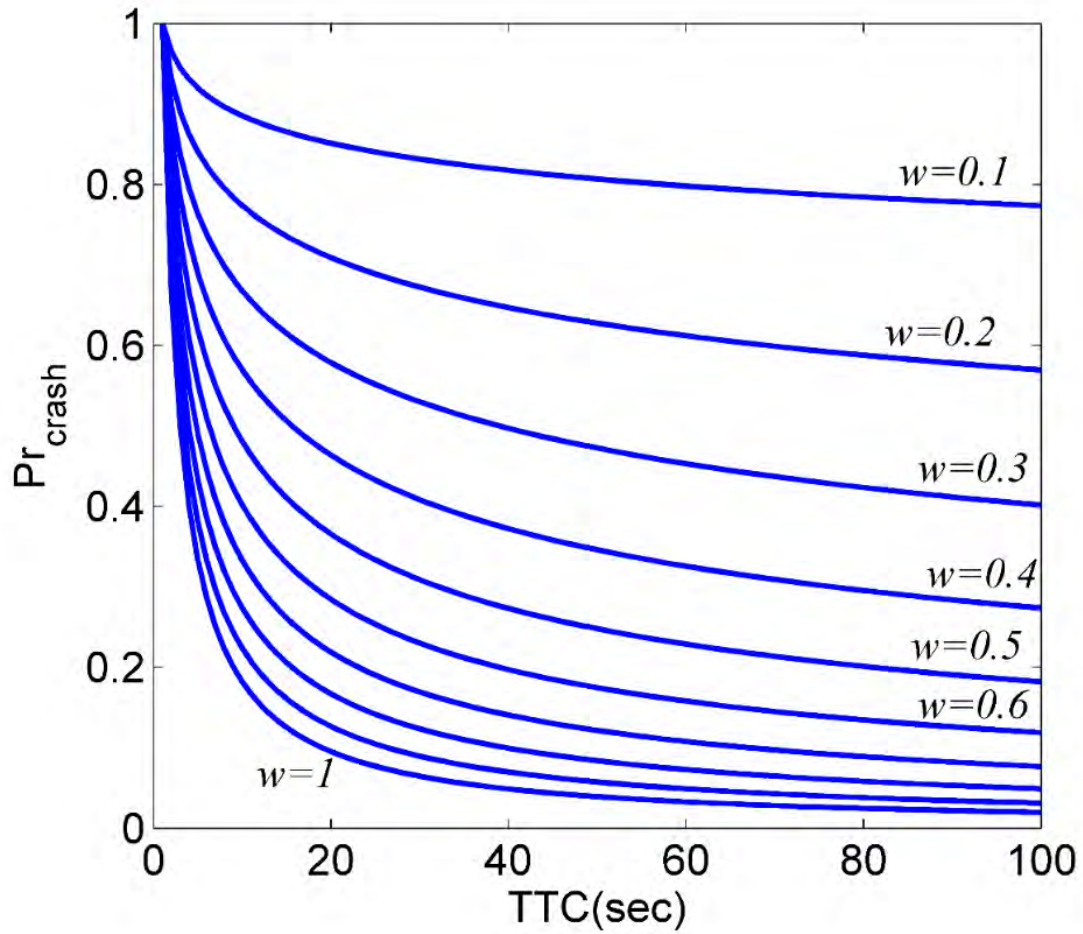
9.3. Chart for Practical Use

To increase the applicability of the developed models, nomographs were established. Nomograph for both OD model and the CP model are provided in Figure 9.2. Figure 9.2(a)

represents the nomograph for OD model, which was developed using equation (5.11). In this nomograph, the values of the variables were labeled in the right and corresponding output was on the left of a particular scale. Total eight scales were provided for eight variables. The output O_i corresponding to all the variables X_{io} were obtained from Figure 9.2 (a). The signs (+ve or -ve) were obtained from the direction indicated on the scale. Inward direction represented positive and outward represented negative. For example, if the subject vehicle speed was 15 m/s, the corresponding output would be +1.6. In contrast, if the leading vehicle speed was 10m/s, the corresponding output would be -2.38. In this way, all the O_i corresponding to X_{io} was determined. Considering the signs, $\kappa = \sum_{i=1}^n O_i$ was determined, and the κ was transferred to the chart on the right of Figure 9.2 (a). Finally, the probability $(Pr(OD=1|X_{io}))$ corresponding to the summed value was obtained. Additionally, Figure 9.2(b) represented the nomograph for CP model. Crash probability $(Pr'(C_{nk}^t | TTC_{nk}^t))$ while overtaking against different TTC was obtained from equation (5.17), which was plotted for different W value in this nomograph. These values obtained from Figure 9.2(a) and 9.2(b) was inserted into equation (5.16) to find the crash probability $Pr'(C_{nk}^t)$ due to overtaking.



(a) Overtaking Probability



(b) Crash probability

Figure 9.2 Nomographs.

9.4. Summary

In this chapter, the correlation among the variables in overtaking model is discussed. It has been seen that subject vehicle speed, lead vehicle speed, lead and opposite vehicle spacing, aggressiveness of the subject vehicle are highly correlated with the overtaking incident. These variable are also found to have significant impact on the overtaking model. Ultimately, for practical use two nomographs has been included in the end of the chapter. These nomographs will make finding the crash probability of an individual vehicle easily.

Chapter 10

Conclusion and Recommendations

10.1. Concluding Remarks

This study investigated the crash probability of different types of vehicle from vehicle trajectory. The crash probability of individual vehicle considered the overtaking probability of subject vehicle as well as the overtaking probability of leading vehicle. However, the overtaking probability of leading vehicle was modified using an importance factor which considered the type of overtaking (normal/flying/piggyback). This overtaking probability model included eight variables. Particularly, a nonlinear random parameter multivariate binary logistic regression model was developed depending on the characteristics of those variables. This model considered normal distribution of the model parameters. Moreover, a new TTC formulation was provided considering the dynamic acceleration of the subject vehicle. Therefore, incorporating both overtaking decision model and TTC formulation, a new crash probability estimation model was proposed. The novelty of the model lied in the consideration of crash frequency. Video from two-lane undivided highway was collected using a mounted camera. Vehicles were detected using background subtraction technique and vehicle trajectories were obtained from Kalman filter. The trajectories were compiled into model compatible samples. Using these samples, firstly, the overtaking decision model was calibrated using Metropolis-Hastings algorithm. Analysis shown that the proposed model achieved 0.66 McFaddens' ρ^2 value. It represented good fit of the model within the data. Furthermore, the constant term of the model was found to be insignificant, which indicated that the effects of the unobserved variables were insignificant. Thus, the considered variables were adequate to express the overtaking decision. Moreover, the exponents of the corresponding variable helped to investigate the impact of the variables. The variables were ranked according to their exponent. It was found that, subject vehicle speed and subject-opposing vehicle spacing had more impact on the model compared to other variables. Finally, crash probability for different types of vehicles was estimated using the CP model. It showed that bus and freight vehicle had higher crash risk compared to other vehicles for the study location. Moreover, nomographs were developed to ease of crash probability estimation for practitioners. Main conclusions from this research are summarized chapter-wise below.

Chapter 2 gives an overview of the recent studies regarding the overtaking models, time-to-collision, background modeling, vehicle detection and vehicle trajectory. It is evident that most of the studies use simulated data for modeling the overtaking phenomena. Besides, the overtaking behavior modeling related studies focus either on explaining the influencing variables or simulating the decision. Thus, they haven't found the risk involved within overtaking maneuver. Moreover, the overtaking models for different vehicle types are still unexplored. Conversely, the traditional formulation of TTC considers only the spacing and the speed differential between the vehicles. It does not consider dynamic accretion of vehicles, which leads to over estimation of TTC. Consequently, the use of traditional TTC decreases the safety. Moreover, it is also evident that vehicle detection is the most convenient procedure for traffic measurement. On the other hand, background subtraction has become a popular technique in vehicle detection. However, it suffers from various visual challenges and very few researches have solved them individually and at the same time. Moreover, no such system has been developed to monitor real-time traffic flow in non-lane-based traffic stream. From the literature survey, it has been revealed that none of the above background model gives any explicit solution to illumination variation while traffic detection. Moreover, now-a-days various State-of-Art vision based tracking methods have become easily accessible. An object tracking method based on Kalman filter was proposed, which is able to ensure an efficient and robust tracking with merge and split of multi-object. The coordinate from tracking can be interpreted as object's trajectory. There have been varied approaches to handle the trajectory of moving objects analysis based on video and some solutions have already been proposed.

Chapter 3 mainly focuses on developing an efficient background model to use in background subtraction technique while vehicle detection. Background subtraction is a very common approach in vision based traffic detection. However, an accurate background is needed to classify the foreground correctly. Unfortunately, it is difficult to get such background as it is not static and it is occupied with objects most of the time. Thus, the necessity of accurate background modeling emerges for accurate traffic detection. In this chapter, an efficient background model has been derived. In this context, three theorems were proposed to define different components of an image, which ultimately differentiates background and foreground. The theorems brought out three important properties of pixels in images. The first theorem entails that, background and foreground have different nature of occurrence. The second one denies co-existence of two types of pixels in a same time-space. Finally, the image was expressed as matrix summation of background and foreground using the third theorem.

Moreover, incorporating these theorems and considering first-order model for illumination variation along with the Fourier transformation for traffic arrival patterns, the background and foreground were defined. These definitions fueled the formulation of the problem, where the concept of static background is utilized. As a solution to this problem, successively three dynamic background models were developed; however, the first model cannot capture illumination variation to the full extent and the second model causes severe accumulation problem due to lack in available frames for dynamic background estimation. Hence, the third model was developed to overcome the limitations of the previous ones which became possible because of incorporating 2 types of parameters: (1) Luminance controlling parameters; and (2) Pollution controlling parameters. Luminance controlling parameters capture the illumination variation; whereas the pollution controlling parameters retard the model pollution due to accumulation of traffic into the background. Besides, a new per pixel binary threshold model related to the final model was developed for the foreground segmentation. It uses a linear relationship between threshold and distance matrix to find per pixel threshold. Although this threshold model contains two parameters, one of them is significant while making binary decision.

Chapter 4 presents a new methodology for traffic measurement using video sensors. PARTS (Pixel Based Heterogeneous Traffic Measurement), a user-friendly tool that incorporates the proposed methodology, has been developed to facilitate real-time traffic measurement. This tool is capable of providing accurate and informative traffic measurements under different road and traffic characteristics. Particularly, the incorporation of luminance and pollution control parameters result in accurate background estimation. Whereas, the new heuristic dynamic threshold-difference function for determining per pixel threshold results in accurate foreground segmentation. Moreover, shadow is removed considering its physical characteristics by the PNS (Positive Negative Segmentation) technique. Ultimately, the concepts of impulse flow wave and segment based speed measurement have contributed to the accuracy in measuring flow and speed respectively. This chapter also includes the description of user-friendly software interface developed for extracting data using the methodology illustrated in this chapter.

Chapter 5 is the most important part of this thesis. In this chapter, the methodology of estimating crash probability from vehicle trajectory has been presented. A model is developed that estimates head-on crash probability for classified vehicle trajectory. Nonlinear random

parameter multivariate binary logistic regression is used to model drivers' overtaking decision. On the other hand, a new formulation of TTC is proposed that considered the dynamic acceleration of the subject vehicle. For model calibration and real-time crash probability estimation, vision based trajectories were exploited. Finally, the conjugation of OD model and TTC formulation gave the classified crash probability.

Chapter 6 provides step-by-step procedure of handling WinBUGS for applying the Markov chain Monte Carlo (MCMC) method. Defining and compiling a model, importing the data, initializing the model, simulating the model, extracting the parameter statistics are included in this chapter. Illustration of each step is also provided for ensuring further insight of the software.

Chapter 7 presents different types of data collected for this thesis work. The data have been sorted into three types. In the first one six video dataset are collected: (1) 'Office'; (2) 'Pedestrians'; (3) 'PETS2006'; (4) 'Highway'; (5) 'Boat'; and (6) 'Blizzard'. These videos contain a mixture of mild to hard challenges such as gradual to sudden illumination variation, stop and go traffic situation, which are similar to the problems that can be addressed through M3. In the second one, three different locations in Dhaka city are chosen to validate the flow measurement methodology incorporated within PARTS. These locations comprise of different roadway and traffic characteristics. Specifically, the first location is an urban signalized intersection containing heterogeneous motorized and non-motorized vehicles. The next location is a 4-lane urban arterial link. It contains high speed uninterrupted motorized vehicles, where passenger cars are dominant. The third location is a two-lane rural highway containing high speed uninterrupted vehicles. However, in contrast to the other two locations, it comprises of mostly large bus and trucks. Two hours videos have been captured in each of the locations. The mounting height of the cameras is at least 20ft and their angle is less than 45 degrees to reduce detecting the object details. For the same period, ground truth data (speed and flow) has also been collected from the video through manual post-processing. For ground truth and PARTS-based speed measurement, a strip of 88ft is chosen within the FOV. In the third, a 2.5 hour video is captured from a two-lane undivided rural highway containing high speed uninterrupted vehicles. To avoid detailed object detection, the mounting height of the cameras is kept at 20ft and their angle was less than 45 degrees. The video contains gradual to sudden illumination variation, vehicular shadows and small camera jitter due to wind.

Chapter 8 illustrates the calibration and validation of all the newly proposed models discussed in this thesis. Calibration of the background model is done using constrained optimization of the parameters. However, the boundary values are calculated from the graphical calibration. Using these calibrated parameters, the model was validated using different video dataset. The validation result shows that, the model output is representative of ground truth yielding more than 95% PCC value and 90% precision-recall value. It was also evident from the result that the main limitation of this model is not being able to capture the background dynamics as observed in ‘Boats’ due to flowing water and camouflage as observed in ‘Blizzard’ where both vehicle and pavement were covered with snow. However, field test shows that the model achieves 93% AAOC in morning peak period, which represent its robustness in traffic detection. Afterwards the background model is tested for traffic measurements using PARTS. This tool is capable of providing accurate and informative traffic measurements under different road and traffic characteristics. Specifically, the tool provides more than 95% Pearson correlation values for speed measurement at all the study locations. Whereas, this value is around 90% in case of flow measurement. Moreover, among the study locations, analyses show that PARTS produces better results in case of the rural highway and urban arterial link compared to the urban intersection. Although the correlation for flow measurement in urban intersection is somewhat lower, it is quite satisfactory. The main reason behind lower correlation is erosion of small vehicles that are present in the intersection by the morphological operations (i.e. dilation and erosion). On the other hand, speed measurement is not impaired by the presence of small vehicles as PARTS considers only aggregated differential centroid movement. Ultimately, this chapter presents the model calibration procedure using Metropolis-Hastings algorithm. Table 9.2 shows the calibrated values of the parameters for all vehicles using Metropolis-Hastings algorithm. The results of the model showed very interesting and significant conclusions which can be summarized below:

- (1) It was found that the increase in subject vehicle speed also increased the probability of overtaking due to increasing speed difference with the leading vehicle.
- (2) If the speed of the leading vehicle speed increased, the probability of overtaking decreased. Because, increase in speed of the leading vehicle caused more difficulty to complete the overtaking maneuver.
- (3) The overtaking probability increased with the speed difference between the subject vehicle speed and lead vehicle speed. It also comprised with the findings in (18).

- (4) The probability of overtaking decreased for increasing speed of the opposing vehicle. Higher speed of the opposing vehicle caused the subject-opposing gap to decrease more quickly making overtaking difficult.
- (5) The higher the spacing between the leading and subject vehicle, the lower the overtaking tendency could be found. Studies (18, 21) also shown same tendency, when following gap decreased. It revealed that the subject vehicle usually overtook, when it came closer to leading vehicle.
- (6) If the spacing between subject and opposing vehicle increased the overtaking probability increased, as larger passing gaps increased the chances of overtaking.
- (7) With the increment of traffic density, the probability of overtake decreased. This is because of larger passing gaps that were available in lower traffic volumes and consequently, the overtaking chances were larger.
- (8) Finally, increase in aggressiveness also increased the probability of overtaking.

The two-tailed t-stat of the coefficients revealed that all the variables were significant at 95% confidence interval except the constant term, as those were greater than the t-critical (1.96) value. Since the constant term was insignificant, it indicated that the considered variables were well enough to describe the overtaking phenomena.

10.2. Limitations of the Study

Although the model performed well and attained a well goodness of fit, there are some limitations of this study:

- (1) The main limitation of background estimation model introduced in this thesis is not being able to capture the background dynamics as observed in 'Boat' due to flowing water and camouflage as observed in 'Blizzard' where both vehicle and pavement were covered with snow. Thus, dynamic natural phenomena such as rainfall/mist will impair the accuracy of the background model.
- (2) PARTS system developed in this study cannot handle longitudinal vehicle occlusions, severe camera vibrations, and camouflage problems at the current stage. Depending on the presence and severity of these problems, the traffic measurement accuracy may get affected.
- (3) The crash probability estimation model developed in this study considers only traffic and driver characteristics related variables. For more complex scenario, the number of variables might increase in the OD model to cope up with the added complexity. Moreover, the

coverage of crash probability estimation is dependent on the field of vision (FOV); whereas FOV is controlled by the mounting height and the angle of the camera.

10.3. Recommendations for Future Research

Since overtaking model and crash probability have been studied more than a decade in the developed world, research on this topic in Bangladesh as well as in other south-east Asian countries is extremely scarce and challenging. This is mainly due to the complexity of data collection and processing and absence of proper mathematical framework. Even though the current study tried to focus some effort in this sector, it can't be viewed as a complete understanding of the highly complex behavior involved in overtaking decision making. Further research to explore other forms of overtaking decision model for better representation of the risk involved in this hazardous maneuver. In this section, some recommendations are provided for future research following the studies carried out in this dissertation. These are listed below,

- (1) The aforementioned (subsection 10.2) two background estimation challenges will be investigated using the same theoretical framework developed in this thesis. Furthermore, detecting traffic at night is also a great challenge, which can also be explored using this framework
- (2) There is a plan to improve the PARTS tool by incorporating more modules to remove its weaknesses as stated in section 10.2. Like-wise, the current research scope does not include the validation of the tool at night time. However, its background estimation module has the capability to estimate background also at night. Additionally, night time vehicle detection is another task that includes different challenges other than day time. Thus, the tool keeps a scope to verify it at night time. Moreover, now-a-days information on vehicle trajectory has become urgent in different fields of research like estimating the lane changes in weaving segments and predicting traffic conflicts. These can also be transformed into separate module and incorporated as a part of it.
- (3) For capturing more complex scenario, increased number of variables in the OD model will be considered to cope up with the added complexity. It keeps a great scope for this research to extend in the future. Moreover, crash probability estimation of different types of vehicle for different location will enable researchers and practitioners to compare the locations in terms of safety. It will also help them to categorize different locations as per crash probability and to determine the magnitude of cost effective treatment for those locations.

REFERENCES

1. Accident Research Institute, Bangladesh University of Engineering & Technology, Dhaka.
<http://ari.buet.ac.bd/Downloads.php>.
2. Gårder, P. Segment characteristics and severity of head-on crashes on two lane rural highways in maine. *Accident Analysis Prevention*, Vol. 38, No. 4, 2006, pp. 652–661
3. Bham, G. H., B. S. Javvadi, and U. R. Manepalli. Multinomial logistic regression model for single-vehicle and multivehicle collisions on urban US highways in Arkansas. *Journal of Transportation Engineering*, Vol. 138, No. 6, 2011, pp. 786-797.
4. Conroy, C., G.T. Tominaga, S. Erwin, S. Pacyna, T. Velky, F. Kennedy, M. Sise, and R. Coimbra. The influence of vehicle damage on injury severity of drivers in head-on motor vehicle crashes. *Accident Analysis & Prevention*, Vol. 40, No.4, 2008, pp. 1589-1594.
5. Wegman, F. *Fewer Crashes and Fewer Casualties by Safer Roads*. SWOV Institute for Road Safety Research, 2004.
6. *Traffic Safety Facts 2014: A Compilation of Motor Vehicle Crash Data from the Fatality Analysis Reporting System and the General Estimates System*. NHTSA, U. S. Department of Transportation, Washington DC, 2016.
7. Polus, A., M. Livneh, and B. Frischer. Evaluation of the passing process on two-lane rural highways. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1701, 2000, pp. 53-60.
8. Cantin, V., M. Lavallière, M. Simoneau, and N. Teasdale. Mental workload when driving in a simulator: Effects of age and driving complexity. *Accident Analysis and Prevention*, Vol. 41, No. 4, 2009, pp. 763–771.
9. *Highway Capacity Manual: HCM 2010*. Transportation Research Board, NRC, Washington, DC, 2010.
10. Oh, C., J. Oh, and J. Min. Real-time detection of hazardous traffic events on freeways. *Transportation Research Record*, Vol. 2129, 2009, pp.35–44.
11. Saunier, N., and T. Sayed, 2008. Probabilistic framework for automated analysis of exposure to road collisions. *Transportation Research Record*, Vol. 2083, 2008, pp. 96–104.
12. Oh, C., and T. Kim. Estimation of rear-end crash potential using vehicle trajectory data. *Accident Analysis & Prevention*, Vol. 42, No. 6, 2010, pp. 1888-1893.

13. AASHTO 1994. *A policy on geometric design of highways and streets*. American Association of State Highway and Transportation Officials, Washington, DC, 1994.
14. Hegeman, G., S. Hoogendoorn, and K. Brookhuis. Observations overtaking manoeuvres on bi-directional roads. *Advanced OR and AI Methods in Transportation*, Vol. 1, 2005, pp. 505–510.
15. Bar-Gera, H., and D. Shinar. The tendency of drivers to pass other vehicles. *Transportation Research Part F*, Vol.8, 2005, pp. 429–439.
16. Jenkins, J. M., and L. R. Rilett. Application of distributed traffic simulation for passing behavior study. *Transportation Research Record: Journal of the Transportation Research Board*, Vol.1899, 2004, pp. 11–18.
17. Pollatschek, M.A., and A. Polus. Modeling impatience of drivers in passing maneuvers. In *Mahmassani, H.S. (Ed.), Transportation and traffic theory: Flow, dynamics and human, interaction*, 2005, pp. 267–279.
18. Farah, H., and T. Toledo. Passing behavior on two-lane highways. *Transportation Research Part F*, Vol.13, No. 6, 2010 pp. 355–364.
19. Llorca, C., A. T. Moreno, A. Lenorzer, J. Casas, and A. Garcia. Development of a new microscopic passing maneuver model for two-lane rural roads. *Transportation Research Part C: Emerging Technologies*, Vol. 52, 2015, pp.157-172.
20. Khoury, J.E., and A.G. Hobeika. Integrated stochastic approach for risk and service estimation: passing sight distance application. *Journal of Transportation Engineering*, Vol. 135, No. 5, 2012, pp. 571–579.
21. Vlahogianni, E. I., and J. C. Golias. Bayesian modeling of the microscopic traffic characteristics of overtaking in two-lane highways. *Transportation Research Part F: Traffic Psychology and Behaviour*, Vol. 15, No. 3, 2012, pp. 348-357.
22. Motro, M., A. Chu, J. Choi, P. S. Lavieri, A. R. Pinjari, C. R. Bhat, J. Ghosh, and R. W. Heath. Vehicular ad-hoc network simulations of overtaking maneuvers on two-lane rural highways. *Transportation Research Part C: Emerging Technologies*, Vol. 72, 2016, pp.60-76.
23. Tapani, A. *Traffic simulation modelling of rural roads and driver assistance systems (Ph.D. dissertation)*. Department of Science and Technology, Linköping University, 2008.
24. Hegeman, G., A. Tapani, and S. Hoogendoorn. Overtaking assistant assessment using traffic simulation. *Transportation Research Part C: Emerging Technologies*, Vol. 17, No. 6, 2009, pp. 617–630.

25. Ghods, A.H., F. Saccomanno, and G. Guido. Effect of car/truck differential speed limits on two-lane highways safety operation using microscopic simulation. *Procedia-Social and Behavioral Science*, Vol. 53, 2012, pp.833–840.
26. Yu, Y., A. El Kamel, and G. Gong. Modeling overtaking behavior in virtual reality traffic simulation system. In *9th Asian Control Conference (ASCC)*, Istanbul, Turkey, June, 2013, pp. 1–6.
27. Barnich, O., and D. M.Van. ViBe: a powerful random technique to estimate the background in video sequences. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 945-948.
28. Hofmann, M., P. Tiefenbacher and G. Rigoll. Background segmentation with feedback: The pixel-based adaptive segmenter. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 38-43.
29. Hadiuzzaman, M., N. Haque, F. Rahman, S. Hossain, M. R. K. Siam and T. Z. Qiu. Pixel-based heterogeneous traffic measurement considering shadow and illumination variation. *Signal, Image and Video Processing*, 2017, pp.1-8.
30. Kasprzak, W. An iconic classification scheme for video-based traffic sensor tasks. In *International Conference on Computer Analysis of Images and Patterns*, 2001, pp. 725-732.
31. Frenando, W.S.K., H.M.S.P.B. Hearth, P.H. Perera, M.P.B. Ekanayake, G.M.R.I. Godaliyadda, and J.V. Wijayakulasooriya. Object Identification, Enhancement and Tracking under Dynamic Background Conditions. In *7th International Conference on Information and Automation for Sustainability*, Colombo, 2014, pp. 1-6.
32. Hu, W., X. Xiao, D. Xie, and T. Tan. Traffic accident prediction using vehicle tracking and trajectory analysis. In *Intelligent Transportation Systems. Proceedings. 2003 IEEE*, Vol. 1, 2003, pp. 220–225.
33. T. S. Kumar, and S. N. Sivanandam. A Modified Approach for Detecting Car in video using Feature Extraction Techniques, *European Journal of Scientific Research*, ISSN 1450- 216X, Vol. 77, No. 1, 2012, pp. 134-144.
34. Li, X., K. Wang, W. Wang, and Li, Y. A multiple object tracking method using Kalman filter. In *Information and Automation (ICIA), 2010 IEEE International Conference on*, IEEE, 2010, pp. 1862-1866.
35. Song, H. S., S. N. Lu, X. Ma, Y. Yang, X. Q. Liu and P. Zhang. Vehicle behavior analysis using target motion trajectories. *IEEE Transactions on Vehicular Technology*, Vol. 63, No. 8, 2014, pp. 3580–3591.

36. Minderhoud, M.M., and P.H.L. Bovy. Extended time-to-collision measures for road traffic safety assessment. *Accident Analysis and Prevention*, Vol. 33, No. 1, 2001, pp. 89-97.
37. Hayward, J. Ch. *Near miss determination through use of a scale of danger. Report no. TTSC 7115*, The Pennsylvania State University, Pennsylvania, 1972.
38. Svensson, Å. *A Method for Analysing the Traffic Process in a Safety Perspective. Doctoral Dissertation*, University of Lund, Lund, Sweden, 1998
39. Farah, H., S. Bekhor, and A. Polus. Risk evaluation by modeling of passing behavior on two-lane rural highways. *Accident Analysis & Prevention*, Vol. 41, No. 4, 2009. pp. 887-894.
40. Bachmann, C., M. Roorda, and B. Abdulhai. Improved time-to-collision definition for simulating traffic conflicts on truck-only infrastructure. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2237, 2011, pp. 31-40.
41. Ni, D., L. Li, H. Wang, and C. Jia. Observations on the fundamental diagram and their interpretation from the human factors perspective. *Transportmetrica B: Transport Dynamics*, Vol. 5, No. 2, 2017, pp. 163-180.
42. Lee, B., Hedley, M. Background estimation for video surveillance. *Image and Vision Computing New Zealand* , Vol. 1, 2002 pp. 315–320.
43. McFarlane, N., Schofield, C. Segmentation and tracking of piglets in images. *British Machine Vision and Applications*, Vol. 1, 1995, pp. 187–193.
44. Zheng, J., Wang, Y., Nihan, N., Hallenbeck, M. Extracting roadway background image: mode-based approach. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1944, 2006, pp. 82-88.
45. Wren, C.R., Porikli, F. Waviz: spectral similarity for object detection. *In Proceedings of the IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2005, pp. 55-61.
46. Kim, H., Sakamoto, R., Kitahara, I., Toriyama, T., Kogure, K. Robust foreground extraction technique using Gaussian family model and multiple thresholds. *In Proceedings of the Asian Conference on Computer Vision*, 2007, pp. 758-768.
47. Stauffer, C., Grimson, WEL. Adaptive background mixture models for real-time tracking. *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999.

48. Guo, L., Du, M. H. Student's t-distribution mixture background model for efficient object detection. *In Proceedings of the IEEE International Conference on Signal Processing, Communication and Computing*, 2012, pp. 410-414.
49. Haines, T. S., Xiang, T. Background subtraction with dirichlet processes. *In Proceedings of the European Conference on Computer Vision*, 2012, pp. 99-113.
50. Elgammal, A., Harwood, D., Davis, L. Non-parametric model for background subtraction. *In Proceedings of the European Conference on Computer Vision*, 2000, pp. 751-767.
51. Ding, X., He, L., Carin, L. Bayesian robust principal component analysis. *IEEE Transactions on Image Processing*, Vol. 20, No.12, 2011, pp. 3419-3430.
52. Barnich, O., Van, D. M. ViBe: a powerful random technique to estimate the background in video sequences. *In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 945-948.
53. Hofmann, M., Tiefenbacher, P., Rigoll, G. Background segmentation with feedback: The pixel-based adaptive segmenter. *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 38-43.
54. Lin, Z., Ganesh, A., Wright, J., Wu, L., Chen, M., Ma, Y. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *In Proceedings of the Computational Advances in Multi-Sensor Adaptive Processing Workshop*, 2009.
55. Wang, J., Bebis, G., Miller, R. Robust video-based surveillance by integrating target detection with tracking. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2006, pp. 137-137.
56. Tavakkoli, A., Nicolescu, M., Bebis, G. A novelty detection approach for foreground region detection in videos with quasi-stationary backgrounds. *In International symposium on visual computing*, 2006, pp. 40-49.
57. Oliver, N. M., Rosario, B., Pentland, A. P. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22 No. 8, 2000, pp. 831-843.
58. Bouwmans, T. Subspace learning for background modeling: a survey. *Recent Patents on Computer Science*, Vol. 2, No. 3, 2009, pp. 223-234.
59. Farcas, D., Marghes, C., Bouwmans, T. Background subtraction via incremental maximum margin criterion: a discriminative subspace approach. *Machine Vision and Applications*, Vol. 23 No. 6, 2012, pp. 1083-1101.

60. Marghes, C., Bouwmans, T., Vasiu, R. Background modeling and foreground detection via a reconstructive and discriminative subspace learning approach. *In Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition*, Vol. 2, 2012, pp. 106-112.
61. Uray, M., Skocaj, D., Roth, P. M., Bischof, H., Leonardis, A. Incremental LDA learning by combining reconstructive and discriminative approaches. *In Proceedings of British Machine Vision Conference*, Vol. 1, 2007, pp. 272-281.
62. Toyama, K., Krumm, J., Brumitt, B., Meyers, B. Wallflower: Principles and practice of background maintenance. *In Proceedings of the 7th IEEE International Conference on Computer Vision*, Vol. 1, 1999, pp. 255-261.
63. Karnna, K. P., Raja, Y., Gong, S. Moving object recognition using an adaptive background memory. *Time-varying Image Processing and Moving Object Recognition*, Vol. 2, 1990, pp. 289–307.
64. Butler, D. E., Bove, Jr. V. M., Sridharan, S. Real-time adaptive foreground/background segmentation. *EURASIP Journal on Advances in Signal Processing*, Vol. 14, 2005, pp. 1-13.
65. Xiuman, D., Guoxia, S., Tao, Y. Moving target detection based on Genetic K-means Algorithm. *In Proceedings of the International Conference on Communication Technology*, 2011, pp. 819–822.
66. Kim, K., Chalidabhongse, T. H., Harwood, D., Davis, L. Background modeling and subtraction by codebook construction. *In Proceedings of the IEEE International Conference on Image Processing*, Vol. 5, 2004, pp. 3061-3064.
67. Culibrk, D., Marques, O., Socek, D., Kalva, H., Furht, B. Neural network approach to background modeling for video object segmentation. *IEEE Transactions on Neural Networks*, Vol. 18, No. 6, 2007, pp. 1614-1627.
68. Luque, R. M., López-Rodríguez, D., Merida-Casermeiro, E., Palomo, E. J. Video object segmentation with multivalued neural networks. *In Proceedings of the 8th IEEE International Conference on Hybrid Intelligent Systems*, 2008, pp. 613-618.
69. Palomo, E. J., Domínguez, E., Luque, R. M., Muñoz, J. Image hierarchical segmentation based on a GHSOM. *In Proceedings of International Conference on Neural Information Processing*, 2009, pp. 743-750.
70. Kim, W., Kim, C. Background subtraction for dynamic texture scenes using fuzzy color histograms. *IEEE Signal processing letters*, Vol. 19, No. 3, 2012, pp. 127-130.

71. Tezuka, H., Nishitani, T.. A precise and stable foreground segmentation using fine-to-coarse approach in transform domain. *In Proceedings of the 15th IEEE International Conference on Image Processing*, 2008, pp. 2732-2735.
72. Gao T., Liu, Z. G., Gao, W.C., Zhang J. A robust technique for background subtraction in traffic video. *In Proceedings of the International Conference on Neural Information Processing*, 2008, pp. 736-744.
73. Guan, Y. P. Wavelet multi-scale transform based foreground segmentation and shadow elimination. *The Open Signal Processing Journal*, Vol. 1, No. 6, 2008, pp. 1-6.
74. Wang, Y., Jodoin, P. M., Porikli, F., Konrad, J., Benezeth, Y., Ishwar, P. Change Detection Challenge. Available from <http://www.changedetection.net>
75. Davis, J., Goadrich, M. The relationship between Precision-Recall and ROC curves. *In Proceedings of the 23rd International Conference on Machine Learning*, 2006, pp. 233-240.
76. Piccardi, M. Background subtraction techniques: a review. *IEEE international conference on systems, man and cybernetics*, Vol. 4, 2004, pp. 3099-3104.
77. Mandellos, N. A., Keramitsoglou, I., Kiranoudis, C. T. “A background subtraction algorithm for detecting and tracking vehicles”. *In Expert Systems with Applications*, vol. 38, no. 3, pp.1619-1631, 2011.
78. Guo, L., Ge, P. S., Zhang, M. H., Li, L. H., Zhao, Y. B. “Pedestrian detection for intelligent transportation systems combining AdaBoost algorithm and support vector machine”. *In Expert Systems with Applications*, vol. 39, no 3, pp. 4274-4286, 2012.
79. Coifman, B., Beymer, D., McLauchlan, P., Malik, J. “A real-time computer vision system for vehicle tracking and traffic surveillance”. *In Transportation Research Part C: Emerging Technologies*, vol. 6, no. 4, pp. 271-288, 1998.
80. Bechler, M., Bohnert, T. M., Cosenza, S., Festag, A., Gerlach, M., Seeberger, D. “Evolving the European its architecture for car-to-x communication”. *In Proc. 16th ITS WC, Stockholm, Sweden*, pp. 1–8, 2009.
81. Buch, N., Yin, F., Orwell, J., Makris, D., Velastin, S. A. “Urban vehicle tracking using a combined 3-D model detector and classifier”. *In Knowledge-Based and Intelligent Information and Engineering Systems KES. Santiago, Chile: Springer-Verlag*, vol. 5711, pp. 169-176, 2009.

82. Park, K., Lee, D., Park, Y. "Video-based detection of street-parking violation". In *Hamid R. Arabnia, ed., 'IPCV', CSREA Press*, pp. 152-156, 2007.
83. Haritaoglu, I., Harwood, D., Davis, L.S. "W 4: real-time surveillance of people and their activities". In *Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809–830, 2000.
84. Jacques, J., Jung, C., Musse, S. "Background subtraction and shadow detection in grayscale video sequences". In *Brazilian Symposium on Computer Graphics and Image Processing, Natal, Brazil*, pp. 189–196, 2005.
85. Chen, X., Zhang, C. C. "Vehicle classification from traffic surveillance videos at a finer granularity". In *Advances in Multimedia Modeling*. Springer Berlin Heidelberg, vol. 4351, pp. 772–781, 2007.
86. Kanhere, N. K., Birchfield, S. T. "Real-time incremental segmentation and tracking of vehicles at low camera angles using stable features". In *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 148–160, 2008.
87. Gupte, S., Masoud, O., Martin, R. F. K., Papanikolopoulos, N. P. "Detection and classification of vehicles". In *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 37–47, 2002.
88. Zheng, J., Wang, Y., Nihan, N. L., Hallenbeck, M. E. "Extracting roadway background image: Mode-based approach", In *Transp. Res. Rec.*, vol. 1944, pp. 82–88, 2005.
89. Cucchiara, R., Piccardi, M., Prati, A. "Detecting moving objects, ghosts, and shadows in video streams". In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no.10, pp.1337-1342, 2003.
90. Zhou, Q., Aggarwal, J. "Tracking and classifying moving objects from videos". In *Proceedings of IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, 2001.
91. McFarlane, N., Schofield, C. "Segmentation and tracking of piglets in images". In *Machine Vision and Applications*, vol. 8, no. 3, pp. 187-193, 1995.
92. Remagnino, C., Baumberg, A., Grove, T., Hogg, D., Tan, T., Worrall, A., Baker, K. "An integrated traffic and pedestrian model-based vision system". In *Proceedings of the Eighth British Machine Vision Conference*, pp. 380-389, 1997.
93. Morris, B., Trivedi, M. "Robust classification and tracking of vehicles in traffic video streams". In *Proc. IEEE ITSC*, pp. 1078-1083, 2006.

94. Su, X., Khoshgoftaar, T. M., Zhu, X., Folleco, A. "Rule-based multiple object tracking for traffic surveillance using collaborative background extraction". In *Advances in Visual Computing*. Springer Berlin Heidelberg, pp. 469–478, 2007.
95. Stauffer, C., Grimson, W.E.L. "Adaptive background mixture models for real-time tracking". In *Computer Vision and Pattern Recognition , IEEE Computer Society Conference on., Fort Collins, CO*, vol. 2, pp. 252, 1999.
96. Frenando, W.S.K., Hearth, H.M.S.P.B., Perera, P.H., Ekanayake, M.P.B., Godaliyadda, G.M.R.I., Wijayakulasooriya, J.V. "Object Identification, Enhancement and Tracking under Dynamic Background Conditions". In *7th International Conference on Information and Automation for Sustainability, Colombo*, pp. 1-6, 2014.
97. Kasprzak, W. "An iconic classification scheme for video-based traffic sensor tasks". In *International Conference on Computer Analysis of Images and Patterns*, pp. 725-732, 2001.
98. Fathy, M., Siyal, M. Y. "A window-based image processing technique for quantitative and qualitative analysis of road traffic parameters". In *IEEE Transactions on Vehicular Technology*, vol. 47, no. 4, pp. 1342-1349, 1998.
99. Figov, Z., Tal, Y., Koppel, M. "Detecting and removing shadows", In *Proc. Of 7th IASTED Int'l Conference on Computer Graphics and Imaging*, 2004.
100. Finlayson, G.D., Hordley, S.D., Drew, M.S. "Removing Shadows from Images". In *7th European Conference on Computer Vision Copenhagen, Denmark*, vol. 2353, pp. 823-836, 2002.
101. Wiess, Y. "Deriving intrinsic images from image sequences". In *ICCV 2001, Eight IEEE International Conference on, Vancouver, BC*, vol. 2, pp. 68-75, 2001.
102. Finlayson, G. D., Drew, M. S., Cheng, L. "Entropy minimization for shadow removal". In *International Journal of Computer Vision*, vol. 85, no. 1, pp. 35-57, 2009.
103. Guo, R., Dai, Q., Hoiem, D. "Single image shadow detection and removal using paired regions". In *Proc. of the 18 IEEE CVPR*, pp. 2033-2040, 2011.
104. Germain, A., Salamaty, N., Susstrunk, S. "Removing shadows from images using color and near -infrared". In *Proc. of the IEEE ICIP*, pp. 1713-1716, 2011.
105. Zhu, J., Samuel, K. G. G., Masood, S. Z., Tappen, M. F. "Learning to recognize shadows in monochromatic natural images". In *Proc. of the IEEE CVPR*, pp. 223-230, 2010.
106. Hockaday, S.L. "Evaluation of image processing technology for applications in highway operations". In *California Polytechnic State University*, 1991.
107. Hoose.,N. "IMPACTS: an image analysis tool for motorway surveillance". In *Traffic engineering & control*, vol. 33, no. 3, 1992.

108. Chatziioanou, A., Hockaday, S., Ponce, L., Kaighn, S., Staley, C. "Video Image Processing Systems Applications in Transportation". In *Vehicle Navigation and Information Systems Conference*, pp. 17-20, 1995.
109. Klein, L. A., Kelley, M. R. "Detection Technology for IVHS", vol. 1, 1996.
110. Kranig, J., Minge, E., Jones, C. "Field Test of Monitoring of Urban Vehicle Operations Using Non-intrusive Technologies", 1997.
111. Michalopoulos, P. G., "Vehicle detection video through image processing: the Autoscope system". In *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 21-29, 1991.
112. Chao, T., Lau, B., Park, Y. "Vehicle detection and classification in shadowy traffic images using wavelets and neural networks". In *Transportation Sensors and Controls: Collision Avoidance, Traffic Management, and ITS, SPIE Proc*, Vol. 2902, pp. 136-147, 1996.
113. Wang, Y., Jodoin, P.M., Porikli, F., Konrad, J., Benezeth, Y., Ishwar, P. "Change Detection Chalange". <http://www.changedetection.net>
114. Hofmann, M., Tiefenbacher, P., Rigoll, G. "Background segmentation with feedback: The pixel-based adaptive segmenter". In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 38-43, 2012.
115. Zhao, Z., Bouwmans, T., Zhang, X., Fang, Y. "A fuzzy background modeling approach for motion detection in dynamic backgrounds". In *Multimedia and Signal Processing*, pp. 177-185, 2012.
116. Braham, M., Van Droogenbroeck, M. "Deep Background Subtraction with Scene-Specific Convolutional Neural Networks". In *International Conference on Systems, Signals and Image Processing*, 2016.
117. Wang, B., Dudek, P. "A fast self-tuning background subtraction algorithm". In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 395-398, 2014.
118. Sedky, M., Moniri, M., Chibelushi, C. C. "Spectral-360: A physics-based technique for change detection". In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 399-402, 2014.