Advisory Safety System for Autonomous Vehicles under Sun-glare

Hamed Esmaeeli

A thesis
In the Department of
Building, Civil, and Environmental Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Doctor of Philosophy (Civil Engineering) at
Concordia University
Montreal, Quebec, Canada

May 2021

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: **Hamed Esmaeeli**

Entitled: **Advisory Vehicle Safety System for Autonomous Vehicles under Sun-glare**

and submitted in partial fulfillment of the requirements for the degree of:

DOCTOR OF PHILOSOPHY (Civil Engineering)

Complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. Chun Wang	Chair
Dr. Luis Amador	BCEE Examiner
Dr. Fuzhan Nasiri	BCEE Examiner
Dr. Anjali Awasthi	External to the program
Dr. Bilal Farooq	External to the University
Dr. Ciprian Alecsandru	Thesis Supervisor

Approved by

Dr. Ashutosh Bagchi	Graduate Program Director
Dr. Mourad Debbabi	Dean of Faculty

May 26th, 2021

ABSTRACT

Advisory Safety System for Autonomous Vehicles under Sun-glare

Hamed Esmaeeli, PhD

Concordia University, 2021

Autonomous Vehicles (AVs) are expected to provide a large number of benefits such as improving comfort, vehicle safety and traffic flow. AVs use various sensors and control systems to empower driver's decision-making under uncertainties as well as, assist the driving task under adverse conditions such as vision impairment. Excessive sunlight has been recognized as the primary source of the reduction in vision performance during daytime. Sun glare oftentimes leads to an impaired visibility for drivers and has been studied from different aspects on roadways. However, there is a lack of knowledge regarding the potential detrimental effects of natural light brightness differential, particularly sun glare on driving behavior and its possible risks.

This dissertation addresses this issue by developing an integrated vehicle safety methodology as an advisory system for safe driving under sun glare. The main contribution of this research is to establish a real-time detection of the vision impairment area on roadways. This study also proposes a Collision Avoidance System Under Sun-glare (CASUS) in which upcoming possible vision impairment is detected, a warning message is sent, and the speed of vehicle is adjusted accordingly.

In this context, real-world data is used to calibrate a psychophysical car-following model within VISSIM, a traffic microscopic simulation tool. Traffic safety impacts are explored through the number of conflicts extracted from the microsimulation tool and assessed by the time-to-collision indicator. Conventional/human-driven vehicles and different type of AVs are modeled for a straight segment of the TransCanada highway under various AVs penetration rates.

The findings revealed a significant reduction in potential collisions due to adjustment of travel speed of AVs under the sun glare. The results also indicated that applying CASUS to the AVs with a failing sensory system improves traffic safety by providing optimal-safe speeds. Furthermore, the CASUS algorithm has the potential to be integrated into driving simulators or real vehicles to further evaluate and examine its benefits under different vision impairment scenarios.

AKNOWLEDGMENTS

This dissertation is the main outcome of a research project that I conducted during the last three and a half years at Concordia University. The dissertation would not have been possible without my willingness to explore, learn, and develop my own ideas; however, I am aware that I have not done this alone. Many people helped and supported me during this process. I would like to take the opportunity to thank them all most sincerely.

I would like to express my deep gratitude to Dr. Ciprian Alecsandru for his kind encouragement and support during the entirety of my Ph.D. study. Under his guidance, I gained invaluable experience in performing scientific research and learned how to overcome various research problems. I collected valuable, high-quality data by cameras, and sensors to develop my work under his support.

I express my deepest acknowledgment to my adorable wife. Without her support and love, I would certainly not finish this interesting but challenging research. Also, I would like to convey my appreciation to my parents for their encouragement in continuing my education.

I would also like to thank my doctoral committee's independent members and the anonymous reviewers of my papers for their valuable comments and suggestions. Revising a manuscript based on the reviewers' comments is not an easy task, but it usually improves the quality of the manuscript substantially. Also, special thanks to the Ministry of Transportation of Quebec (MTQ) for kindly providing me with and letting me use the data they have extensively collected for TransCanada.

Many thanks to MSc. Reza Marandi at Concordia University for the interesting discussions we had and the moments we shared on the development of code and 3D models of obtained results. One of the best things about doing a Ph.D. is that it allows one to interact with and learn from bright people every day.

TABLE OF CONTENTS

LIST OF FIGURES ..... **vi**
LIST OF TABLES ..... **ix**
LIST OF ABBREVIATIONS ..... **x**
Chapter 1 INTRODUCTION **1**
1.1 Research Motivation 1
1.2 Research Objectives 2
1.3 Scope and Limitations 2
1.4 Research Contributions 4
1.5 Research Layout 5
Chapter 2 LITERATURE REVIEW **6**
2.1 Driving behavior in vision impairment 7
2.2 General considerations about modelling and simulation of systems 9
2.3 Autonomous Vehicles 20
2.4 Advanced Collision Avoidance System 29
2.5 Concluding Remarks 34
Chapter 3 EXISTING VISIBILITY AND CREATION OF DATABASE **35**
3.1 Sun Positioning Algorithm 37
3.2 Sun Glare Detection Algorithm 39
3.3 Occlusion Test 41
3.4 Weather Data Analysis 43
3.5 Closing Remarks 45
Chapter 4 A MICROSIMULATION MODEL FOR REDUCED VISIBILITY TRAFFIC CONDITIONS **46**
4.1 Simulation Setup 46
4.2 Calibration 47
4.3 Traffic Flow Composition 48
4.4 Rear-End Crash Assessment 50
4.5 Closing Remarks 52
Chapter 5 COLLISION AVOIDANCE SYSTEM UNDER SUN-GLARE **54**
5.1 Advanced Collision Warning System 56
5.2 Intelligent Braking Assistance 57
5.3 Closing Remarks 58
Chapter 6 NUMERICAL ANALYSIS AND SIMULATION **60**
6.1 Vision impairment detection 60
6.2 A Microsimulation Model for Reduced Visibility Traffic Conditions 68
6.3 Collision Avoidance System Under Sun-Glare 82
Chapter 7 CONCLUSION AND FUTURE DIRECTIONS **88**
Chapter 8 References **91**
8.1 Appendices A: Vision Impairment Detection Python Code 109
8.2 Appendices B: Microsimulation Interface 121

LIST OF FIGURES

Figure 1 . Core framework of driving parameters 3
Figure 2. Study a system 10
Figure 3. Wiedemann 99 car-following model 14
Figure 4. Levels of vehicle autonomy by NHTSA 21
Figure 5. Building blocks of automated driving 22
Figure 6. Different Type of on-vehicle sensors 23
Figure 7. Functional responsibility of on-vehicle mobility sensors 24
Figure 8. Conflict types in SSAM 28
Figure 9. Components of visibility and sun glare detection algorithm 35
Figure 10. The methodology of vision impairment detection 36
Figure 11. Sun position coordinate system 39
Figure 12. Glare angular limits 41
Figure 13. Sketch of a single lane contains sun glare and shadow 42
Figure 14. a) Plan view of the area for occlusion test; b) Glare condition; c) Occluded point 43
Figure 15. Time analysis of a conflict process 51
Figure 16. Component of Collision Avoidance System Under Sun-glare 55
Figure 17. Activation Layout of Advanced Collision Warning System 57
Figure 18. Layout of Intelligent Braking Assistance 58
Figure 19. Study area; a) Center of Montreal include solar path; b) Section of Downtown 61
Figure 20. The expected period of vision impairment in minutes for the morning peak (6:30 to 8:30) in downtown Montreal for the entire year 63
Figure 21. Total time of sun glare (yellow) and shadow (blue) occurrence for Simpson St (a) and (c) and Mackay St – (b) and (d). 65
Figure 22. Annual critical points on road network of downtown Montreal 66
Figure 23. Sun exposure investigation on Chemins de la cote saint Catherine on 18th July 67
Figure 24. Captured sun exposure by fisheye lens 68
Figure 25. Location of study area – Highway A20 Northbound 69
Figure 26. Daily cumulative occurrence of sun glare during the entire year 70
Figure 27. Validation of Sun glare detection algorithm on the highway A20 70
Figure 28. The atmosphere impact on variation of sun irradiance respect to the sun elevation ... 71

Figure 29. Impact of clouds on receiving sun irradiance on the detector at 17th May 72
Figure 30. Prediction of sun glare detection algorithm for the occurrence of sun glare 73
Figure 31. Changes in desired speeds due to sun glare (orange) and overcast sky (blue) 75
Figure 32. Traffic delays of AVNs in light traffic flow and various operational speed 77
Figure 33. Traffic delays of AVNs in moderate traffic flow and various operational speed 77
Figure 34. Traffic delays of AVNs in near congestion traffic flow and various operational speed 77
Figure 35. Traffic delays of AVAs in light traffic flow and various operational speed 78
Figure 36. Traffic delays of AVAs in moderate traffic flow and various operational speed 78
Figure 37. Traffic delays of AVAs in near congestion traffic flow and various operational speed 78
Figure 38. Relative change in the number of conflicts in light traffic flow with different speed reduction and AVN penetration ratio 79
Figure 39. Relative change in the number of conflicts in moderate traffic flow with different speed reduction and AVN penetration ratio 80
Figure 40. Relative change in the number of conflicts in near congestion traffic flow with different speed reduction and AVN penetration ratio 80
Figure 41. Relative change in the number of conflicts in light traffic flow with different speed reduction and AVA penetration ratio 81
Figure 42. Relative change in the number of conflicts in moderate traffic flow with different speed reduction and AVA penetration ratio 81
Figure 43. Relative change in the number of conflicts in near congestion traffic flow with different speed reduction and AVA penetration ratio 82
Figure 44. Positioning vision impairment points on VISSIM model 83
Figure 45. Relative change in the number of conflicts in light traffic flow by applying ACWS with different speed reduction and AV penetration ratio 84
Figure 46. Relative change in the number of conflicts in moderate traffic flow by applying ACWS with different speed reduction and AV penetration ratio 84
Figure 47. Relative change in the number of conflicts in near congestion traffic flow by applying ACWS with different speed reduction and AV penetration ratio 85

Figure 48.Relative change in the number of conflicts in light traffic flow by applying IBA with different speed reduction and AV penetration ratio..... 86
Figure 49. Relative change in the number of conflicts in moderate traffic flow by applying IBA with different speed reduction and AV penetration ratio..... 86
Figure 50. Relative change in the number of conflicts in near congestion traffic flow by applying IBA with different speed reduction and AV penetration ratio..... 87

LIST OF TABLES

Table 1. Summary of literature for the failure of safety sensors 25
Table 2. Driving behavior of AVs logic in VISSIM for longitudinal movement 49
Table 3. Indicated variables in scenario development of mixed traffic 50
Table 4. Decision Sight Distance for different design speed (AASHTO) 57
Table 5. Structure of collected data 61
Table 6. Sky condition during analysis period 73
Table 7. Results of hypothesis test on the impact of sun glare for the average speed of vehicles 74
Table 8. VISSIM optimum parameters by Particle Swarm Optimization (PSO) 76
Table 9. Changes in maximum of critical TTC based on vehicle type 79
Table 10. Optimal speed reduction for different level of AVs in the mixed traffic flow 81

LIST OF ABBREVIATIONS

AASHTO	American Association of State Highway and Transportation Officials
ADAS	Advanced Driver-Assistance System
AE	Autumnal Equinox
AV	Autonomous Vehicle
CASUS	Collision Avoidance System Under Sun-glare
ACWS	Advanced Collision Warning System
DEM	Digital Elevation Model
DOT	Department of Transportation
DSD	Decision Sight Distance
FHWA	Federal Highway Administration
GIS	Geographical Information System
GPS	Geographic Positioning System
IBA	Intelligent Braking Assistance
KM	Kilometer
KPH	Kilometer per hour
MTL	Montreal
MTD	Maximum Trail Deceleration
MTQ	Ministry of Transportation of Quebec
NHTSA	National Highway Traffic Safety Administration
NOAA	National Oceanic & Atmospheric Administration
PSO	Particle Swarm Optimization
SSD	Stopping Sight Distance
TTC	Time-To-Collision

Chapter 1 INTRODUCTION

The safe and efficient operation of vehicles on roadways highly depends on the optimal vision and reaction of drivers. The quality of drivers' vision may be reduced by the existence of some hindering factors (e.g., adverse weather) or impairment of vision (e.g., sun glare). The shining of the direct sunlight into the drivers' eyes has a hindering impact on reducing drivers' vision performance and creating a low safety situation. Under bright sunlight along the road alignment drivers' vision may be impaired, and the visibility of low-contrast objects may be diminished to the extent that some objects might even be rendered invisible. The drivers' perception in traffic flow might be negatively affected by sunlight which leads to a potential increase in traffic collision. Therefore, the need to ameliorate traffic safety and improve traffic operations' efficiency is addressed in this study by developing a generic modeling approach to identify hazardous driving locations on any given network.

1.1 Research Motivation

Traditional strategies, such as expanding tree coverage as well as changing driving direction, have been applied to reduce the negative impact of bright sunlight and to provide adequate stopping sight distance (Waldron, 2013). However, physical countermeasures are challenging to implement because receiving sunlight on any point on the earth depends on the daytime and the season. Instead, innovative technologies in communication and control system could be exploited to minimize the visual disturbance caused by sun glare (Li, & Yu, 2016).

Intelligent Transportation Systems (ITS) provide a transportation modeling and deployment framework employing advances in information and communication technology to ameliorate transportation operations' safety and efficiency. The Advanced Driver-Assistance System (ADAS) is one example of such automated user assistant services that have been tested in various forms of automated vehicles over the past several decades. The benefits of ITS systems such as ADAS deployed in AVs are considered as a specific solution, promising to change the surface transportation sector fundamentally. Developing AVs capabilities may alleviate potential safety issues related to vehicles interactions among themselves and with other road users, especially on urban roadways during exposure to direct sun glare. Therefore, this thesis focuses on developing

a predictive control framework for the current transportation networks applicable to both highways and arterial roads.

1.2 Research Objectives

The main objective of this thesis is to develop a methodology to dynamically identify the locations of potential vision impairment and to integrate into a framework of cooperative warning and control system for AVs. Thus, in order to consider a common vehicle interaction, a two-lane segment of roadway section is selected when vehicles approach to the location with sun exposure. A control goal is defined to facilitate the stopping maneuvers, to avoid traffic flow disruption, and to comply with safety constraints. In this regard, lack of adequate perception-reaction time and criteria to determine safe-stop distance are investigated for the cases of safe-stop maneuver. Every driver chooses their individual speed according to their perception of the visibility and their comfort level. In this study, a traffic simulation environment is used to evaluate different perception of visibility on a federal-provincial highway under sun exposure.

This study also analyzes drivers' speed behavior during sun exposure to guide the calibration of microscopic simulation. The study also investigates how different autonomy levels of AVs improve traffic operations and safety performance on impaired visibility areas. The result of the study includes evaluation of traffic operation and safety assessment for different types of AVs under sun exposure. This research aims to aggregate the proposed methodologies in a single package CASUS (Collision Avoidance System Under Sun-glare) for the AVs with failure sensory system to improve traffic safety by providing optimal-safe speed.

1.3 Scope and Limitations

This dissertation mainly focuses on the multi-level framework, where a network wide algorithm is developed to identify the location and time of sun glare occurrence. Because the nature of vision impairment could comprise different combinations of AVs and downstream obstacles, different safe-stop scenarios can occur. When a driver approaches the vision impairment location, the model will be adjusted based on the changes in driving behaviors. This change may result from the inaccurate estimation of object behaviors and motions, unexpected disturbances due to types of

vision impairment, and delays caused by human-driven vehicles' reaction time. The following assumptions and limitations pertain to the study:

- Sun glare detection is limited to a sun cone of ±15º for the azimuth.
- The findings are limited to freeways based on the available data for model calibration.
- The irradiance of the Sun light on the detector is considered for the receiving light on drivers 'eyes.
- Driving behaviors are emphasized on the longitudinal direction of vehicle movement (i.e., acceleration and deceleration) when restricted by a downstream leading vehicle.
- The proof-of-concept analysis for implementation in microsimulation is conducted using the Wiedemann car-following model in PTV VISSIM microsimulation software.
- Traction and grip of tires and pavement condition is considered constant for the dry pavement on the moderate temperature of fine grade pavement.

The development of sun glare detection algorithm and evaluation of driving behavior in CASUS are illustrated in Figure 1.

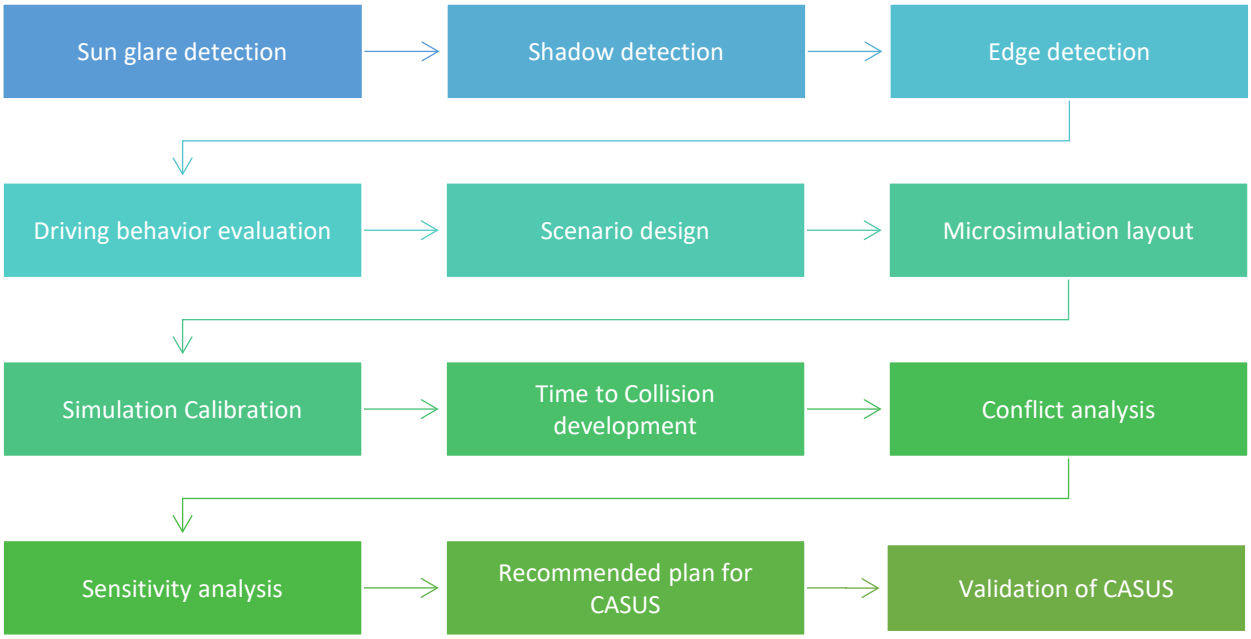


Figure 1 . Core framework of driving parameters

In the first step, a glare detection scheme is employed to locate vision impairment on any given road network and to optimize the control inputs for AVs. In the second step, actual data of traffic flow is analyzed to determine changes in driving behavior and find a significant correlation between driving behavior and visibility change. Careful considerations on the selection of the study area size, data collection requirements, and impact of vision impairment should be made during the base model development. In the third step, driving behavior parameters of car-following model and conflict zones will be calibrated for the base model. In the fourth step, all outputs of the simulated model need to be analyzed for the sensitivity of changes in traffic flow parameters, vehicle compositions, and transition to different level of AVs, and effective speed on safety enhancement. Finally, the development of CASUS will be expected to identify the best action in response to vision impairment of drivers for the case study.

1.4 Research Contributions

There is a lack of research to explain the use of AVs as an intelligent solution for improvement of road safety during vision impairment. The main contribution of this PhD study is to provide knowledge of different vision impairment spots (i.e., sun glare and contrasting shadow) for urban roadways and freeways. It is achieved by considering factors, such as the angle of the sun, the time of day, the geography, the geometric design of the road and the physical environment in the evaluation of sun glare risk.

A combination of different control systems under CASUS is simulated with the psychophysical car-following model to provide an integrated microsimulation environment with real-time AVs control. This model contributes to the simulation of other types of uncertain incidents on the roadway and emulating the corresponding decision-making process on certain types of vehicles in the network.

This study also contributes to a better explanation of developing an advisory system for AVs to mitigate potential conflicts in vision impairment. Developing a new assessment method for conflict analysis and proposing new thresholds for analyzing vehicle trajectories is a novel methodology to evaluate potential conflicts by using AVs in mixed traffic flow.

1.5 Research Layout

This dissertation is structured as follows:

Chapter 2 presents the relevant literature review, including detection of vision impairment, use of microscopic modeling, and application of ADAS for the improvement of road safety to learn about the impact of vision impairment on driving behavior.

Chapter 3 describes the novel methodology developed in this study to model critical points of vision impairment including sun exposure and contrasting shadows on any type of roadways.

Chapter 4 explores the development of a psychophysical car-following model when approaching sun glare exposed roads for the various levels of vehicle automation. The developed models are integrated into the proposed road safety model.

Chapter 5 contains the development of an advisory system for AVs to provide optimal traffic safety and operation in the sun-glare detection framework.

Chapter 6 evaluates the proposed methodologies on Montreal Island to validate the methodology.

Chapter 7 provides conclusions and future research directions.

Chapter 2 LITERATURE REVIEW

This chapter presents a comprehensive literature review intended to provide context for this dissertation. The concepts of this dissertation research are founded on decades of multi-disciplinary research that have molded microsimulation modeling concepts and practices.

The literature review includes seven topics:

1. The first section reviews driving behavior while visually impaired and provides background information for the role of driver vision on driver performance under different visibility reduction situations. Then, astronomical models—notably, relevant glare detection models—are briefly reviewed to determine the best-fit options for the framework. Moreover, driving behavior under vision impairment is explained.
2. The system modeling and simulation section provides an overview of traffic stream characteristics and microsimulations for modeling driving behaviors. Within this section, various car-following models are introduced in the state of research and are compared in terms of accuracy and success for modeling vision impairment. Various factors that influence the accurate and realistic portrayal of driving behavior are also introduced into the microsimulation.
3. The autonomous vehicles section begins with clear definitions for different automated vehicle levels and provides the latest models for autonomous vehicles within a microsimulation environment.
4. The traffic safety performance section covers analytical methods for safety analysis and the classification of conflict types in different road environments.
5. The advanced driver-assistance systems section includes the development of driver assistant systems in autonomous vehicles with a focus on the functionality of sensors and driver maneuvers in urban roadways. Also, a model-based vehicle control scheme is briefly introduced.
6. The advanced collision warning system section introduces the essence of driver-vehicle interaction and the timing of warning messages.

7. The intelligent braking assistance section provides a background analogy for visual stopping distance and human perception. An overview of intelligent braking assistance systems in the state of research is considered for vision impairment system benefits.

2.1 Driving behavior in vision impairment

Low visibility and challenging weather conditions cause numerous traffic accidents each year (Pisano et al., 2008). A Canadian study (Andrey et al., 2001) on weather hazards indicated that 16% of fatal collisions and 18.5% of personal injury collisions occurred under adverse weather conditions (i.e., rain, snow, hail, and fog). The study also indicated that over 25% of property-damage-only collisions occurred during low visibility weather. The remaining crashes occurred in the presence of no precipitation, but a few studies have concentrated on crashes under clear skies. Vision is the most influential input channel for maneuvering a vehicle (Tonnis et al., 2006). Shining intensive lights into drivers' eyes was the central issue of prior studies as a probable cause of accidents. A study found that sun glare (i.e., a continuous source of extreme brightness corresponding to sunlight) was the critical cause of vision impairment for drivers' during the day (Ho, Ghanbari, & Diver, 2011).

2.1.1 *Vision Impairment*

Vision works through the transformation of light into energy by eye receptors (Browman & Goldstein, 1989). Visible light is limited to the electromagnetic wavelengths of 380 to 720 nm. Most perceived light between these wavelengths is reflected into the eyes from objects in the environment. The visual system resolves fine detail visual acuity by measuring high-contrast stimuli. People have different contrast sensitivities and are able to detect objects clearly within their visual acuity. Contrast sensitivity tests show that when the contrast between an object and its background is high, the object is seen to be smaller and vice versa (Elliott, Whitaker, & MacVeigh, 1990).

The movement of either an object or the observer decreases visual acuity (e.g., looking at a sign on the side of the road) and is called dynamic acuity. Dynamic acuity deteriorates with increased relative motion (Burg, 1966). A study (Chrysler, Danielson, & Kirby, 1996) showed that older persons can detect a 42-inch-tall object in a closer distance (230 ft) than younger persons (360 ft). They also simulated low contrast situations (adverse weather conditions) and found that the

detection distance of the static object was reduced by 51% when compared to clear weather; however, the impact of high contrast levels (extreme sources of light) was not considered.

The occurrence of sun glare is directly related to the position of the sun towards an observer's sightline. The trajectory of the sun around the earth changes based on the value of the declination angle. Several astronomical algorithms have been developed to increase the efficiency of recognizing the sun's actual position. Many studies have aimed to decrease computational effort. One of the early modern approaches was proposed by Spencer (1971), followed by Michalsky (1988), who made important updates and improved probable errors. NOAA (National Oceanic & Atmospheric Administration) sun position is the most recent algorithm proposed by the Earth System Research Laboratory at the National Oceanic & Atmospheric Administration to enhance sun position accuracy.

The threat of sun glare on the visibility of drivers has been investigated in a case study and a novel method to estimate the location of sun glare on the roadway segment was proposed (Churchill, Tripodis, & Lovell, 2012). The authors found dazzling as the most severe and harmful impact of sunlight, which occurs when an intense sun ray shines directly into the center of a driver's eyes while the eyes try to recognize an object in the same alignment as the sunray. Dazzling sunlight has a destructive impact on central vision and causes a permanent blind spot in the driver's eyes (Reidenbach, 2009). Dazzling causes discrepancies in received visual data and may hinder the recognition of traffic states such as the lateral movement of surrounding vehicles or road users (e.g., lane changing and turning) and longitudinal change of leading vehicles (e.g., braking lights and red-light violations). Drivers need to be concerned with all surrounding objects for safe driving. Consequences of vision deficiencies, particularly dazzle effects, are extreme and inescapable upon transition locations where a driver moves between a road segment under shadow to a segment that is under sun glare.

The reverse of the dazzle effect has been identified as blindness (e.g., the black hole effect at a tunnel entrance). In comparison to dazzle conditions, driving a vehicle in a longitudinal alignment of the road when blind does not harm a driver' eyes but causes an obvious threat for other road users. Glare-related crashes occur around the world. In Japan alone, 10,352 sun glare-related incidents were reported between 2007 and 2011 (Hagita & Mori, 2014). In Great Britain, sun glare

has been involved in 36 deaths as well as nearly 3000 annual traffic incidents as they temporarily blind drivers with sunlight through the windscreen (DTGMH, 2012).

2.1.2 *Driving Behavior*

Vision impairment has both physical and psychological impacts on driving behaviors. Driving under temporary blindness has a physical impact on driving behavior. This physical effect increases the chance of a collision by reducing a driver's perception-reaction time. On the other hand, the uncertainty of driving under sun glare condition causes a psychological impact, resulting in a delay in traffic operations (e.g., indeterminate shockwave and abrupt deceleration).

A study investigated the impact of adverse weather (rain and snow) on traffic speed in a freeway case study in Canada (Ibrahim & Hall, 1994). Their research showed a reduction in speed between 2 km/h and 38 km/h for light rainfall and heavy snowfall, respectively. Another study (Hawkins, 1988) in the UK also showed a similar reduction in speed in adverse weather conditions due to impaired visibility. The Highway Capacity Manual (FHWA, 2000) also suggests that free-flow speeds decreased by 2 to 14% and 20 to 35% due to light rain and heavy snow conditions, respectively. The manual and corresponding studies failed to consider the psychological impact of road environment (i.e., clear skies) on driving behaviors and only focused on the physical impact of adverse weather conditions.

While some studies (Fry, 1955; Hammond et al., 2001; Mace et al., 2001; Stringham, Garcia et al., 2011; Theeuwes et al., 2002) supported the finding that glare causes discomfort and has a negative impact on visual performance, few works have investigated the effect of glare on driving behavior and road safety. Shepard (1996) investigated the effect of low visibility conditions on driving, such as adverse weather. Also, vision impairment for drivers and subsequent changes in driver behavior is a potential factor in congestion. Goodwin (2002) investigated the impact of sun glare on effectiveness and found that lower visibility increases travel time and delay substantially by +12%.

2.2 General considerations about modelling and simulation of systems

The scope and complexity of a particular system depends on the objective of a project (Law, 2015). Systems need to identify the optimal collection of necessary variables under different conditions at a particular time. Experimenting the actual implementation of a system is the best mechanism

for forecasting changes; however, experiments are often costly and disruptive. Thus, a system is designed using a model with a scaled representation used for the actual system (Law, 2015).

A model can be in physical form to represent larger systems at a scale that are tangible, or in mathematical form to describe logical and quantitative relationships as well as identify system reactions under different conditions. The mathematical model includes two evaluating procedures: analytic solution and simulation (Figure 2). Analytic solutions provide an exact global solution by obtaining a simplistic closed-form model and simulations evaluate highly complex models. Simulations are defined as numerical evaluation of a model that estimates the desired true characteristics of a model. The microsimulation environment is a known way to evaluate vehicle interactions as a collection of entities and in this study was used to understand the safety implications of vision impairment.

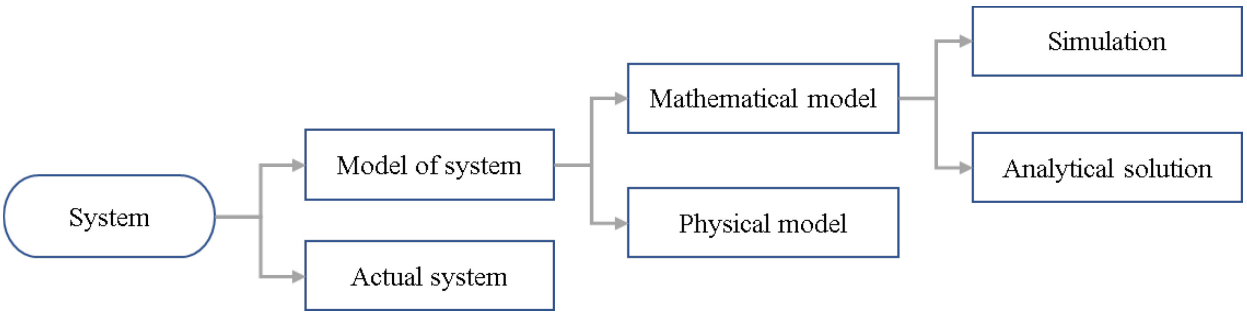


Figure 2. Study a system
(Law, 2015)

Simulations provide a model for evaluating a system's performance and predicting system responses under different stochastic demands and constraints. The simulation modeling process helps decision-makers alleviate uncertainty, optimize system operations, and optimize resource allocation with every innovation and design (Maria, 1997). Advanced computing systems provide a systematic process for repetitively performing a simulation to replicate different system states and alternatives. However, simulation values are directly related to a modeler's domain knowledge and ability to correctly apply constraints that represent realistic conditions. Significant changes in any traffic flow characteristic can result in a huge change in model outcomes.

Traffic simulation modeling facilitates the prediction of traffic stream characteristics under different constraints. Due to the high complexity of traffic streams and their dependency on several continuously evolving factors—including but not limited to roadway geometry, traffic control strategies, travel demand, environmental conditions, and driving behaviors—simulation modeling

presents unique opportunities to accommodate the stochastic nature of these factors and forecast the influence of different combinations and severe factor fluctuations on traffic streams (Lieberman & Rathi, 2001).

2.2.1 *Traffic Stream Characteristics*

Traffic flow is described by three fundamental variables: flow, speed, and density (Hall, 2001). Greenshields derived a linear relationship between speed and density when evaluating traffic flow (Greenshields, Bibbins, Channing, & Miller, 1935). In 1967, Drake et al. produced evidence that the relationship between these traffic stream characteristics were very complex (Drake, 1967). Their study indicated that while some models could better produce the speed–flow–density relationship seen in empirical data; none exhibited a perfect fit. This finding raised two questions, the first related to the quantity and derivation of the empirical data used to describe these fundamental relationships, and the second related to the car-following models used to create synthetic data points (Hall, 2001). The crux of these raised questions is the ability or inability of the input data to adequately describe realistic traffic conditions.

Driving behavior in car-following models is a crucial element of traffic flow theory. They define the longitudinal motion of vehicles in a shared lane to accelerate and decelerate to a desired speed (Krauß, Wagner, & Gawron, 1997). Car-following models have evolved from basic to sophisticated over the past half-century due to their utilization as a key microsimulation software process (Brackstone & McDonald, 1999; Olstam & Tapani, 2004; Saifuzzaman & Zheng, 2014; Toledo, 2007). Car-following models were introduced in the early 1950s (Pipes, 1953; Reuschel, 1950) and they have been calibrated and improved over times. The basic car-following model considers fundamental rules and learns to drive using a series of Laplace transformations that predict vehicle velocity within a time gap (Pipes, 1953).

In 1958, the General Motors (GM) model was developed as a simple linear model that used vehicle acceleration as a function of relative velocity between following and leading vehicles (Chandler, Herman, & Montroll, 1958). The model worked under the assumption that a following vehicle reacts to changes in the steady state of the leading vehicle with the smallest possible change. Another study developed the GM model into a non-linear relationship between relative speed and following distance to predict vehicle acceleration (Gazis, Herman, & Rothery, 1961). The GM model was developed in other studies, including the incorporation of behaviors that modeled more

than one downstream vehicle (Helly 1959), relative speed, distance (Edie, Gazis, Helly, Herman, & Rothery, 1965; Hanken & Rockwell, 1967; Rockwell, Ernst, & Hanken, 1968), and deviation from desired time headways for different drivers and conditions (Van Winsum, 1999). A major limitation of the GM models is the sole use of headway for determining the sensitivity of stressful drivers (Papadimitriou & Choudhury, 2017).

Some car-following models have been widely used and integrated into microsimulation tools, such as the Gipps model (Gipps, 1981) in AIMSUN, Fritzsche (Fritzsche & Ag, 1994) in PARAMICS, and Wiedemann (Rainer Wiedemann, 1974) in VISSIM. The Gipps model is a first safe-distance model that considers free-flow speed and car-following driving (Gipps, 1981; Toledo, 2007). In the Gipps model the driver chooses the lower speed between the two driving modes. Driving under unconstrained traffic situations allows one to travel at the desired speed, while in congested conditions speed is estimated and limited for safety. The dilemma of the Gipps model is that it does not consider the psychological impact of driving and only considers a rigid driving mode.

Another car-following model, Fritzsche, was developed to consider the psycho-physical impact of drivers through follower vehicle reactions based on the difference in speed and spacing under five regimes and six thresholds. However, the Fritzsche model assumed maximum acceleration to be a given constant, which is not realistic (Fritzsche & Ag, 1994).

Wiedemann proposed another psychophysical model for the microsimulation tool VISSIM. The model developed a full decision-action-point car-following model, which is essentially a psychophysical model that includes certain thresholds for relative speed and distance behind a lagging vehicle for taking action (Rainer Wiedemann, 1974). When a vehicle approaches a slower leading vehicle, the driver recognizes some action points for conscious reaction. Wiedemann included four regimes for discrete driving: free flow, approaching a slower leader vehicle, following in near steady-state equilibrium, and critical braking action. Regimes worked based on the use of different acceleration functions based on velocity and average speed for following vehicle and five boundaries' regimes [1-5] according to having zeroes of the following function:

$$AX = L_{n-1} + AX_{add} + RND1 \times AX_{mult} \quad [1]$$

$$ABX = AX + BX \quad [2]$$

$$SDX = AX + EX \times BX \quad [3]$$

$$SDV = \left(\frac{\Delta x - L_{n-1} - AX}{CX}\right)^2 \quad [4]$$

$$OPDV = CLDV \times (-OPDV_{add} - OPDV_{mult} \times NRND) \quad [5]$$

$$BX = (BX_{add} + BX_{mult} \times RND1)\sqrt{u} \quad [6]$$

$$EX = EX_{add} + EX_{mult} \times (NRND - RND2) \quad [7]$$

where,

AX is the desired distance between vehicles in standstill situations,

L_{n-1} is the physical length of the lead vehicle,

ABX represents the desired minimum following distance at low-speed differences between vehicles,

SDX is the perception threshold for modeling the maximum following distance between 1.5 to 2.5 times ABX;

SDV is the approaching point where a driver has awareness, they are approaching a slower leader;

$CLDV$ is the reduction of speed differences at short and decreasing distances; and

$OPDV$ is the increase in speed difference when a driver realize they are traveling at a slower speed than the leader

$AX_{add}, AX_{mult}, BX_{add}, BX_{mult}, EX_{add}, EX_{mult}, OPDV_{add}, OPDV_{mult}, RND1$,

$RND2, RND3, RND4, NRND$ are additional dependent parameters for the model. CX is assumed to be 40.

The following regime constituted the thresholds for ABX, SDX, OPDV, and SDV based on the assigned acceleration a_{null} [8], where $BNULL_{mult}$ is a calibration parameter. $a_{null}$is a positive value when passing either OPDV or SDX and is negative when passing either SDV or SDX. The maximum acceleration a_{max} occurs during free flow speed and after reaching the desired speed a_{null} is used. Equation [9] represents the computation of maximum acceleration based on the difference in current speed u and the maximum desired speed u_{max}, including model constants

(i.e., BMAX and FAKTORV). Wiedemann specified a linear model for maximum acceleration [10] and deceleration [11] based on current speed.

$$a_{null} = BNULL_{mult} \times (RND4 + NRND) \quad [8]$$

$$a_{max} = BMAX_{mult} \times (u_{max} - u \times FAKTORV) \quad [9]$$

$$Maximum\ acceleration = 3.5 - \frac{3.5}{40}u \quad [10]$$

$$Maximum\ deceleration = -20 + \frac{1.5}{60}u \quad [11]$$

Wiedemann developed his model in 1999 to reduce the level of uncertainty attributed to random variables and proposed the W99 model for freeways (H. Wiedemann & Stüve, 1996). A decade later, the W99 car-following model was developed to include logic for autonomous vehicles. Figure 3 shows a couple of driver interactions in the W99 where a vehicle approaches another vehicle and how to a driver should react when following the lead vehicle. The subject driver begins to decelerate until an individual threshold, which is a function of acceptable speed difference and spacing, is reached. The driver then maintains a speed at or below the current speed of the leader until reaching other driving behavior thresholds for different maneuvers (Wiedemann, 1974).

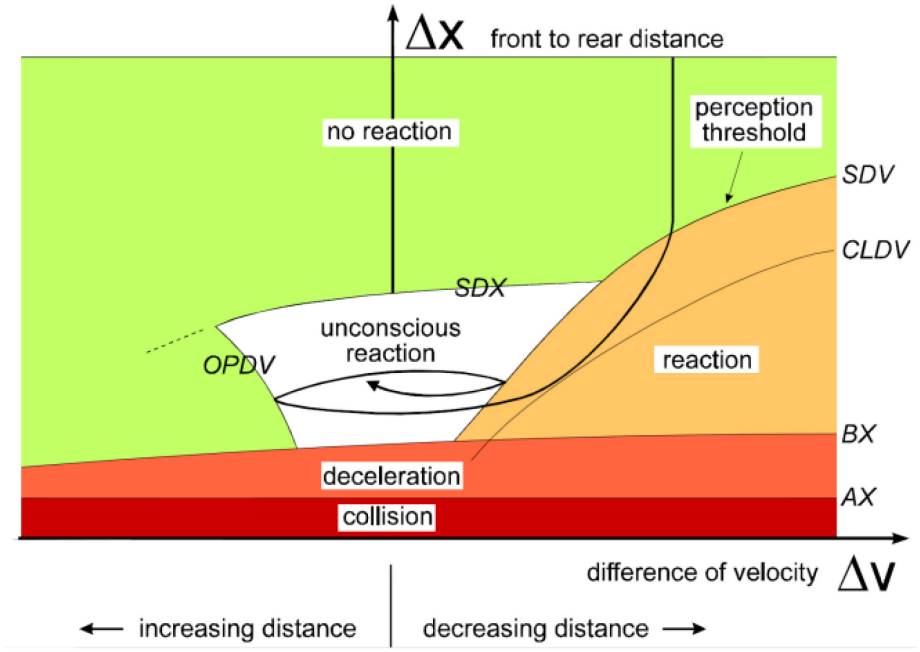


Figure 3. Wiedemann 99 car-following model (VISSIM 11.0 Manual)

Wiedemann 99 model calculates the CC parameters for the adjustment of regions outline in Figure 3.

$$AX = L + CC0 \quad [12]$$

$$BX = AX + v \times CC1 \quad [13]$$

where,

AX is the collision threshold $[meter]$

BX is the deceleration threshold $[meter]$

L is the length of the lead vehicle $[meter]$

v is the velocity of the vehicle which has smaller speed in each pair of vehicle interaction $[m/s]$

$CC0$ is the average standstill distance between vehicles $[meter]$

$CC1$ is the headway time for measuring the average following distance $[seconds]$

AX and BX are the most influential parameters of W99 in following procedure. These parameters included the minimum standstill distance and headway which have the importance influence on occurrence of collisions. Other thresholds of the Wiedemann 99 model were defined in the following.

$$CLDV = \frac{CC6}{17000} \times (\Delta x - L)^2 - CC4 \quad [14]$$

$$SDV = -\frac{\Delta x - BX - CC2}{CC3} - CC4 \quad [15]$$

$$SDX = BX + CC2 \quad [16]$$

$$OPDV = -\frac{CC6}{17000} \times (\Delta x - L)^2 - \delta \times CC5 \quad [17]$$

where,

Δx is the spacing between the subject vehicle and its leader $[meter]$

δ is a dummy variable that is 1 when the speed is greater than CC5 and 0 otherwise,

$CC2$ is the allowed safe following distance to surpass before the following vehicle accelerates within maximum link speeds $[meter]$

$CC3$ is the time taken to reach the safe following distance when leader is slower [$seconds$]

$CC4$ is the sensitivity of the vehicle to the negative change of leader's speed [m/s]

$CC5$ is the sensitivity of the vehicle to the positive change of leader's speed [m/s]

$CC6$ is the influence of distance on speed oscillations [$m.s$]

One of the challenges with a psychophysical model rests on the distributions of thresholds. Continuous field measurements of different traffic conditions are required to calibrate the model in a realistic manner. Thresholds for the Wiedemann car-following model were designed based on limited available data (i.e., the driving behavior of people in Karlsruhe), and must be calibrated for field measurements.

The intention for identifying the best car-following model in different studies was to simulate the real stream of traffic and predict traffic operations. In each of the above studies, the car-following models was incorporated into a microsimulation software package and used to predict network-wide traffic flow. Among these car-following models, only the Wiedemann (R Wiedemann, 1994) model stimulated a full action psychophysical model for driving behavior (free flow, approaching, following, and emergency braking to represent the effect of vision impairment in the simulation environment).

2.2.2 *Traffic Simulation Modelling*

In traffic simulation modeling, the system state represents a "scenario" that includes transportation network configuration and traffic demand. Traffic simulation models are typically used to evaluate system performance in terms of known traffic stream characteristics, such as alternative treatments, new designs, safety assessments, etc. (Lieberman & Rathi, 2001). Three common models' categories are: macroscopic, mesoscopic, and microscopic. Macroscopic modeling is used in transportation planning for network-wide modeling and involves only a traffic stream's fundamental parameters. Contrarily, microscopic modeling is used in traffic operations to predict detailed vehicle interactions with infrastructure and coexisting traffic. Mesoscopic modeling is the middle ground between the macro and micro models and it considers high level vehicle interactions through derived traffic stream relationships (Lieberman & Rathi, 2001).

Among the three models, microscopic is evaluated through simulations and reflected realistic traffic flow conditions, resulting in different outputs from randomly fluctuating inputs. The following subsection briefly introduces the background for microsimulation modeling due to the emphasis of this dissertation research, specifically its concern with driving behaviors.

Different theories for driving behaviors have been applied using microsimulation tools to predict the behavior of individual drivers. Driving models were traditionally developed based on traffic flow theories and more recently they have been based on the psychological nature of drivers (Brackstone & McDonald, 1999; Toledo, 2007). Each vehicle in a system has a unique speed, acceleration, and lateral movement based on the used car-following model (Maerivoet & Moor, 2008). Therefore, car-following models play a significant role in predicting roadway capacity and traffic flow patterns (Ossen & Hoogendoorn, 2011; Rothery & Roy, 2001). The successful calibration of car-following models is a fundamental step in achieving realistic microsimulation models.

In recent years, microscopic traffic simulation tools have become progressively more common when investigating traffic operations on roadways. Traffic simulation models have been used in research on traffic planning and for the development of intelligent traffic systems with different geometries, traffic demands, vehicle routes, and driver behaviors. Simulation environments provide safer, less expensive, and faster model development in terms of transportation operation and traffic management. The psychophysical car-following model of Wiedemann incorporated the first use of inter-deriver heterogeneity for driving behavior to replicate realistic traffic conditions, which was then applied using VISSIM microsimulation software.

VISSIM is a microscopic traffic-simulation software widely used to study homogeneous traffic conditions (Siddharth & Ramadurai, 2013). The psychophysical model comprises both physical activities (i.e., perception-reaction threshold and unconscious car following) and psychological behavior (e.g., accelerating and decelerating). The Wiedemann model considers longitudinal vehicle movement and has a rule-based algorithm for lateral movements (Vashitz et. al., 2008).

Microsimulations are generally stochastic and model a corridor or at a project-specific scale to represent unpredictable traffic flow and driving behavior. Microsimulation software provides an opportunity for users to interpret driving behaviors to evaluate heterogeneous driving behavior and alternative traffic control designs (Kim & Mahmassani, 2011).

2.2.3 *Accuracy of Driving Behavior*

The model calibration and validation stage is essential to a simulation modeling accurate field conditions (Ahmed et al., 2018). Calibration adjusts model parameters to accurately mirror prevailing road network conditions. The main adjustable model parameters were car-following behavior, gap acceptance, vehicle speed distributions, lane changes, alternative routes, and vehicle acceleration distributions. Validation is defined as the comparison of simulated model results with actual data to test simulation model accuracy. The model validation stage detects appropriate parameter settings that produce outputs close to the measured field results. The validated parameters are maintained in the base scenario to represent a roadway sections constant driving behavior and operational characteristics.

The calibration of microsimulation traffic models is a prerequisite for effective results when developing the study area. Three strategies have been introduced for model calibration: adjustment of specified parameters, specified route choices, and traffic operations. Car-following model calibrations need to use disaggregate trajectory data to identify driving behavior and traffic flow. However, further more detailed data and the optimization of input parameters necessitates the interpretation of calibration results (Ossen & Hoogendoorn, 2008). The complexity of calibrating car-following modes using trajectory-level data has led to decision points that dictate the formulation of optimization problems based on the choice of model parameters, performance measurements, goodness of fit functions, and optimization algorithms (Punzo, Ciuffo, & Montanino, 2012).

A study applied an optimization algorithm to test the impact of three effectiveness measures, headway, relative velocity, and spacing, on some car-following models (Punzo & Simonelli, 2005). The authors noted that spacing is the most reliable parameter for measuring performance. Another study formulated an optimization algorithm to evaluate the maximum parameter likelihood function for the simulation calibration of multiple trajectories (Hoogendoorn & Hoogendoorn, 2010; Ossen & Hoogendoorn, 2005). The authors used three different objective functions to penalize relative, absolute, and mixed errors. They noted that the mixed-function errors showed the most optimal calibration results. The use of this optimization method concerning spacing reduced errors in average velocity.

Another study recommended using Mean Absolute Error (MAE) to adjust parameters between real-world and simulated model conditions (Dowling, Skabardonis, & Alexiadis, 2004). Creating an objective function based on MAE provides an iteration process for finding the best answer for a simulation models sensitive parameter. Several systematic processes have been proposed for minimizing MAE using various optimization methods (Zhizhou, Jian, & Xiaoguang, 2005). Common optimization techniques for calibrating traffic simulations are multi-threading (Aghabayk, Sarvi, Young, & Kautzsch, 2013), genetic algorithms (Zhizhou et al., 2005), Mont Carlo (Henclewood, Suh, Rodgers, Fujimoto, & Hunter, 2017), simulated annealing (Lidbe, Hainen, & Jones, 2017), and Particle swarm optimization (M. Karimi, Miriestahbanati, Esmaeeli, & Alecsandru, 2019).

Hourdakis (2003) presented three-stage statistical procedures for the calibration and validation of simulation models: 1) Volume-based calibration; 2) Speed-based calibration; and 3) Objective-based calibration (Hourdakis et al., 2003). Their study was conducted using the PARAMICS simulation software platform and did not address specific modifications of model parameters. Afterward, many studies have used these methods to calibrate microsimulation models, but few of them focused on VISSIM calibration or specific parameters.

In a recent study, Karimi et al. (2019) investigated the function of various optimization algorithms on the VISSIM model and proposed an optimization method to calibrate and validate VISSIM parameters, particularly for freeway models. They investigated the amendment of driving parameters using genetic algorithm, whale optimization algorithm, particle swarm optimization, and multi-objective particle swarm optimization in a single step. Their results showed that the PSO algorithm had the best fit when calibrating a single objective (i.e., a one-lane roadway) in the simulation model (Mohammad Karimi et al., 2019).

A calibrated car-following model accuracy is its ability to reproduce real-world conditions using sensitivity analysis or synthetic data. The cross-validation strategy has been used in many studies, which is the use of different datasets than those used to validate the car-following model (Kesting, Treiber, & Helbing, 2009). The behavior predicted by the validation strategy replicates expected global outputs in microsimulation modeling applications. The cross-validation procedure has also been used to interpret a calibrated models with a heterogeneous behavior under different driving

conditions. Calibration interpretation uses the same driver for different conditions and has been used in multiple studies.

2.3 Autonomous Vehicles

The design and implementation Intelligent Transportation Systems has become a major focus for government transportation agencies, for example the USDOT (Barbaresso et al., 2014). Innovative transportation technologies such advanced warning messages (e.g., messages from a drivers' smart advisory system), vehicle advisory systems (e.g., intelligent braking systems), and Variable Message Signs (VMS) are needed to eliminate visual disturbances caused by glare. Many studies have previously investigated traffic warning system design, the use advance-warning signals for approaching intersections, or impending signal changes (Schultz & Talbot, 2009). Therefore, reacting to the dynamic conditions of sun glare needs the use of in-vehicle technologies.

2.3.1 *Automation Level*

Several automotive industries have invested funds in developing automated vehicle technologies to improve comfort, traffic flow, and road user safety (Friedrich, 2016). Autonomous Vehicles (AVs) are highly reliable when controlling electrical systems that improve performance and safety by enabling agile actions, such as antilock braking and electronic stability control (Grimm, 2008; Macario et al., 2009; Mössinger, 2010). Several studies have investigated how AVs affect driving behavior at a microscopic-scale, such as acceleration/deceleration rates, reaction time, and capacity (Hamdar, Qin, & Talebpour, 2016; Zhao & Sun, 2013; De Charette et al., 2013; Treiber, Kesting, & Helbing, 2007; Van Arem, Van Driel, & Visser, 2006;). The use of automated technologies in AVs comprises a variety of sensors and system control mechanisms. Sensors provide a vehicle's inputs and information on surrounding conditions, which the control system uses to perform controlled movements and to determine next position on its trajectory.

An increase in vehicle automation level reduces human driver responsibility and turns the role of the driver into a supervisor. The National Highway Traffic Safety Administration (NHTSA) provided a common terminology for AVs and categorized the autonomy levels of vehicles into 5 levels (Committee, 2014; Mössinger et al., 2013; Ranney, Garrott, & Goodman, 2001). Figure 4 illustrates the NHTSA classification for vehicle automation levels. Level zero represents conventional vehicles in which the driver is responsible for full vehicle control. Level one utilizes

one or two rudimentary automated functions (e.g., electronic stability control and pre-charged braking systems). Level two comprises at least two primary control functions that help the driver control the vehicle (e.g., adaptive cruise control and lane-centering systems). Level three supports full control with limited self-driving automation under specific traffic and environmental conditions, such as Google's self-driving car (Poczter & Jankovic, 2014). Level four vehicles can undertake full control of safety-critical driving functions while monitoring roadway conditions for an entire trip.

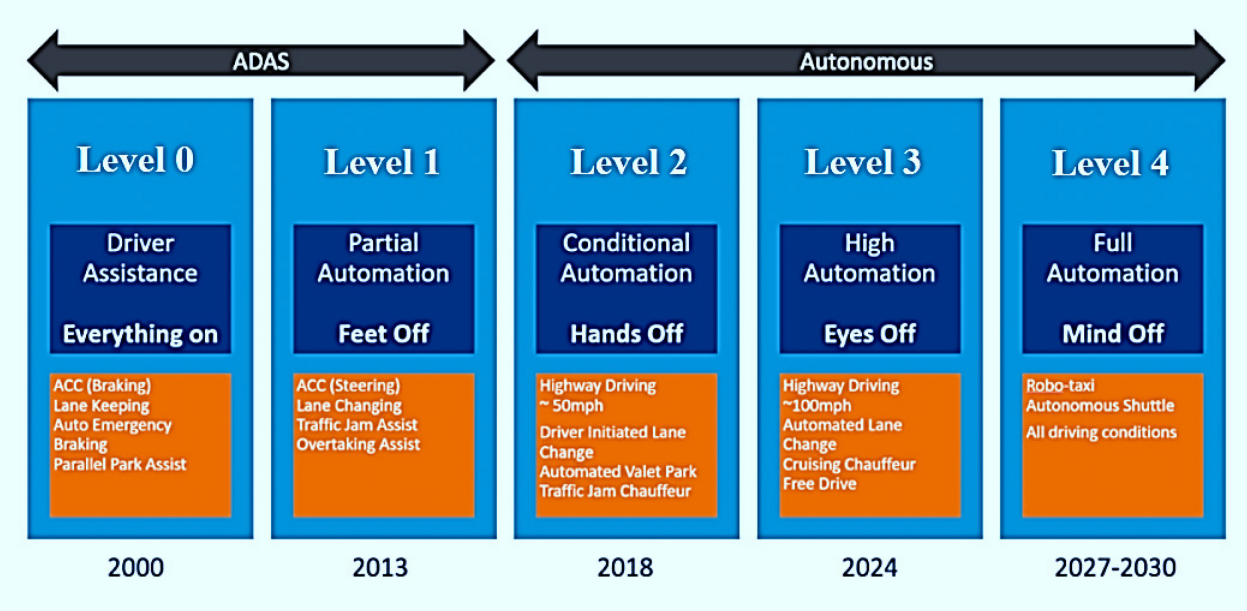


Figure 4. Levels of vehicle autonomy by NHTSA (R. N. Charette, 2009)

2.3.2 ***On-vehicle Sensors***

The Pyramid blocks for automated driving introduced a procedure for all autonomous vehicles, including control systems, sensing, and decision-making levels (Figure 5). The sensing system is the keynote of any autonomous vehicle, consisting of cameras, radars, and lidars. Sensors gather data at various resolutions and have a wide range of hardware and analysis costs. Sensors are mostly used to detect vehicle interactions with either obstacles or other road users. In order to develop the longitudinal acceleration model for autonomous vehicles, Talebpour & Mahmassani (2016) utilized sensors to obtain a safe distance for a linear car-following model.

ITS technologies employ different sensors to develop applications or controllers and improve vehicle performance, road safety, and comfort. On-vehicle sensors have been sorted into six categories based on the application of the sensor (Abdelhamid, Hassanein, & Takahara, 2014).

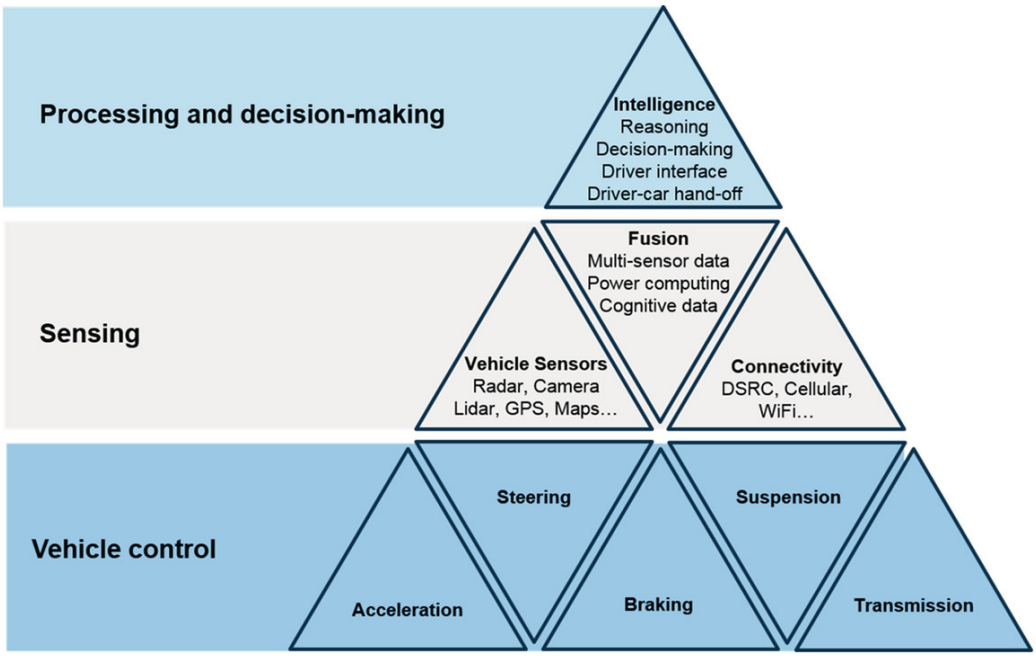


Figure 5. Building blocks of automated driving
(Watzenig & Horn, 2017)

The first category of sensors accounts for basic safety systems and focuses on detecting accidents and events (e.g., speed sensors, cameras, RADAR, and ultrasonic). The second category has a diagnostic role and provides real-time information about vehicle performance (e.g., gyroscope, temperature, and consumption). The third category monitors traffic conditions to improve traffic management systems. These sensors are combined with safety sensors, and they analyze data differently. The fourth category combines common sensors and diagnostic sensors but are designed to support driver comfort and convenience. The fifth category monitors environmental conditions to warn drivers of adverse weather (e.g., temperature, pressure, and distance sensors). The sixth category monitors driver health and behavior (e.g., camera, thermistors, and electrocardiogram). Figure 6 illustrates the approximate location of some safety and diagnostic sensors in a vehicle

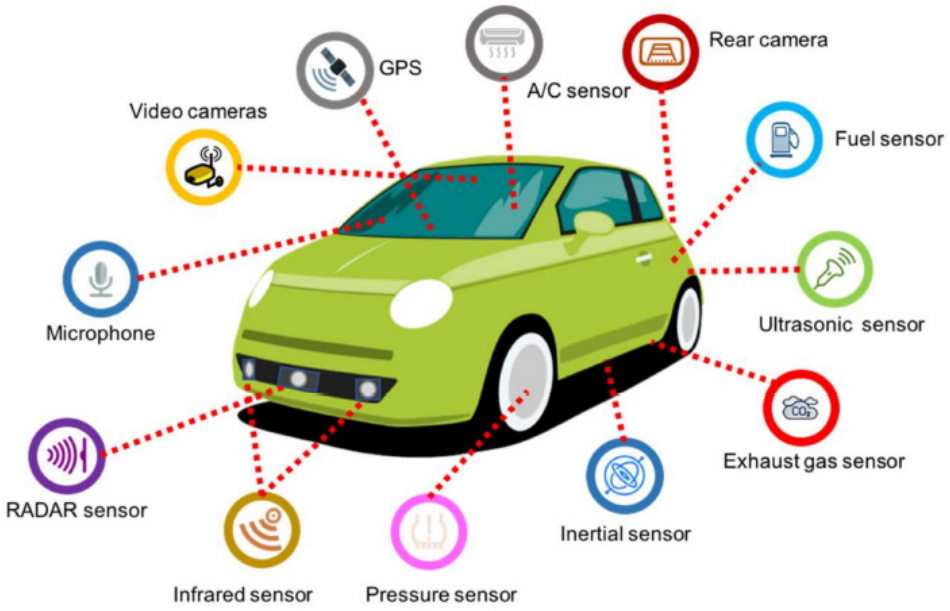


Figure 6. Different Type of on-vehicle sensors
(Guerrero-Ibáñez, Zeadally, & Contreras-Castillo, 2018)

Among the variety of in-vehicle sensors, gyroscopes and accelerometers are used to determine a vehicle's position, orientation, and velocity. The combination of gyroscopes and accelerometers with Global Positioning Systems (GPS) improves the detection of vehicle parameters. The utmost importance of these sensors is the improvement of road safety. Ultrasonics and electromagnetics are common safety sensors that detect objects at a short-range. They are useful for parking assistance or similar short-range detection. Moreover, these sensors are affected by weather condition, which reduces their accuracy (Ziraknejad, Lawrence, & Romilly, 2014).

RADAR and laser sensors are used for safety applications and were developed to scan long distances around a vehicle to detect objects (C.C. Wang, Thorpe, & Suppe, 2003). This collected data may warn the driver or activate emergency braking to avoid collisions. However, RADAR has a limited visual field and may not detect small road users if they are not in the center of a lane. Cameras are integrated with other sensors to detect perfect images around a vehicle. However, cameras are affected by adverse weather and lose their accuracy. LIDAR is a key component in AV evolution as they observe surrounding vehicles with high accuracy, detect object edges, and provide 360-degrees visibility in all weather conditions (Sun, Bebis, & Miller, 2006). However, their size, cost, integration with other components, higher aperture angle positions, inability

detecting colors and contrast, and unreliability under bad weather conditions (snow, fog, rain, and airborne dust particles) are huge drawbacks to their adoption.

Global Positioning Systems (GPS) are the main sensor for positioning an AVs as they provide the location of vehicles on a precise map of a roadway (Goel, Dedeoglu, Roumeliotis, & Sukhatme, 2000). This system may suffer from interfere and signal failures due to atmospheric disturbances in adverse weather (H. Lee & Mousa, 1996). Figure 7 shows an example of each sensor's responsibilities in terms of AV functions. GPS is a prerequisite sensor for AVs as it has perfect resolution under clear skies.

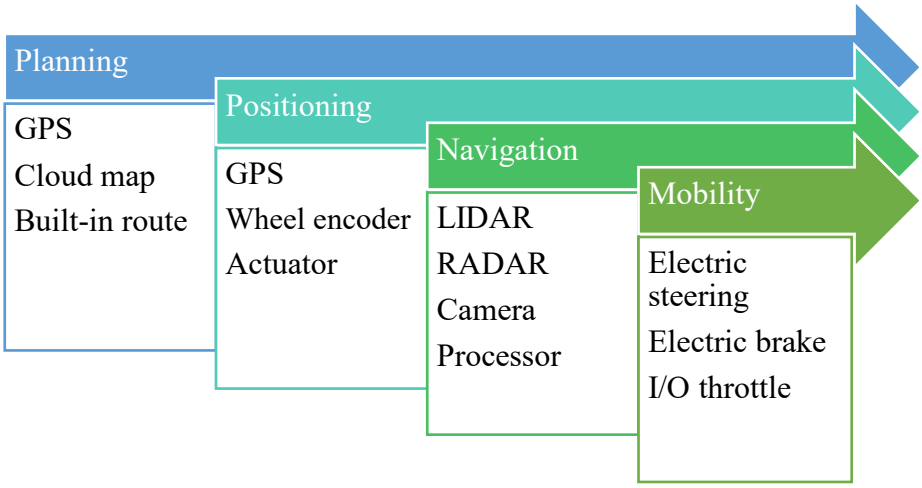


Figure 7. Functional responsibility of on-vehicle mobility sensors

Table 1 shows a summary of previous studies on the failure of in-vehicle sensors in safety applications. LIDAR had the highest probability of failure (10%), and GPS had the lowest probability of failure (approximately 1%).

Current AVs prefer to adjust speed and use reliable sensors, such as RADAR, instead of relying on a combination of expensive sensors, such as LIDAR and Cameras. For example, Volvo's safety kit reduces vehicle speeds to 50 km/h in urban areas when it detects ambiguous phenomenon in front of the vehicle, such as a crossing pedestrian or cyclist or adverse weather (Jakobsson, 2004). However, Tesla's team do not use LIDAR in their cars and rely only on cameras and RADAR sensors (Yan, Xu, & Liu, 2016)

Table 1. Summary of literature for the failure of safety sensors

Sensor	**Experiment**	**Probability of failure**	**Description**
LIDAR	Simulation	10%	Laser malfunction, optical receiver, short circuit, over voltage (Duran et al., 2013)
RADAR	Mathematical	2%	Detection due to noises (Swerling, 1997)
Camera	Simulation	4.95%	Foreign particles, shockwave, overvoltage, vibration (Duran et al., 2013)
GPS	Experiment	0.92%	Real-life tests performed in 4 different signal environment for more than 14 hours (Kuusniemi, Lachapelle, & Takala, 2004)

2.3.3 *Advanced Driver-Assistance Systems*

Advancements in information technology have opened many new potential applications in the transportation sector. ITS utilizes advanced information technology to assist drivers and enhance safe driving conditions for motorists. Driver assistance functions as a complementary support system designed to take over the lateral or longitudinal movement of a vehicle under certain conditions, such as parking assistance (Zhao, 2015). Integrated vehicle control systems will provide the highest level of safety by utilizing affordable in-vehicle sensors with different ranges and detection methods.

Advanced Driver-Assistance Systems (ADAS) were introduced as an advanced in-vehicle assistance system powered by an advanced communication system (Wu, & Boriboonsomsin, 2010). ADAS is specialized electronic equipment embedded on Autonomous Vehicles (AVs). ADAS deployment has gathered a nearly 21 billion US dollar turnover in 2015 and is expected to expand its market size to over 104 billion by 2022 (R. De Charette, 2012).

All detection and control procedures for ADAS driver-assistance systems work based on the quality of their installed sensors. ADAS uses various in-vehicle sensors (e.g., radar, ultrasonic, photonic mixer, camera, and night-vision devices) that improve vehicle control capabilities and enhance road-driving safety. On-vehicle sensors, such as ultrasonic, infrared, video cameras, and microwave radars, have the highest level of reliability; however, they have some inefficiencies in terms of sun exposure. Depending on the circumstances of the front vehicle, infrared cameras (for night vision) detect speed and approaching distance to announce warning messages to the driver. If it is impossible to prevent a collision, some inter-vehicle safety equipment is prepared and adjusted for better safety.

Ultrasonic sensors use energy waves to detect objects as well as weather and visibility conditions but must be mounted in a down-looking configuration, be perpendicular to an object, and are susceptible to high-wind speeds. Infrared works for different angles as well as day and night but can be sensitive to inclement weather conditions and ambient light. Video cameras show in-depth object detection with tracking possibilities but have poor performance under shadows, adverse weather, and intensive lights. Microwave radars operate by measuring the reflection of energy, but the reception of sidelobe radiation results in false detections. A new wideband spread-spectrum radar has recently been developed to remove the disadvantages of microwave radars as well as extend the detection range to 20 ft. However, the object detection capability of these sensors is limited due to them only detecting on a perpendicular sightline and they are hindered by roadside objects (e.g., parked vehicles).

Regarding the significant portion of human errors causing accidents (almost 90%), ADAS has enormous potential in avoiding traffic mishaps in the future (Ranney et al, 2001). Surveys (M. Zhao, 2015) showed the key role of ADAS in averting 40% of car crashes, based upon the type of ADAS installed, accident situation, and traffic environment. Alongside a rise in traffic safety and driving comfort, it is expected that ADAS will decrease traffic jams (Ranney et al., 2001) and reduce green-house gas emissions (M. Zhao, 2015).

2.3.4 *Performance Measures*

Prior studies on the development of ITS technologies have had common decision variables, including the number of VSLs deployed, the location of VSLs, and posted speed limits. Many studies were conducted to optimize the number and location of VSL under various road geometries and engineering judgments (Borrough, 1997; Habtemichael & de Picado Santos, 2013; Islam, Hadiuzzaman, Fang, Qiu, & El-Basyouny, 2013). For example, a study identified the optimal location of VSL signs when approaching a bottleneck on a highway (Xu Wang, Seraj, Bie, Qiu, & Niu, 2016). The authors evaluated ideal locations for VSL based on the acceptable average deceleration rate of drivers under free-flow speeds. Their study results showed the acceptable distance for a change in speed limit to be 700 m on upstream an incident point on a highway.

Additionally, a study evaluated the acceptable reduction of speed limits based on the standard deceleration rate suggested by the Transportation and Traffic Engineering Handbook (Homburger, Keefer, & Mcgrath, 1982), and realized that the effective speed reduction is at least 8 km/h. The

identification of proper locations for the installation of VSLs and acceptable speed changes by drivers provided the basis for the construction of a model layout and the evaluation of traffic operations for various speeds and traffic flow compositions.

OF all the various vehicle autonomy levels, some studies have focused on the simulation of platform and control strategies for traffic networks containing only fully connected vehicles (Kato et al., 2002; Rajamani et al., 2000). Several studies have considered AV interactions with conventional vehicles that result in a diminishing of traffic operation disturbances using autonomous vehicles (Gkikas, 2016; Merat & Lee, 2012; Saffarian, de Winter, & Happee, 2012; Walch et al., 2017). The mixed traffic of conventional and semiautomated vehicles was also partially studied to address traffic flow characteristics and environments (Bose & Ioannou, 2003). Presents for automated technologies have also caused difficulties in assessing transportation facility safety due to a lack of models that can predict crash potentials. Several studies have used traffic simulation analysis and interfaces to propose methods for improving traffic safety such as variable message signs and warning signals.

Several studies have investigated how AVs affect driving behavior at the micro-scale, such as acceleration/deceleration rate, reaction time, and capacity (Kato et al., 2002; Martin, Kalyani, & Stavanovic, 2003; Poczter & Jankovic, 2014; Rajamani et al., 2000; Ranney et al., 2001). Some studies have considered AV interaction with conventional vehicles that result in diminished traffic operation disturbances (Gkikas, 2016; Merat & Lee, 2012; Saffarian et al., 2012; Walch et al., 2017). The alteration of default driving model parameters in a microsimulation cannot imply full automation. Microsimulation safety evaluations may show inaccurate increases in simulated traffic conflicts due to smaller headway and safety distances for some sorts of AVs (Jeong, Oh, & Lee, 2017).

Conflict represents a situation where two vehicles may collide at different angles (Gettman & Head, 2003). The number of conflicts per hour can be used as a surrogate for the number conflicts per year (Kockelman et al., 2016). The Surrogate Safety Assessment Model (SSAM) is a common microsimulation interface that measures the number of conflicts and their severity based on trajectory files (Genders & Razavi, 2016a; Kockelman et al., 2016; Muley, Ghanim, & Kharbeche, 2018). An increase in congestion causes a higher number and severity of conflicts, which may increase the likelihood of crashes (Chen et al., 2018).

The SSAM model uses five parameters to estimate the severity of simulated conflicts: Time-To-Collision (TTC), Post Encroachment Time (PET), Deceleration Rate (DR), Maximum Speed (MaxS), and Speed Difference (DeltaS). Of these five parameters, TTC and PET have an upper limitation of 1.5 and 4, respectively, when considering conflicts with crash potential. It has been well recognized that a higher TTC value as well as lower speed and headway variance result in safer driving (Garber, Miller, Sun, & Yuan, 2006; Minderhoud & Bovy, 2001; Svensson & Hydén, 2006). Other parameters have been used to analysis conflict severity. The model distributes conflicts based on their approach angle (Figure 8) into three different types: rear-end, lane-changing, and crossing (Ghanim & Shaaban, 2019).

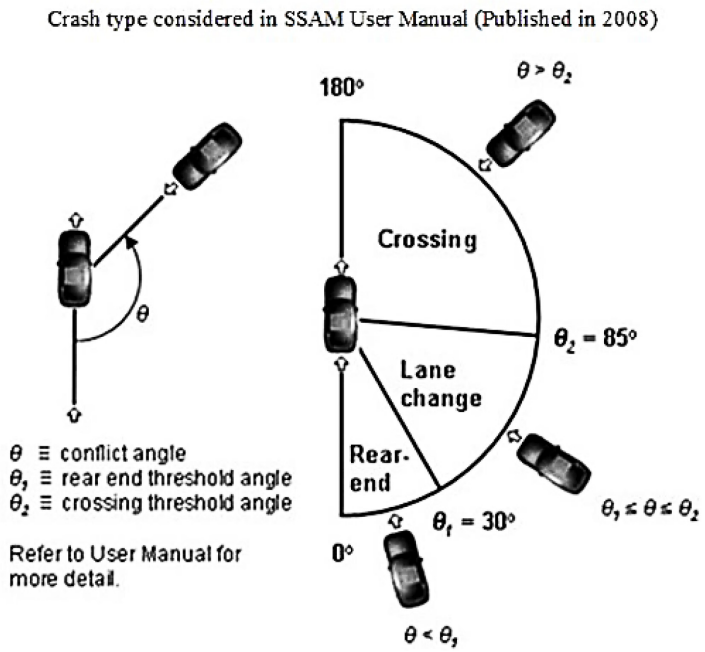


Figure 8. Conflict types in SSAM (Souleyrette & Hochstein, 2012)

SSAM provides acceptable results for different conflict types at intersections, but surrogate assessments of homogenous traffic flow have shown the misdetection of rear-end conflicts (Gettman & Head, 2003). Some studies have analyzed the risk level of conflicts based on posteriori descriptions of collected data to study the evolution of safety indicators related to traffic (Kuang, Qu, & Wang, 2015; Minderhoud & Bovy, 2001; J. Wang, Kong, Fu, & Stipancic, 2017). They used headway distribution, TTC, or the aggregated crash index to assess the severity of conflicts in a simulation model. The TTC indicator method (Minderhoud & Bovy, 2001) considers differences in the speed and acceleration of each pair of vehicles. It is the most realistic situation

for evaluating rear-end conflicts when traffic flow has a few changing lanes (e.g., homogenous traffic flow on highways or midblock in urban roadways).

Some studies examined potential AV applications to improve safety and mobility (Kurker et al., 2014; Olia, Abdelgawad, Abdulhai, & Razavi, 2016). The highest level of AVs has the ability to communicate with surrounding vehicles and infrastructures to provide reduce the likelihood of collisions (Genders & Razavi, 2016b). The authors also evaluated the limitations of vehicle TTC at different speeds and improved this limit by considering the relative speeds of vehicles. The results provided proof for reducing TTC thresholds for the use of connected vehicles.

Surrounding infrastructure and nonautonomous vehicles (i.e., human drivers) have a tremendous impact on AV performance due to AV low market penetration into general traffic. Recent studies by AV industry leaders have indicated that a majority of AV-involved crashes were due to interactions with conventional vehicles (Ahn et al., 2016; Delphi & Sepehri, 2016; Dixit, Chand, & Nair, 2016; Hörl, Ciari, & Axhausen, 2016; Takamatsu, 2016).

Other studies paid particular attention to understanding AV technological maturity and penetration rates through roadway capacity (Litman, 2017; Xiao et al., 2017). Talebpour & Mahmassani, (2016) simulated the impact of different autonomous vehicles' compositions on mixed traffic flows and found that AVs improved effectiveness. In another experimental simulation study, Shladover (2012) found increases in capacity were marginal up to a 50% AV penetration rate. However, model development has not been reliable due to the use of many car-following and lane-changing parameters. A sensitive AV parameter in car-following models is look-ahead distance, which is the detection range of utilized sensors.

As the benchmark for dynamic control technologies, the effect of autonomous vehicles on road safety and traffic operations has not yet been studied in their role as a countermeasure for visually impaired locations on roadways.

2.4 Advanced Collision Avoidance System

Vehicles equipped with ADAS receive warnings in critical situation. Also, automatic control allows the driver/vehicle to react to dangerous events in a faster and more effective way, such as automated braking or acceleration/deceleration systems. According to a report by Texas

Instruments and Bosch (Troppmann & Hoeger, 2005), there are two major categories of ADAS (Passive and Active). Passive includes all types of warnings (e.g., alert mechanisms or detection) whereas braking or collision avoidance are an active safety system.

Driver delays in recognizing dangerous situations causes at least 9% of all annual crashes in the US (NHTSA, 2008). With the rapid development of automated technologies, various ADAS applications have been designed to overcome human driver limitations and improve driver and passenger safety. Rear-end collisions account for almost 30% of all collisions in Canada and the US (Ranney et al., 2001; L. Wang, Abdel-Aty, & Lee, 2017). In order to reduce forward collisions, many studies have focused on the development of collision warning systems (C. Lee, Hellinga, & Saccomanno, 2003). Collision warning systems have been recognized as an intelligent system tailored to driver preferences (J. D. Lee, Hoffman, & Hayes, 2004).

2.4.1 *Advanced Collision Warning System*

The successful detection of hazards and the displaying of warnings occurs when a driver braking response exceeds a perceptual threshold. It depends on the kinematic information of the vehicles involved, driver reaction time, and the maximum deceleration of the subject vehicle used to trigger a warning. Many studies have investigated the success of perceptual based systems in TTC and the use of headways for safe and unsafe driving situations (Graham, 1999; Hogema & Janssen, 1996; Hurwitz & Wheatley, 2001; Nilsson, 1993; Van der Horst, 1991). They found that a TTC limit of 3 seconds reduced the number of false alarms, considering the division of headway space and a following vehicle velocity provides a perfect range for warning initiation. However, the perceptual system used some assumptions, such as restricted reaction time, zero deviation from relative velocities, and the same braking profile for all involved vehicles. These assumptions may limit the headway criterion from capturing all collisions because 75% of lead and following vehicles in rear-end collisions do not follow the same car-following state (Horowitz & Dingus, 1992).

Kinematic-based systems use better assumptions, including driver-reaction time and maximum deceleration rate to create active alerts in a potential collision situation. The initiation range is based on the required minimum stopping distance plus a two-meter safety margin as follows [12]:

$$R_w = \frac{V_f^2}{2a_f} + T_f V_f + 2 \qquad [12]$$

where,

R_w is the initial range of the warning [meter],

V_f is the velocity of the following vehicle [m/s],

a_f is deceleration rate of the following vehicle [m/s^2],

T_f is the reaction time of the following vehicle [seconds].

The first version of this algorithm was developed with some suggestions for maximum deceleration rate and driver reaction time under the same initial velocity to allow the following vehicle to miss the lead vehicle by 2 meters (Burgett, Carter, Miller, Najm, & Smith, 1998). Other researchers did not assume the same initial velocity for the involved vehicles and modeled the deceleration rate using the last minute of braking data (Brunson, Kyle, Phamdo, & Preziotti, 2002). Nevertheless, some studies developed a stationary lead vehicle system and suggested a 5.5 m/s^2 average deceleration rate for the following vehicle (Krishnan, Gibb, Steinfeld, & Shladover, 2001). However, driver comfort was not considered. Other studies considered a comfortable deceleration rate based on a vehicle's current speed and a 1.1s mean reaction time to prevent collisions (Olson & Sivak, 1986). The result of this development enhanced the detection of threatening scenarios.

Additionally, providing a cautionary warning to increase distance when facing common roadway incidents increases driver braking reaction time capability and provides a comfortable deceleration rate (Burgett et al., 1998). Uncertain lead vehicle behavior and changes in roadway coordination at incident points, such as the initial location of vision impairment, are obvious failure points that require a warning message that does not use front vehicle sensors (i.e., Camera, LIDAR, and RADAR).

Intelligent driver-assistance was developed using a GPS and vision-based system to provide real-time information to the driver (B. Zheng & Lachapelle, 2004). The system sends a passive alarm to the driver when a hazardous situation is detected downstream based on the vehicle's position, orientation, and velocity as well as road conditions and the surrounding environment (N.-N. Zheng et al., 2004). The focus of the study was on lane recognition and pavement condition. The authors used a high-precision offline GIS database to determine the roadmap base and used a real-time

GPS and sensory systems to warn the driver of downstream incidents based on their position on a digital map.

2.4.2 *Intelligent Braking Assistance*

Automated technology has been developed in AVs to overcome limitations in human driver tasks. In addition to the use of ACWS to warn drivers, some studies developed a control logic model called Intelligent Braking Assistance (IBA). The system works based on the functionality of the installed RADAR sensor in the front of the vehicle. Collecting robust data through this sensory system leads to the estimation of TTC for the use of a perceptual threshold system. When detecting a threatening situation, the control system takes control of brakes and steers away from a potential collision (Brunson et al., 2002).

The notable reduction in rear-end collisions due to collision avoidance systems has caused industries to develop their own safety models. Honda and Mazda consider driver preference and road environment in their safety systems (Krishnan et al., 2001; Seiler, Song, & Hedrick, 1998; Wilson, 2001). Overall, their safety systems used the kinematic-based model and were developed using sensory systems (Happian-Smith, 2001; Xuesong Wang, Chen, Zhu, & Tremont, 2016).

Another study developed basic collision avoidance system elements by conducting a series of studies on driver reactions to minimize the number of nuisance alerts and showed an increase in collision avoidance (R J Kiefer et al., 1999). The author extended their work to address timing requirements by using two last-second brakes to estimate potential braking severity (R J Kiefer et al., 2003). The authors calculated the warning range based on the prevailing driver reaction time of both vehicles (Raymond J Kiefer, LeBlanc, & Flannagan, 2005). Experimental results indicated that the developed model, under delayed timing conditions, was able to execute successful braking in over 85% of trials.

Braking assistance is predominantly a longitudinal control system that is rarely activated and acts as an additive support to a certain driver reaction. This control system is an important function of active safety systems. However, braking affects a vehicle's speed and has ab indirect effect on other safety systems (Summala, Nieminen, & Punto, 1996). Braking may result in deceleration (i.e., adjust headway with a leader vehicle) or a complete stop. Industries have developed automated braking as a fully automated headway control system that relies on stopping sight

distance and automation braking activation (Hoedemaeker & Brookhuis, 1998; Pauwelussen & Minderhoud, 2008; Rudin-Brown & Parker, 2004; Young & Stanton, 2007).

Road safety improvements through detection, warnings, driver perceptions, and braking requires that drivers have sufficient distance in a situation. This distance is called the Stopping Sight Distance (SSD) and is mathematically represented using key driver and vehicle characteristics. The key parameters of SSD are vehicle speed, perception-reaction time, braking distance, driver and object eye height, and pavement friction. Among these parameters, friction is uncertain due to its relationship with speed, tire condition, and pavement condition (Kouchaki, Roshani, Prozzi, Garcia, & Hernandez, 2018). In order to represent the driving environment, AASHTO (2001) developed a driver-performance model for stopping sight distance as follows [18].

$$SSD = 0.278Vt_r + \frac{V^2}{254(\frac{a}{g} \pm G)} \qquad [18]$$

where,

t_r is the perception-reaction time $[seconds]$,

V is the initial velocity of the vehicle in $[km/h]$,

a is deceleration rate in m/s^2, g is the gravitational acceleration, and

G is the percentage gradient of the road.

Vehicles equipped with IBA can calculate critical distance and provide quick stops under ordinary circumstances based on SSD and kinematic constraint criteria. Critical distance includes SSD and the vehicle final spacing (assumed as two meters). Additionally, the road gradient can be measured using GPS data to determine a vehicle's exact geographic position at each time step (Fraczek, 2003). Development of AWCS and IBA algorithms based on SSD avoids the difficult problem of estimating the deceleration rate of the lead vehicle and focuses on the qualities of the following vehicle.

Deceleration rate is recognized as the main kinematic constraint for avoiding a collision. A study reviewed previous developed emergency braking systems and proposed an interval between 0.4g to 0.85g for verifying a realistic deceleration (Goudie, Bowler, Brown, Heinrichs, & Siegmund, 2000). Another study used a driving simulator to determine the impact of early and late

deceleration rates on collisions (Hoffman, Lee, & Hayes, 2003). The results showed fewer collisions when receiving an earlier warning due to it providing a smaller criterion for deceleration rate (0.4g). However, earlier warnings resulted in more nuisance alerts. The NHTSA developed an algorithm with a fixed deceleration rate of 0.55g and a 1.5 second reaction time based on the Mont Carlo simulation (Najm, Stearns, Howarth, Koopmann, & Hitz, 2006). Comparing the false alarm and positive rates across various parameters showed that the fixed deceleration rate criterion could be adopted across all kinematic conditions.

2.5 Concluding Remarks

The occurrence of sun glare was investigated from different aspects of roadways. However, identifying the location of blindness under the sun glare and the transition between sun glare and contrasting shadows have not been considered yet. Besides, the simulation environment was not considered for emulating detection of uncertain incidents on roadways, such as sun glare occurrence.

The Wiedemann car following was recognized the best for modelling driving behaviour in vision impairment. The application of potential driver assistance and vehicle control systems in response to driver vision impairment has not been emulated in a microscopic simulator. Moreover, the SSAM showed an untrustful method for assessing conflicts in homogenous traffic flow. A new method of conflict analysis should be defined based on changes in the velocity and acceleration of the vehicles and sensitive parameters in the car-following model.

Chapter 3 EXISTING VISIBILITY AND CREATION OF DATABASE

This section presents a step-by-step method for systematically identifying vision impairment locations on roadways caused by sun glare under different built environment (Figure 9). The proposed methodology can be applied to any study area with suitable geographical information data. The development of a suitable glare-detection algorithm at a network level relies on a complex analytical procedure built on a geographical information system. The algorithm's input data consists of a complete 3-D road network geometric alignment, including elevations, as well as hillshade data and solar radiation information for the study area. Figure 10 represents the core of the proposed methodology in four stages. First, robust sun position algorithms are used on each analysis period for a given road segment. Second, geometry model analyzes the visibility reduction of drivers based on the visible sun trajectory and situation of the vehicle on the road segment. Third, the projection of buildings' elevation on the road segment identifies shadows as a different cause of vision impairment. Finally, the impact of cloud coverage considers providing the most accurate results for vision impairment analysis.

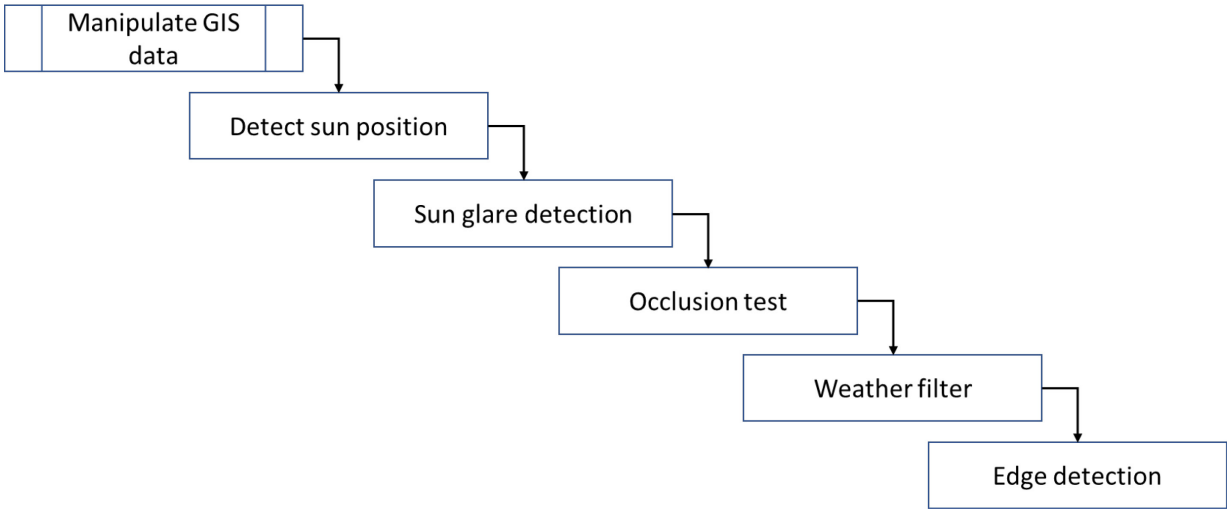


Figure 9. Components of visibility and sun glare detection algorithm

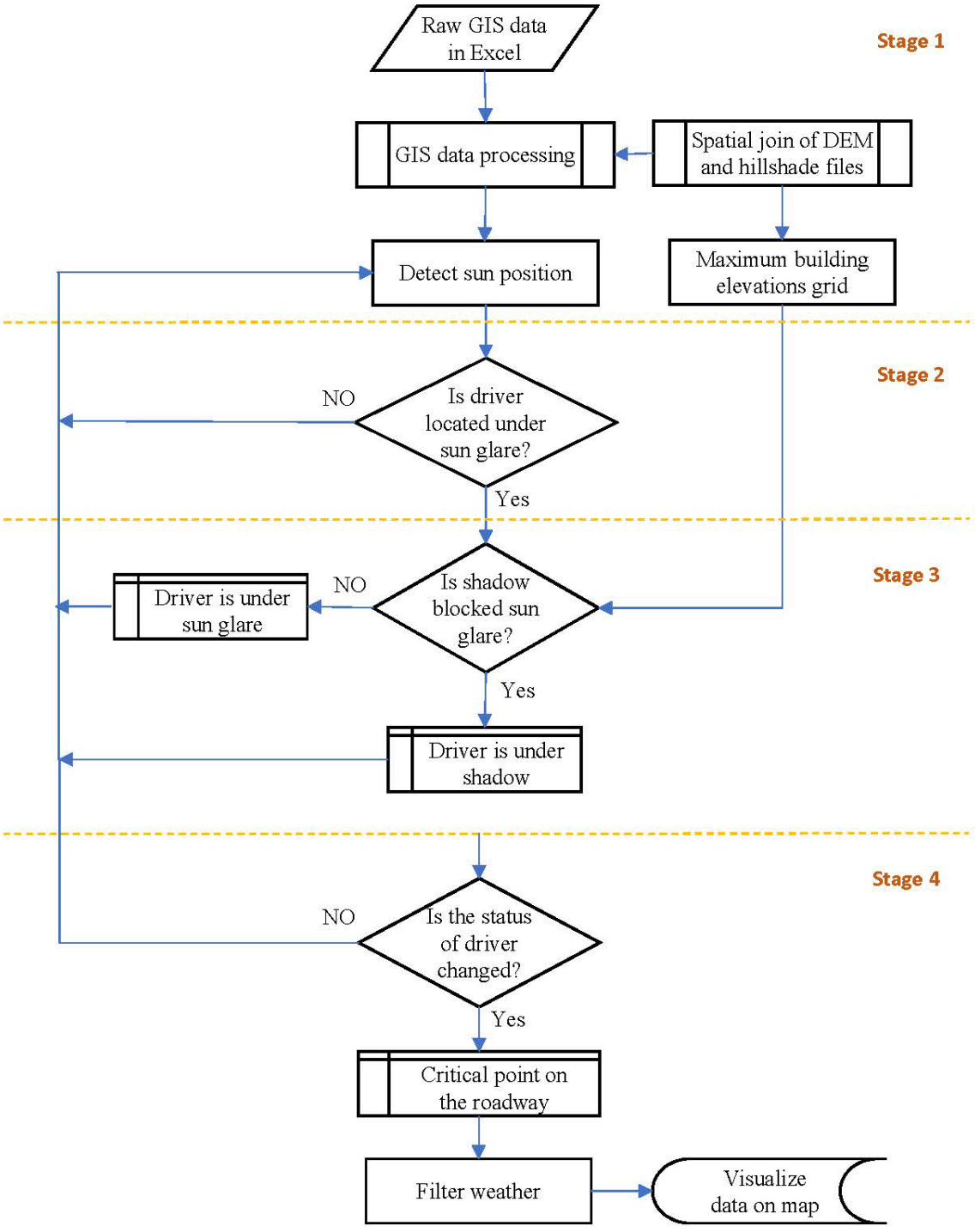


Figure 10. The methodology of vision impairment detection

3.1 Sun Positioning Algorithm

The occurrence of sun glare is directly related to the Sun's position concerning the observer's sightline. The apparent path of the Sun changes over the course of a year. Several astronomical algorithms have been developed to increase the efficiency of calculating the actual sun position (Michalsky, 1988; Spencer, 1971). They explore the temporal and spatial dimensions of the Sun at a specific point on the earth's surface. The algorithm used in this study to detect the Sun's position for a given location on the earth's surface has been developed by the National Oceanic and Atmospheric Administration (NOAA) Global Monitoring Division (Cornwall, C., Horiuchi, A., Lehman, n.d.). This algorithm requires the GIS coordinates and the timestamp of the location to be analyzed. The algorithm can process the coordinates input in the form compatible with the WGS84 geographic system (i.e., a coordinate pair of latitude - positive north and longitude - positive east). This solar calculator extrapolates the Gregorian calendar back through time to consider changes in the daylight system [19-25]. The modified NOAA algorithm uses the Universal Time (UTC) reference, and it works with a converted daylight system.

First, the fractional year γ, is calculated in radians to estimate the equivalent time, t_e, and declination angles, a_d.

$$\gamma = \frac{2\pi}{365}\left(day\ of\ year - 1 + \frac{hour - 12}{24}\right) \quad [19]$$

$$t_e = 229.18(0.000075 + 0.001868\ cos\,\gamma - 0.032077\ sin\,\gamma - 0.014615\ cos\,2\gamma - 0.040849\ sin\,2\gamma) \quad [20]$$

$$a_d = 0.006918 - 0.399912\ cos\,\gamma + 0.070257\ sin\,\gamma - 0.006758\ cos\,2\gamma + 0.000907\ sin\,2\gamma - 0.002697\ cos\,3\gamma + 0.00148\ sin\,3\gamma \quad [21]$$

Next, the algorithm can measure a true solar time, tst, in minutes,

$$tst = hour \times 60 + minute + {}^{seconds}/_{60} + t_e + 4 \times longitude - 60 \times timezone \quad [22]$$

Where, t_e is in minutes, $lat\ is\ the$ longitude in degrees (positive to the east of the prime Meridian), and $time\ zone$ is in hours from UTC.

The availability of true solar time provides corresponding solar hour angle, a_{sh}, based on each latitude, lat, and time to calculate the sun elevation θ_s and sun azimuth φ_s in degrees.

$$a_{sh} = (tst/4) - 180 \quad [23]$$

$$sin(\theta_s) = sin(lat) \times sin(a_d) + cos(lat) \times cos(a_d) \times cos(a_{sh}) \quad [24]$$

$$cos(\pi - \varphi_s) = -\frac{sin(lat) \times sin(\theta_s) - sin(a_d)}{cos(lat) \times cos(\theta_s)} \quad [25]$$

Additionally, the solar path is bent due to the refraction effect from the earth's atmosphere. The refraction effect displays unrealistic sun elevation when it is close to the Earth's horizontal surface (i.e., sunrise and sunset). The refraction analysis outlined in this study needs to be considered to identify the Sun's apparent position [26], particularly the sunrise's critical time when drivers might be affected by potentially devastating eye damage from the sunshine parallel to the sightline. Saemundsson proposed a refraction correction model when the actual angle elevation is known (Saemundsson, 1986). Besides, the change in pressure and temperature has been accounted for by Meeus (Meeus, 1988) with an algorithm that provided an updated formula in the NOAA algorithm [23].

$$\theta_{s,r} = \theta_s + (P/1010) \times [283/(273 + T)] \times \left[1.02/tan\left(\theta_s + (10.3/\theta_s + 5.11)\right)\right] \quad [26]$$

where,

θ_s is the actual elevation of the Sun,

$\theta_{s,r}$ is the corrected (i.e., apparent) elevation of the Sun,

P is the atmospheric pressure and

T is temperature.

In addition to analyzing the position of the Sun at a specific time, the location of the observer point is required to identify the effects of the position of the Sun. The proposed model assumes a homogeneous vertical/horizontal alignment of each road segment of a study area. Thus, the road segments are divided into the longitudinal five-meter long subsegments, the average length of typical passenger vehicles. The vehicles' movement on the road is similar to changing their positions between five-meter-long cells or subsegments. The driver position in a passenger car is assumed to be located nearly in the middle of each subsegment, regardless of the vehicle type. Also, AASHTO safety design guidelines assume an average car driver's eye equal to 1.08 m for passenger cars (T. AASHTO, 2011). The Sun's position is calculated through the modified algorithm of sun position for any points on the road within the analyzed period. The results of sun position are represented as azimuth and elevation angles (Figure 11) and are the preliminary input into the sun glare detection algorithm (Figure 9).

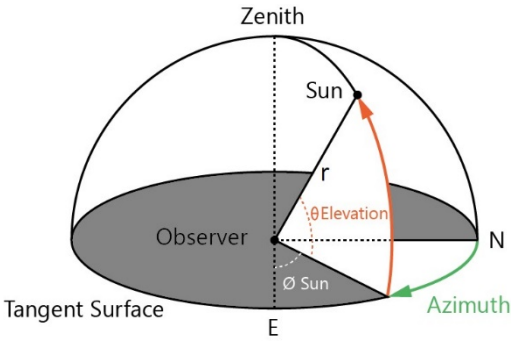


Figure 11. Sun position coordinate system

3.2 Sun Glare Detection Algorithm

In order to detect the sun glare phenomenon, it is vital to evaluate the alignment of driver sightline accurately. The sun glare occurrence is identified in the horizontal and vertical plane of the driver's vision towards the sun position. The glare angle is obtained from the angular distance function between the line of sight and the sun position. Several studies were conducted to quantify the angular distance at which glare causes distraction on vehicle control. One of these studies provided a review of the disability due to the sun-glare phenomenon (Vos, 2003). Another study focused on the lack of an appropriate angular displacement of signs in drivers 'vision. It concluded that the positioning of the Sun at the elevation of 20 degrees has the maximum aperture for visibility (Zwahlen, 1989).

Another study proposed a methodology to predict driver vision sensitivity under the sun glare (Jurado-Piña & Mayora, 2009). The author analyzed the visibility of different drivers' age based on the external and road surface illuminances and determined that the vertical angle of 25 degrees or less has the most vision impairment on the older driving population. According to another study, drivers' vision impairment is limited horizontally within the 30 degrees of drivers' vision apex, which are represented as acute vision angle (Gong, Walker, Hall, & Hurst, 1990). Regardless of the threshold values used, the assessment of the position of the sun relative to the observer's location on roadways requires an accurate position identification in a three-dimensional coordinate system.

The sun azimuth is an angular measurement in a spherical coordinate system (Figure 11). The most common conventional method of measuring azimuth for analyzing solar position is clockwise from due north, according to National Renewable Energy Laboratory (2013). The azimuth of sun position can be converted to a Cartesian system using **[27-29]**. First, the azimuth angle is converted into the polar coordinate system, where φ_s represents an angle from the east, and r is the distance

between observer point and sun, and to have a unit vector ($r = 1$), u represents the direction vector **[29]** in the cartesian system (i.e., x, y, z).

$$x = r * cos(\varphi_s) \quad [27]$$

$$y = r * sin(\varphi_s) \quad [28]$$

$$u = [cos(\varphi_s), sin(\varphi_s)] \quad [29]$$

The sun position and sun glare detection algorithms are coded using Python to eliminate the use of Geographical Information System (GIS) software and analyze algorithms' combination. The evaluation of sun glare occurrence needs to be considered for every subsegment of the road network. The compiled GIS data into the Python script is used to analyze the position of the Sun from any observer points. In our experiment, road segments are divided into 5-meter discrete subsegments (average vehicle length) and are examined based on corresponding road IDs. It is the most straightforward method by which the tangent vector can be estimated to calculate each subsegment slope as defined in **[30]**.

$$\delta = tan^{-1}\left(z_e - z_s / \sqrt{(x_e - x_s)^2 + (y_e - y_s)^2}\right) \quad [30]$$

where,

δ - represents the slope of a segment, it is calculated based on the tangential difference between x_s, y_s, z_s coordinates of start point of a segment and x_e, y_e, z_e at the end of segment.

Amount of $\varphi_{driving}$ has to be defined as a positive angle from the east on the horizon tangent surface; therefore, since $atan2$ in **[31]** returns values between $-\pi \; to \; \pi$, the angle is modified by adding 2π when negative values are returned.

$$\varphi_{driving} = atan2(y_s, y_e) \quad [31]$$

In this study, the range of drivers' sightline that might be affected by sun glare is represented by a cone with the apex on driver's eyes. In principle, the direction of driving is defined according to the direction of driver's vision (i.e., the windshield and front side windows). Consequently, the formulation in **[32]** represents a limitation in detecting the sightline under the vision impairment condition.

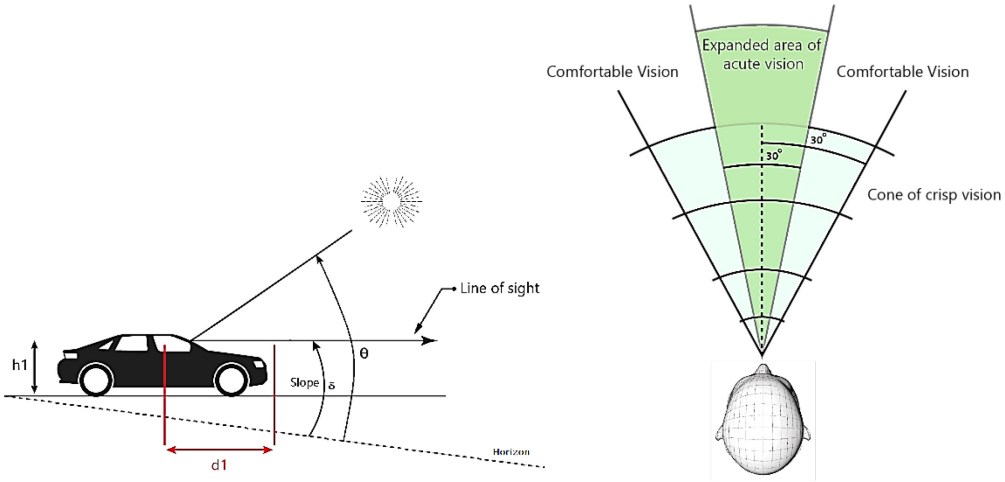
$$abs|\delta - \theta| < 25 \;, \quad AND \;, \quad abs\left|\varphi_{driving} - \varphi_{sun}\right| < 30 \quad [32]$$

where,

θ – represents the elevation of the Sun,

φ_{sun} – is the conversion of sun azimuth to measure from the east direction (Figure 11).

The slope of a road segment, δ, is identified from the elevation values at the ends of the road segment (Figure 12a), while the driving direction, $\varphi_{driving}$, is evaluated based on the x, y coordinates at the two ends of the road segment. The extended area of the acute vision is an acceptable 30 degrees limit (Figure 12b) between the horizontal alignment φ_{sun} of the sun and driving direction $\varphi_{driving}$.



a) Vertical angle from sightline *b) Horizontal angle from sightline*

Figure 12. Glare angular limits

3.3 Occlusion Test

Driving under the sun glare occurs on many road networks with a straight alignment. However, the existence of shadow on the roadway has not perception threat for drivers, transition between shadow and sun glare may be an obvious threat for driver's health and road user's safety. According to the output of sun glare and shadow detection algorithms, transition points can be categorized in three kind of vision impairment comprise: entrance to the sun glare from normal condition, entrance to the sun glare from shadow, and entrance to the shadow from sun glare (Figure 13). Finding the precise location of transition points assists the simulation model to imitate actual driving behavior on roadways.

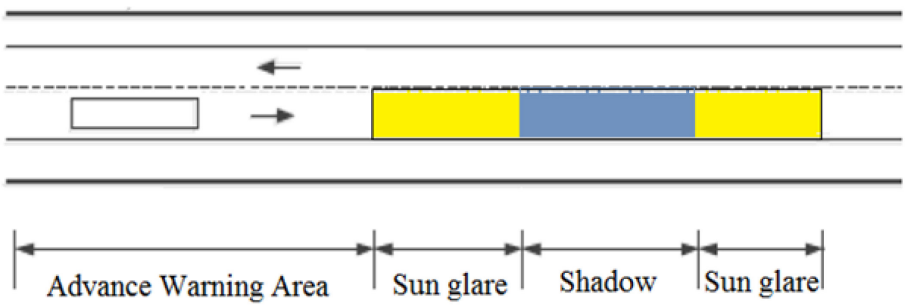


Figure 13. Sketch of a single lane contains sun glare and shadow
(yellow: the segment under sun glare, grey: the segment under contrasting shadow)

Various surface structures (e.g., buildings, overpasses, etc.) may temporarily block solar radiation and create contrasting shadows on specific roadway segments. The coverage of this type of shadows has a variety of length and shape on the road surface based on the building dimensions, road geometry, time of day, and the season. Therefore, a road user is exposed to sun glare at different locations along the road depending on the date and time of day, the coordinates of user's location, the azimuth and elevation angles of the Sun. The height of the shadow generating structure and terrain data analysis are required to consider impact of shadows on roadway. However, highly detailed resolution of this kind of analysis is a computationally intensive task. To demonstrate the feasibility of the proposed methodology with a computationally efficient approach, a fishnet of highest elevation points is generated with 5 by 5 meters 2D dimension to represent the maximum elevation of terrain and buildings roofs within each grid cell. Each cell is represented by its center point and corresponding coordinates of x, y, and the maximum elevation of cell. Figure 14a provides an example of how the defined grid can be utilized to evaluate the position of the observer with respect to the Sun.

Identifying a restricted influential area of grid cells provides an efficient algorithm for the occlusion test. Figure 14a illustrates that the rectangular investigation area (i.e., occlusion area) is extended from the location of drivers on the roadway, O, to the farthest length of locating maximum elevation point, E, to obstacle the sun ray. The coordinates of observer vector ($\overrightarrow{OE}$) are calculated as the center line of the rectangular area. Coordinates of occlusion area corners use to diagnose and recover the center of cells under the influence area by fitness functions **[33-36]**. The horizontal angle of the Sun, φ_{sun}, is modified to represents an angle from east-west axes.

$$x_i = x_{observer} \pm abs\left|5 * \sin \varphi_{sun-modified}\right| \qquad [33]$$

$$y_i = y_{observer} \pm abs\left|5 * cos\,\varphi_{sun-modified}\right| \quad [34]$$

$$x_i = x_{occluded} \pm abs\left|5 * sin\,\varphi_{sun-modified}\right| \quad [35]$$

$$y_i = y_{occluded} \pm abs\left|5 * cos\,\varphi_{sun-modified}\right| \quad [36]$$

For the purpose of the occlusion test, a ray detection algorithm figures out which center points inside the tangent rectangle locates inside the occlusion area by providing a direct link between each center point and the observer point to evaluate the number of intersections with sides of the inner rectangle. If a center point has no intersection with the sides of the rectangle, it will be considered as a point inside the occlusion area.

The center points identified inside the occlusion area are examined for the occlusion test (i.e., shaded cells in Figure 14a). The distance between the observer points and each center point is calculated to identify the corresponding sun elevation and investigate the situation of receiving sunray at this angle on the observer point (Figure 14b & 14c).

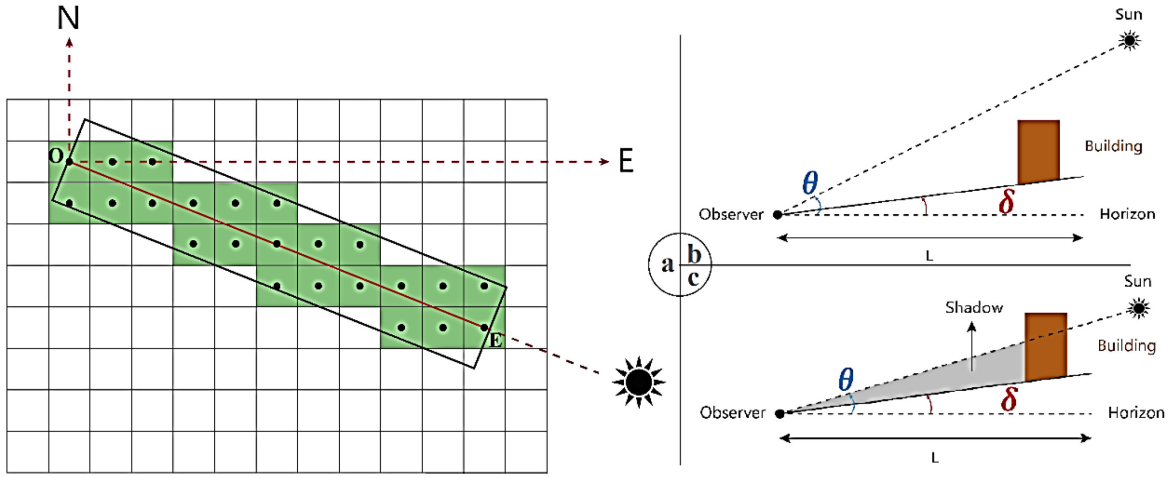


Figure 14. a) Plan view of the area for occlusion test; b) Glare condition; c) Occluded point

3.4 Weather Data Analysis

In addition to the occlusion of sunlight caused by buildings and terrain shape, clouds are another cause of the shadow. Cloud coverage has a bright shadow; thus, cloud coverage alone is not likely to create vision deficiency and it does not affect road safety. However, clouds may avoid the occurrence of sun glare within a specific time. The occurrence of sun glare is sensitive to atmospheric conditions and specific locations.

Cloud coverage is evaluated through the difference between the detector's received irradiance and the actual sun radiation. The Sun's radiation has a direct relation with the sun elevation. The atmosphere scatters and absorbs the radiation from the Sun. During sun glare, the Sun has lower elevations; hence, sunray passes the longer path through the atmosphere. Prior studies have investigated various mathematical models on the weather data to identify the clear sky and find a relation between collected solar radiation and sunlight intensity (Littlefair, 1985, 1988; T Muneer, 1995; Tariq Muneer, 2007; Perez, Ineichen, Seals, Michalsky, & Stewart, 1990; Ullah, 1996). Among different mathematical models, a study (Perez et al., 1990) is recognized as the most accurate and is selected to predict the hourly solar irradiance, S_i, values **[37]**.

$$S_i = \left[(I_{hd} + I_d)/I_{hd} + 1.041z^3\right] \Big/ \left[1 + 1.041z^3\right] \qquad [37]$$

where,

I_{hd} is the horizontal diffuse irradiance,

I_d is the normal direct irradiance,

z is the radians' solar zenith angle.

While I_{hd} and I_d are included in the collected weather data, z is the complementary angle of sun elevation and evaluated for any position of the Sun in the middle of each 5 minutes. On the other hand, clouds decrease the irradiance in lower sun elevations (i.e., after the sunrise or before the sunset), but this reduction has no fixed amount; instead, they reduce the sunlight by a certain percentage for that cloud type. The variation of incident irradiance value from **[37]** in various angles identifies the cloud coverage situation during the analysis period.

Historical atmospheric data from weather stations does not provide reliable or accurate cloud coverage information for specific locations on the roadway. Also, the heterogeneous displacement of clouds adds to the complexity of evaluating the percentage of glare occurring on partially cloudy days. Therefore, except the overcast weather (i.e., cloudy, snowy, foggy, rainy atmospheric conditions) all other conditions will be categorized as clear sky, and this is the data used for testing the sun glare detection algorithm.

3.5 Closing Remarks

In order to reveal critical points concerning sun glare occurrence on a road network, this section combined geographical road network data, an astronomical algorithm, specific trigonometric formulation, and data analysis in Python (9.1.1). The methodology has two steps, the sun glare detection algorithm and sun glare occurrence identification on the given network. The workflow of sun glare detection comprises of processing of data sources, the sun positioning algorithm (9.1.2) and identifying the critical angular location of the Sun for driving sightline (9.1.4). The methodology processes the GIS road data for a specific sub-segment of road and calculates the annual sun position by considering the refraction factor for the analysis period. The angular position of the Sun is evaluated within the critical sightline of the driver on the horizontal and vertical plane.

The algorithm developed in this section could be employed on many different procedures to reduce the sun glare effect on driving behaviors for existing or future roads. Driving in the current roadways, which are highly exposed to sun glare, needs additional traffic management facilities, such as variable message signs or providing a driving-assistance application to decrease the risk of vision deficiency. Therefore, in the next section, identifying the safe speed for a passage of roadway under sun glare occurrence is merely achievable.

Chapter 4 A MICROSIMULATION MODEL FOR REDUCED VISIBILITY TRAFFIC CONDITIONS

This chapter presents step by step the methodology used for analyzing drivers' speed behavior during sun exposure. A driver approaching a location that exhibits vision impairment conditions may inaccurately estimate the vehicle's proximity to surrounding objects, which may negatively impact traffic safety. AVs can be deployed as a potential countermeasure for these conditions. The visual impairment situation could comprise different combinations of AVs and conventional vehicles, and different safe-stop scenarios can occur. In this regard, a simulation model is generated for a base scenario and is developed to consider the impact of automated technologies on traffic operations and road safety.

The sun glare detection algorithm's development was the preliminary step to identify the duration of changes in drivers' visibility conditions. This section uses the developed dynamic integrated framework to detect sun glare occurrence on any roadways in the previous section. Supplementary information of cloud coverage is evaluated for a more accurate diagnosis of sun glare during the analysis period.

A calibration method for the development of car-following in the microsimulation environment is proposed to generate the base scenarios. Further development of scenarios needs to assess the model's sensitivity for changes in traffic flow parameters, vehicle compositions, location of the change in the speed limit, and penetration of AVs. The best fit of AVs' driving mode will be explained in detail, and a new method for the safety analysis of vehicle trajectories is presented.

4.1 Simulation Setup

Change in drivers' visibility affects the cognitive system and results in a psychophysical impact on driver's behavior. The emulation of driver behavior are the main concepts in car-following models. VISSIM microsimulation uses the Wiedemann psychophysical car-following model and simulate traffic maneuvers in the model to reflect the effect of visibility reduction on traffic operations.

The use of microsimulation to emulate traffic volume observed in the overcast sky (i.e., clear visibility) defines the base-case scenario or *Scenario 1*. Then, the vision-sensitive parameters are adjusted for driving under the sun glare through the impaired-vision base scenario or *Scenario 2*.

Scenario 1: Basic clear vision scenario - no sun glare exists (i.e., desired speed distribution under overcast sky)

Scenario 2: Basic vision impaired scenario - account for abrupt changes of traffic flow under the occurrence of sun glare

In the next step, the emulation of driving behavior in the car-following model is calibrated for both basic scenarios with congruent traffic measurements by the use of the best fit optimization method.

4.2 Calibration

Calibration is the third step of this study and a requirement for developing the simulation model to change the default parameter values in the model and minimize the error between the actual and simulated measures. For the calibration of the baseline scenarios, the traffic flow demand, the desired speed normal distributions, and the headway distribution are considered to take the longitudinal movement into account. The deviation of simulation outputs from observed data is through the objective function of the calibration process. Equation **[38]** represents the objective function of the longitudinal movement of vehicles in the network to reflect the real-world's driving behavior.

$$\boldsymbol{Obj} = Minimize\left\{\frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}}\frac{\mathbf{1}}{\boldsymbol{m}}\sum_{\boldsymbol{j=1}}^{\boldsymbol{m}}\frac{\left|\boldsymbol{h_{ij}^{obs}}-\boldsymbol{h_{ij}^{sim}}\right|}{\boldsymbol{h_{ij}^{obs}}}\right\} \qquad [38]$$

where,

i represents the different random seeds,

j is the different headway distribution histogram bins,

h_{ij}^{obs} and h_{ij}^{sim} identify the observed and simulated bins of random seed in i and j, respectively.

A sensitivity analysis is performed to test the possible combination of the model parameters on the certain boundaries. The Wiedemann car-following model in VISSIM needs to be adjusted for the perception-sensitive parameter: Average Standstill Distance ($CC0$), Gap time distribution ($CC1$), and additional desired safety distance ($CC2$). According to the Wiedemann car-following model, lower and upper bounds of each parameters are defined as $CC0_l = 1.22m, CC0_u = 1.67m, CC1_l = 0.7sec, CC1_u = 3sec, CC2_l = 2m, CC2_u = 7m,\ MTD_l = -3.6\ m/s^2$, and $MTD_u = -2.43m/s^2$.

The calibration time will be reduced dramatically by applying stochastic optimization algorithms. When longitudinal movement is the dominant movement in urban roadways, a single objective function can be considered. The PSO optimization method has been successfully used in prior studies to calibrate the parameters of the simulation model that are sensitive to longitudinal movements.

The PSO evaluates each set of parameters values to estimate the objective function value obtained from the simulation runs. The calibration algorithm runs VISSIM using different values of traffic behavior parameters and estimated the objective function. The optimization algorithm then evaluates the error of the objective function and proposes another set of parameters value closer to the global optimum to reduce the objective's value. In each iteration, the optimization algorithm tests the new set of parameters multiple times and chooses a new set for the next iteration. The final set of parameter values that yield the minimum of the objective function are used in the microscopic simulator to model the conventional vehicles.

4.3 Traffic Flow Composition

AVs' performance in a traffic stream is expected to have a different impact on traffic operations and safety to a certain degree. Prior studies on the development of ITS technologies for conventional vehicles evaluated the ideal location of 700 meters and effective speed reduction of at least 8 km/h for changing speed on the upstream of an incident point. However, the composition of AVs in traffic flow may change these limits due to using dynamic communication between AVs.

The most feasible mean to examine the potential impact of AVs on traffic flows is through simulation modeling. This study specified driving behaviour of automated vehicles by three safe

operating driving modes: normal, cautious, and all-knowing. Different driving modes of AVs are implemented in VISSIM via the Wiedemann 99 car-following with adjusted sensitive driving behaviors parameters.

A vehicle controlled by the normal AV driving mode behaves similar to a conventional vehicle that also has additional abilities (e.g., measuring distances, maintaining smaller headways, estimating speeds of the surrounding vehicles, etc.). On the other hand, a vehicle using the cautious AV driving mode follows the vehicles similar normal AVs but with larger headway for adapting a safe driving behavior. VISSIM support the activation of absolute braking system for cautious AV mode to maintain safer driving. Finally, the vehicle controlled through the all-knowing AV driving mode has enhanced perceptive and predictive capabilities and follows the leading vehicle with smaller headways than other AVs.

Table 2 shows the driving behavior parameters of AVs for the Wiedemann 99 car-following model. The average standstill ($CC0$) is the desired distance between the two vehicles; $CC1$ represents the headway time between two vehicles, and $CC2$ represents the safety distance to restrict distance variation between the approaching vehicle and the leader one. Increased acceleration is complementary parameter of driving behavior in VISSIM to set higher acceleration in the following process in terms of significant increase in the speed of leading vehicle. The default value of increased acceleration is 100% for human drivers and is increased based on the development in AV driving mode.

Table 2. Driving behavior of AVs logic in VISSIM for longitudinal movement

Driving Logic	**Interacting objects**	**CC0**	**CC1**	**CC2**	**Absolute braking**	**Maximum Deceleration Trail**	**Increasing Acceleration**
Cautious	2	1.5	1.5	0	Activated	-2.5	100%
Normal	2	1.5	0.9	0	NA	-3	105%
All-knowing	10	1	0.6	0	NA	-4	110%

The traffic models are applied to consistent scenarios concerning the transition of different AV types. Different scenarios are created to evaluate changes in speed limit for conventional and AVs to determine the safest driving metrics when drivers approach a vision impairment location. Therefore, conflict analysis of each scenarios leads to having the most reliable plan for the speed selection.

The pace of change in population and types of AVs are highly uncertain. Therefore, the combinations of penetration rates for different types of AVs lead to comparable traffic operations metrics. Contrary to the normal and all-knowing AVs, which use small gaps and trustworthiness of the sensory system, cautious AVs driving mode provides large gaps. The restriction of the gap and using an absolute braking system make an obstacle in front of cautious AVs exceed their automated technology. Thus, two types of AVs (i.e., normal and all-knowing) are investigated in this study, with a high potential of penetration into the mixed traffic. Table 3 shows the list of indicated variables accounts for scenario development of mixed traffic flow.

Table 3. Indicated variables in scenario development of mixed traffic

Vehicle types	AVs penetration [%]	Traffic flow per lane [veh/h/lane]	Desired speed [km/h]
Conventional (CV) Normal (AVN) All-knowing (AVA)	0 10 30 70 90	500 (Light) 1000 (Moderate) 1700 (Full)	100 90 80 70

4.4 Rear-End Crash Assessment

Rear-end crashes have been identified as a frequent type of traffic collision in road safety science. The investigation of the significant contributing factors to this type of crash validates the road safety development model's quality. It is essential to understand the effects of driving behavior, interactions of vehicles, and road characteristics on rear-end crashes. In this scheme, the potential area for a rear-end crash is estimated by developing of a technique based on stopping distance.

The figure shows the time analysis of a conflict in time step i and the prospected time until a collision occurs at $i + TTC$. The follower vehicle (green) provides a gap s_i from the rear bumper of the lead vehicle (yellow) at time step i. According to different vehicle interaction regimes, s_i may be reduced until time t when the follower keeps the minimum following distance (Δx_{safe}) to the lead vehicle. The car-following model reacts at this point to decelerate the follower and avoid any occurrence of collision in the network. Therefore, each pair of vehicles' trajectories should be analyzed based on the required time, TTC, or occurring a collision when vehicles maintain their instantaneous velocity and acceleration in the following time step.

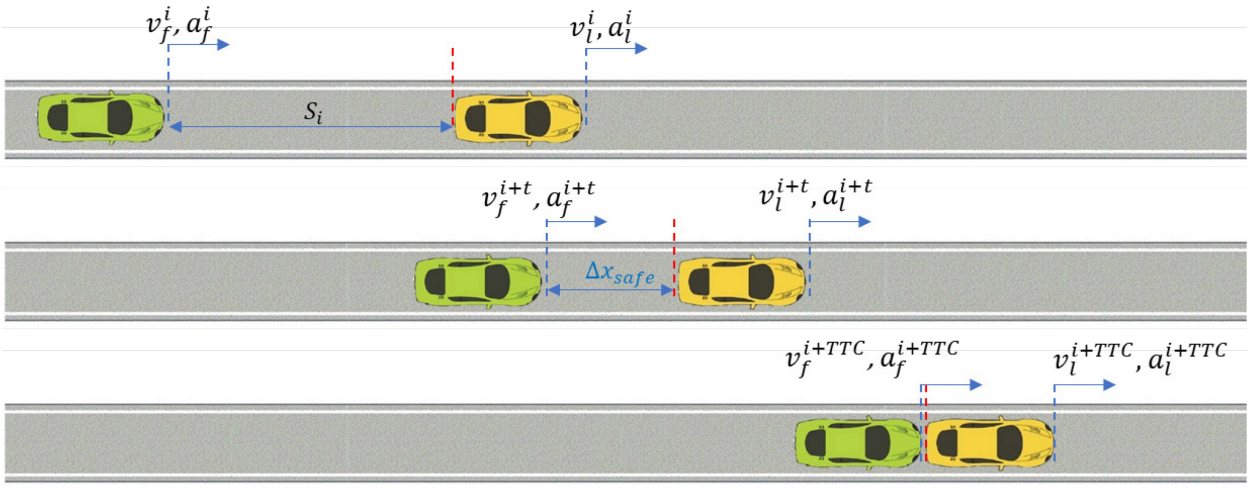


Figure 15. Time analysis of a conflict process

The microscopic simulator creates trajectory files by running the model containing information about each vehicle's movement in each time step. TTC is calculated based on the required time for occurring a collision when each pair of vehicles maintains its instantaneous velocity and acceleration in every time step.

Using the temporal and spatial information, the TTC indicator will be evaluated based on the interaction between vehicles includes positions, speeds, and accelerations for every time step, see equations **[39-41].**

$$TTC = \begin{cases} -\dfrac{\Delta x}{\Delta v}, & if\ \Delta v < 0, \Delta a = 0 \\ \dfrac{\Delta v}{\Delta a} - \dfrac{\sqrt{\Delta v^2 - 2\Delta x \Delta a}}{\Delta a} & if\ \Delta v < 0, \Delta a < 0 \\ -\dfrac{\Delta v}{\Delta a} - \dfrac{\sqrt{\Delta v^2 - 2\Delta x \Delta a}}{\Delta a} & if\ \Delta v \geq 0, \Delta a < 0 \\ +\infty & otherwise \end{cases} \quad [39]$$

$$\Delta v = v_l - v_f \quad [40]$$

$$\Delta a = a_l - a_f$$

$$\Delta x = \sqrt{\left(x_l - x_f\right)^2 + \left(y_l - y_f\right)^2} \quad [41]$$

where,

x_f, y_f, v_f, a_f are the kinematic components of vehicles including front coordinates, speed, and acceleration for both follower and leader. The components of followers represent with f and lead vehicle by l indices.

Equation **[41]** determines the net distance between the front bumper of the following vehicle and the rear bumper of the leading vehicle and Δv and Δa for the difference in speeds and accelerations of vehicles. In order to extract rear-end conflicts, the *TTC* threshold, TTC_t, is applied to get critical scenarios of interactions between vehicles. TTC_t in highways uses a larger choice of vehicles' interactions, includes the regular capacity headway, $\bar{h}$, and the calibrated headway parameter of the car-following model, $CC1$. The TTC_t is chosen to estimate conflicts in typical scenarios with different visibility as well as scenarios that simulate AVs penetrations **[42, 43]**.

$$\bar{h} = \frac{1}{q_{max}} \qquad [42]$$

$$TTC_t = \bar{h} + CC1 \qquad [43]$$

where, q_{max} is the maximum flow of the highway.

The total number of conflicts is not the only sensitive variable that results in a collision. The uncertainty in rear-end clearance needs a deeper analysis of *TTCs* based on the vehicle type to consider the severity of collisions with a new conflict measure. The change in the velocity of vehicles involved in a conflict demonstrates a suitable severity measure to evaluate the severity of conflicts. This study defines three-critical points to analyze *TTCs* by clustering an equal number of conflicts in three different indexes (i.e., severe, moderate, light). The severe index level includes a cluster of one-third of conflicts in the base model with the smallest TTC_t and represents the number of the most critical conflicts. The results of the safety analysis are used for controlling the operational speed when sun glare may occur by defining an advisory strategy.

4.5 Closing Remarks

This section employed the vision impairment location detection algorithm to analyze driving behavior in sun glare and determine the best solution to improve traffic operations and safety in mixed traffic flow. The workflow of vision impairment detection comprised of processing of

vehicles location, the sun positioning algorithm, and identifying the critical angular location of the Sun for driving sightline on the horizontal and vertical plane.

The VISSIM microsimulation and Wiedemann 99 car-following model were proposed and investigated for the parameters of driving behaviors on traffic operations and road safety. A TTC_t was introduced for identifying critical conflicts and synchronization with different driving modes of AVs. The conflict severity index was presented for clustering critical conflicts based on their severity.

Chapter 5 COLLISION AVOIDANCE SYSTEM UNDER SUN-GLARE

Prior research in ADAS (Advanced Driver-Assistance System) has led to various sensing, vehicle control, and communication technologies. The replacement of safety-related driver functions with technology makes a substantially safer system before. The safety application, including both drivers' warning system (passive) and active control system (proactive), proposes a dynamic integrated ADAS advisory. Passive mode generates user advisories or messages to prepare the driver for pre-emptive action. Proactive mode produces urgent warnings or actions to alert the driver about immediate action and avoid a crash. The integrated safety application provides a suitable mix of passive and proactive modes of ADAS under vision impairment.

Collision Avoidance System Under Sun-glare (CASUS) is an advisory application of ADAS to fill the gap of vision impairment and provide an effective activation of Advanced Collision Warning System (ACWS) and smooth braking by Intelligent Braking Assistance (IBA). CASUS uses the in-vehicle sensor-based system to detect long-range objects and GPS chips to support the subject vehicle's tracking. All the received data are analyzed in terms of vehicle speed, position, and applied brake. With the help of the processed data, potential collisions are predicted. The collision predictions are conveyed to the vehicle driver via a driver-vehicle interface mechanism. Figure 16 illustrates the components of CASUS comprises ACWS (i.e., driver vehicle interface) and IBA (i.e., safety application). Vision impairment locations provide a proactive plan of critical points to smooth deceleration in an urgent situation. CASUS behaves as a decisive application between using ACWS and IBA modes.

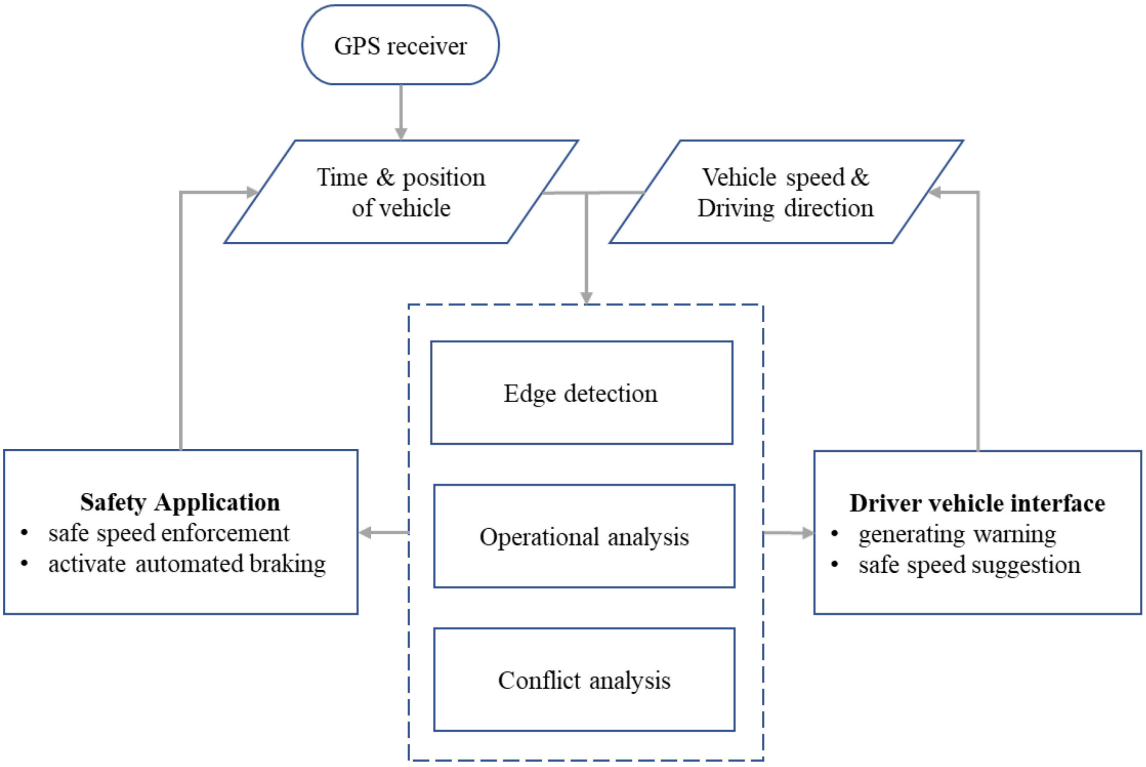


Figure 16. Component of Collision Avoidance System Under Sun-glare

The requirement of an advisory application to enhance road safety under sun exposure leads to the development of a four-level hierarchical framework. The higher level has access to the subject vehicle's information (time, location, speed, level of vehicle's automation) on the roadway using data provided by the global navigation system. Then the framework calculates the occurrence of vision impairment on the downstream road segment and provides this information to the driver. On the second level, the system will assist the driver in being aware of ahead visibility condition to avoid abrupt deceleration at the edge of vision impairment. The third level of the framework calculates the length of the sun glare occurrence to control the sensitivity of collision avoidance sensors and proposes a safe speed to the driver based on the expected change in visibility condition. The lower level of the framework, which contains the subsets of interacting AVs and the existence of downstream obstacles, controls automated braking system to provide a comfortable deceleration and avoid a collision.

5.1 Advanced Collision Warning System

The ACWS mode relies on the vision impairment detection algorithm described in Chapter 3 and the simulation model's outcome shown in Chapter 4 to alert about vision impairment and safe speed based on the location of the subject vehicle and the time of day. This mode provides a safety measure for further front distance in free-flow speed. The effective range for receiving a warning message is identified from the optimal-safe speed results of the simulation model's sensitivity analysis for various traffic flow conditions as shown in Chapter 4.

The preliminary stage of the model determines the occurrence of vision impairment on the following road for the length of 2 km ahead of the vehicle. Then, the passive mode of CASUS will be activated to calculate D, the real-time vehicle's distance to the beginning of sun glare, in the cartesian system through the equation **[44]**.

$$D = \sqrt{(Z_s - Z_v)^2 + (Y_s - Y_v)^2 + (X_s - X_v)^2} \quad [44]$$

where,

X_s, Y_s, Z_s are coordinates of the vision impairment location

X_v, Y_v, Z_v are coordinates of the vehicle location

By locating the vehicle within the premeasured range of receiving warning message R_w , the vehicle receives a warning including the distance to the vision impairment location and a proposed safe-driving speed is estimated. The warning will remain active until the vehicle reaches to the end of the sun glare segment and traveling of R_e. Figure 17 illustrates the location of the vehicle on the road approaching to the commencement of the vision impairment and distance for receiving a warning message. R_w is equal to decision sight distance **[45]** for avoidance maneuver class D or E for speed change in sub-urban and urban roads in AASHTO (A. AASHTO, 2001). Decision Sight Distance (DSD) provides an additional margin for error in the sight distance and afford sufficient length to have a safe drive. Table 4 represents some measured DSD for the urban and suburban road classes in different speeds. The value of DSD is substantially greater than stopping sight distance due to considering a much larger maneuver time for changing in speed.

$$R_w = DSD = 0.278\, V_v t \quad [45]$$

where,

V_v is the current speed of vehicle in every time step *[km/h]*

t is the total pre-maneuver and maneuver time suggested by AASHTO for different classes. t is defined 12 and 14 seconds for suburban and urban roadways, respectively.

Table 4. Decision Sight Distance for different design speed (AASHTO)

Design Speed [*km/h*]	**50**	**60**	**70**	**80**	**90**	**100**	**110**	**120**	**130**
Urban DSD [*meter*]	195	235	275	315	360	400	430	470	510
Suburban DSD [*meter*]	170	205	235	270	315	355	380	415	450

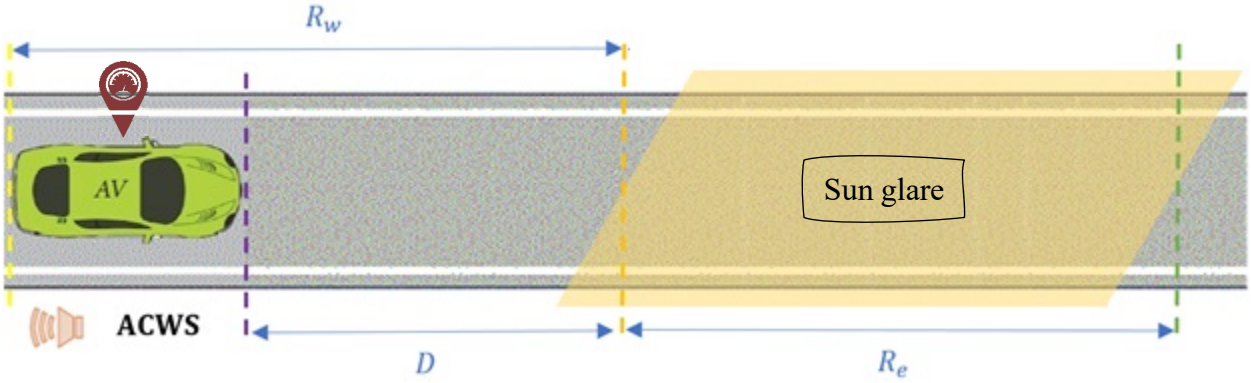


Figure 17. Activation Layout of Advanced Collision Warning System

5.2 Intelligent Braking Assistance

The noncompliance of drivers with the proposed safe speed by ACWS needs a complementary safety system to control the speed of vehicle with a gentle deceleration in a particular safe distance from the vision impairment location. CASUS observes the vehicle's speed and distance to the vision impairment location to active IBA as a proactive mode for the vehicle's safe passage through the vision impairment segment. IBA assists the driver in avoiding collision by providing hierarchical deceleration and emergency braking in facing a downstream obstacle.

CASUS supervise the vehicle in approaching to the vision impairment segment by adjusting the speed of AVs with a comfortable deceleration rate. The IBA will apply automatic braking from the minimum threshold of comfortable braking distance with a comfortable deceleration rate **[46]** in a hierarchical process (Figure 18). In order to consider the compliance of following vehicles for the speed reduction process, a hierarchical speed reduction of 10 km/h is applied between current and optimal-safe speed.

$$D_I = \frac{V_s^2 - V_v^2}{254\left(\frac{a}{9.81} \pm G\right)} \qquad [46]$$

where,

D_I is the activation distance of IBA on the upstream of the vision impairment location $[meter]$

V_s is the optimal-safe speed of driving under vision impairment $[km/h]$

V_v represents the current speed of the vehicle $[km/h]$

a is the comfortable deceleration rate, 2.5 $[m/s^2]$

G consider as the gradient of the road

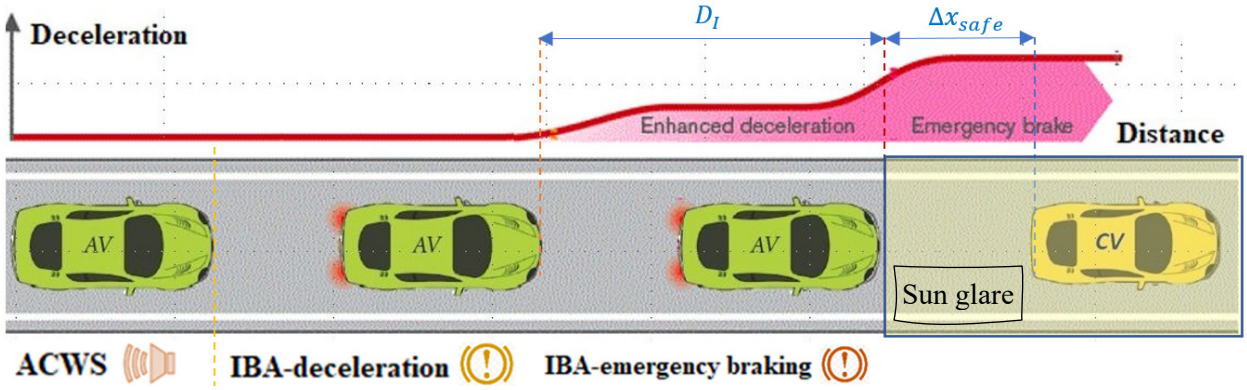


Figure 18. Layout of Intelligent Braking Assistance

5.3 Closing Remarks

This section proposed the CASUS algorithm by applying ACWS and IBA to provide a safe drive under vision impairment. The passive mode of CASUS, ACWS, used DSD criteria in AASHTO for applying the minimum distance for sending a warning to vehicles. The warning includes the information of impaired visibility situation in downstream and the optimal-safe speed for the vehicle's safe passage. In terms of noncompliance of drivers to the proposed speed adjustment, the proactive system, IBA, is activated to observe the vehicle's simultaneous speed and applying deceleration to keep the speed less than the proposed optimal-safe speed threshold.

The same layout of the model and car-following parameters have been used for validating the CASUS algorithm in microsimulation. The criteria of receiving warning messages at the premeasured distance from the location of vision impairment are implemented in VISSIM by running a script (see Appendix 9.2.1) to update the vehicle's status in the network at every time

step. Also, enabling the script for all vehicles in the network provided vehicles' controllability for applying the optimal-safe speed threshold. The developed model of CASUS in VISSIM provides a trajectory of vehicles to evaluate the impact of CASUS on road safety.

Chapter 6 NUMERICAL ANALYSIS AND SIMULATION

This chapter focusses on deploying the proposed methodology under prevailing traffic conditions. The preliminary step used the sun glare detection algorithm at a macroscopic scale to determine the hotspot locations for vision impairment due to sun glare on roadways. In the second step, traffic flow data under different visibility conditions were analyzed to identify a correlation between driving behavior and impaired visibility. Selection of an appropriate study area, data analysis, and impact of vision impairment should be made for the base model development. In the third step, the car-following model's driving behavior parameters in the microsimulation environment are calibrated for the base model. In the fourth step, the output of the simulated model was analyzed to determine the sensitivity of the model with respect to changes in traffic flow parameters, vehicle compositions, advisory speed limit placement, and penetration rate of AVs. Finally, the optimal solution for improving traffic safety and performance is to identify the best action in response to drivers' vision impairment.

6.1 Vision impairment detection

The risk of dazzling or blindness at critical locations negatively affects road safety of vulnerable road users, particularly on the most hazardous locations close to junctions and mid-blocks with pedestrian crossings. Conducting an empirical study examines the theory based on the real-world data and validates the modeling approach. Montreal is the second largest city in Canada and was used to evaluate drivers' vision impairment due to the variety of high-rise buildings in the downtown area (Figure 19).

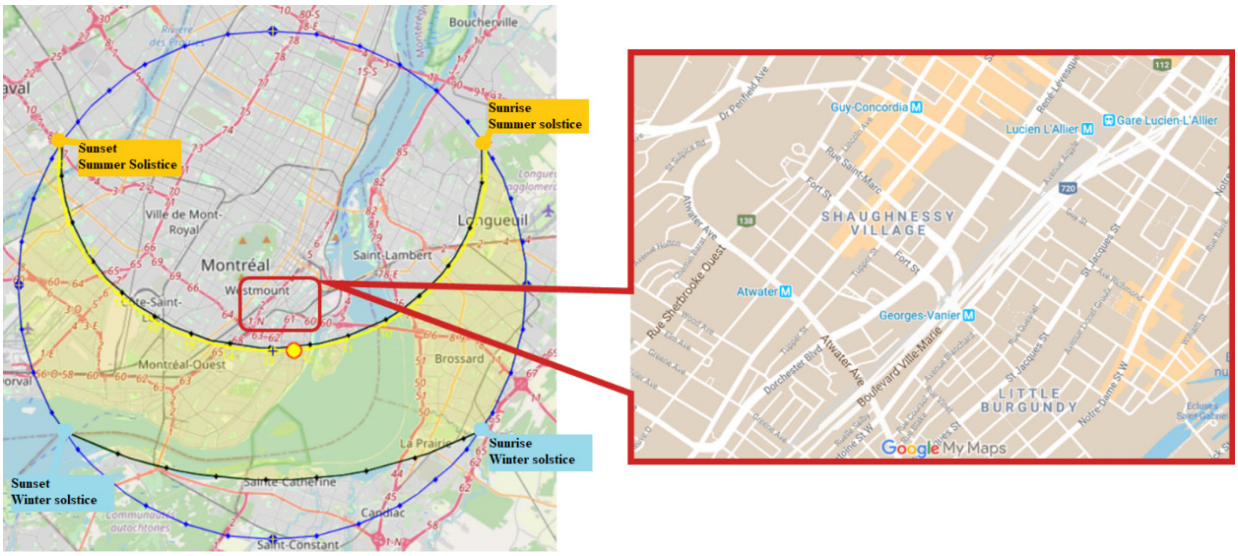


Figure 19. Study area: a) Center of Montreal include solar path; b) Section of Downtown

6.1.1 ***Data Collection***

The data needed for this research was grouped into GIS data and traffic flow data. GIS data analysis (Chapter 3) and traffic flow data (Chapter 4) was used to determine individuals' driving behaviors under the sun glare occurrence. Table 5 shows the list of collected data and corresponding descriptions and resources.

Table 5. Structure of collected data

Data	Type	Description	Source
Roadways	GIS	Double-side curbs of roadway for Montreal	Ville de Montreal
Digital Elevation Model	GIS	Raster Image of topography for Montreal	Canadian Open Government
Hillshade file	GIS	Raster Image by Air Lidar for Montreal	Concordia
Weather - hourly	Database	Hourly cloud coverage of weather at nearest airport	YUL airport
Traffic flow	Database	Volume, Speed, and Headway for the A20 highway by radar	MTQ
Vision Impairment	Video	Captured by installed camera in vehicle	Author
Vision Impairment	Photo	Captured by Fisheye lens	Author
Weather – in minutes	Database	Irradiance, pressure, humidity, temperature at radar station	La Fontaine Tunnel

6.1.2 *GIS Data Analysis*

The proposed methodology for sun glare detection requires the GIS of roadways in triple coordinates, grid cells of building elevation, historical weather data, and analysis period to reveal locations under vision impairments. The roadway GIS for downtown Montreal (Montreal, 2014) was discretized into smaller road segments (i.e., average 100 meters), but it missed the elevation information, and this was a primary parameter used in the methodology. To account for this limitation of the GIS data, the elevations from the digital elevation model (DEM) file were used in this case study (Montreal, 2015). The utilized DEM file was generated from an aerial image (Montreal, 2019a) and represents terrain elevation in the different light spectrum. In order to generate an elevation attribute into GIS roadway for Montreal the DEM file was converted into a contour layer, and then the elevation of the nearest contour line was interpolated on both ends of the road segments.

The dataset of the building's elevation was not available for Montreal and was generated from a combination of DEM and the hillshade files (Montreal, 2019b). The hillshade file is a shaded air lidar image of the surface terrain with the sun's relative position taken into account to enhance the appearance of surface objects. The resulting grid cells of the building's elevations was converted from the polyline layer of both DEM and hillshade images by joining the elevations' attribute from DEM to every segment of the building edges. The maximum elevation of the building's edges within each cell of 5 by 5 meters was clustered and represented through the center of each cell. This data processing accelerated the algorithm by homogenization of the building's elevations.

The prepared sources of data were utilized as inputs into the algorithm in the requested form. In the input stage, the developed Python programming scripts used tabular data format of the roadways network, the building’s elevations grid cells, the historical weather data, and integrated the analysis peak period to prepare them for the evaluation algorithms.

In the evaluation stage, the coordinates of the driver were considered at the center of each subsegment to evaluate the critical angular position of the Sun according to the NOAA algorithm (Laboratories & Laboratory, n.d.) and the corresponding refraction factor. Also, the occlusion role of the building’s elevation was evaluated through the grid cell data. The solar vector was constructed from each subsegment to the Sun position, every 5 minutes, for the entire analysis period, and the optimization method chose the most influential elevation cells along this vector. In

this regard, corners of the occlusion area were identified with a buffer of 5-meter from the solar vector, and the ray crossing method chose the best fit of cells within the occlusion area. The occlusion test evaluated the condition of influential building's elevation cells on driver's visibility by connecting the center of the cell and driver's location and comparing the elevation angle of the Sun with an elevation angle of each cell within the occlusion area.

6.1.3 *Vision Impairment*

The assessment of vision impairment results obtained for downtown Montreal illustrates the duration of sun glare and contrasting shadows in downtown Montreal for each month of the year, during the morning peak period (Figure 20). It can be seen that there is a relative mirroring of the January to May months versus the July to November ones, and this can be explained by fact that June and December are solstice months and the period of rotation of the Earth around the Sun. Moreover, the grid-like pattern of roadways in Montreal caused neither sun glare nor contrasting shadow within the analysis period between May and July. On the contrary, Montreal drivers experienced the highest amount of sun glare and shadows (i.e., 6000 minutes and 13500 minutes, respectively) during each February and October. Thus, the apparent position of Sun on these months caused more minutes of vision impairment and need particular safety countermeasures on the analyzed network.

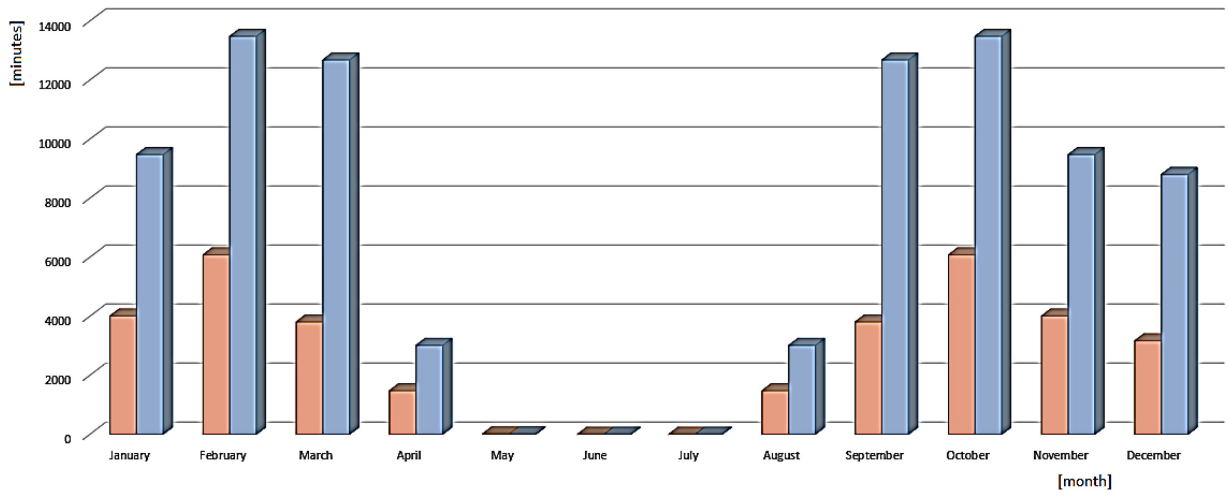
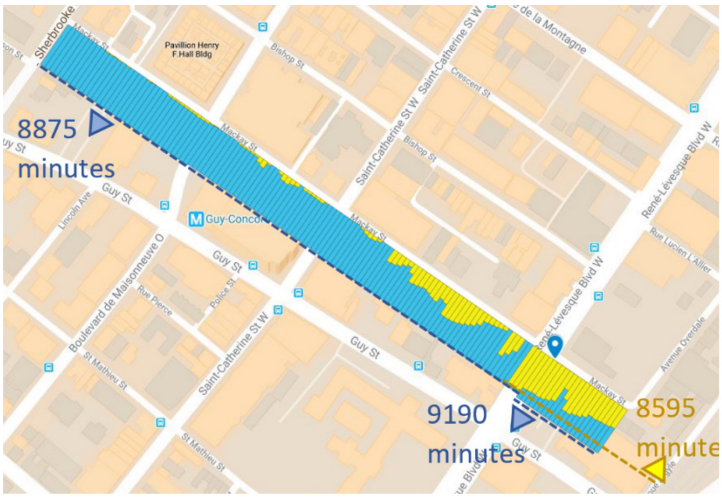


Figure 20. The expected period of vision impairment in minutes for the morning peak (6:30 to 8:30) in downtown Montreal for the entire year
Contrast shadows (blue bars), Sun glare (orange bars)

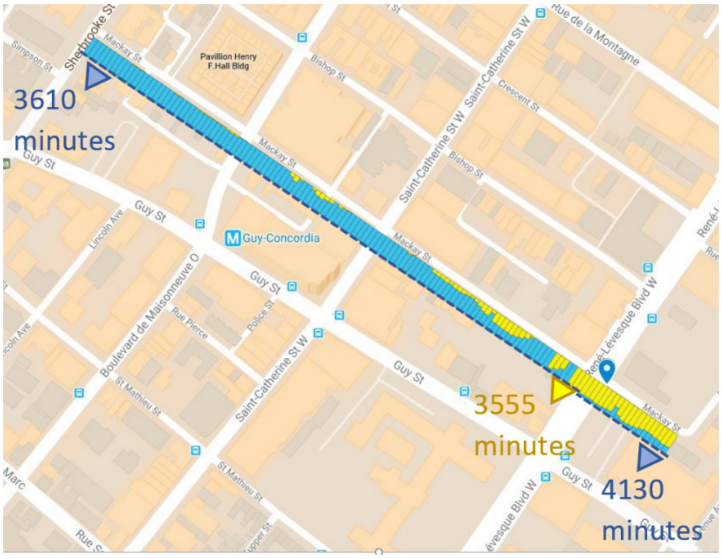
The best confidence level of detecting critical points was considered by eliminating the cloud coverage period from outcomes. The weather data including cloud coverage and precipitations collected from the nearest weather station at YUL airport for the period of 2005 to 2015 to determine the clear sky and overcast weather conditions. The impact of cloud coverage on annual glare occurrence was investigated on two arbitrarily selected sections of roads with same driving direction but with different gradient (i.e., gradient 4.5% - Simpson St. and level road approx. 1% - Mackay St, respectively). Figure 21 shows the accumulated time of sun glare occurrence in the division of occluded condition (blue bars) and the visible sun glare (yellow bars) for the selected roads. The event of visible sun glare was high in the approach of the vast junction with René Lévesque Blvd on the right bottom corner of the Mackey Street (Figure 21, b and d) while the most length of the street was under the event of invisible sun glare (i.e., shadow of high-rise building). On the other hand, drivers on the Simpson Street (Figure 21, a and c) might experience intermittent sun glare and contrasting shadow along the road. This is the main purpose of deterioration in road safety due to the blindness, as an impact of the transition from the visible sun glare to the contrasting shadow. Although the distinction of the clear sky based on the weather data reduced total minutes of vision impairment to half, the pattern of sun glare occurrence was consistent.

a) Annual glare conditions assuming 365 days/year of clear sky conditions (gradient road)



b) Annual glare conditions assuming 365 days/year of clear sky conditions (level road)

c) Annual glare conditions considering the clear sky days of the 2006-2016 period (gradient road)



d) Annual glare conditions considering the clear sky days of the 2006-2016 period (level road)

Figure 21. Total time of sun glare (yellow) and shadow (blue) occurrence for Simpson St (a) and (c) and Mackay St – (b) and (d).
(Reference: Map data ©2019 Google)

The algorithm was applied to wider area in downtown Montreal to reveal critical points of vision impairment in dazzling due to the sun glare or blindness due to the contrasting shadows. The algorithm evaluated the visibility situation of driver on morning peak hours (i.e., 6:30 am to 8:30 am) for the 20th of each month. Figure 22 illustrates a relative comparison of critical point occurrence

(i.e., sun glare to contrasting shadow and vice versa) in three seasonal day samples for the selected range in Downtown Montreal. The cloud coverage was evaluated for any identified results of critical points from the 10 years record of weather data and only critical points under the clear sky are shown on the Google map open-source data visualization tool and different months are shown in different colors (blue: February, orange: April, yellow: December). Although Figure 20 showed that drivers encountered comparatively less than a moderate amount of sun glare and shadow in April (i.e., approximately 1200 and 2800 minutes, respectively), the number of critical points in April was significantly higher in comparison to February and December.

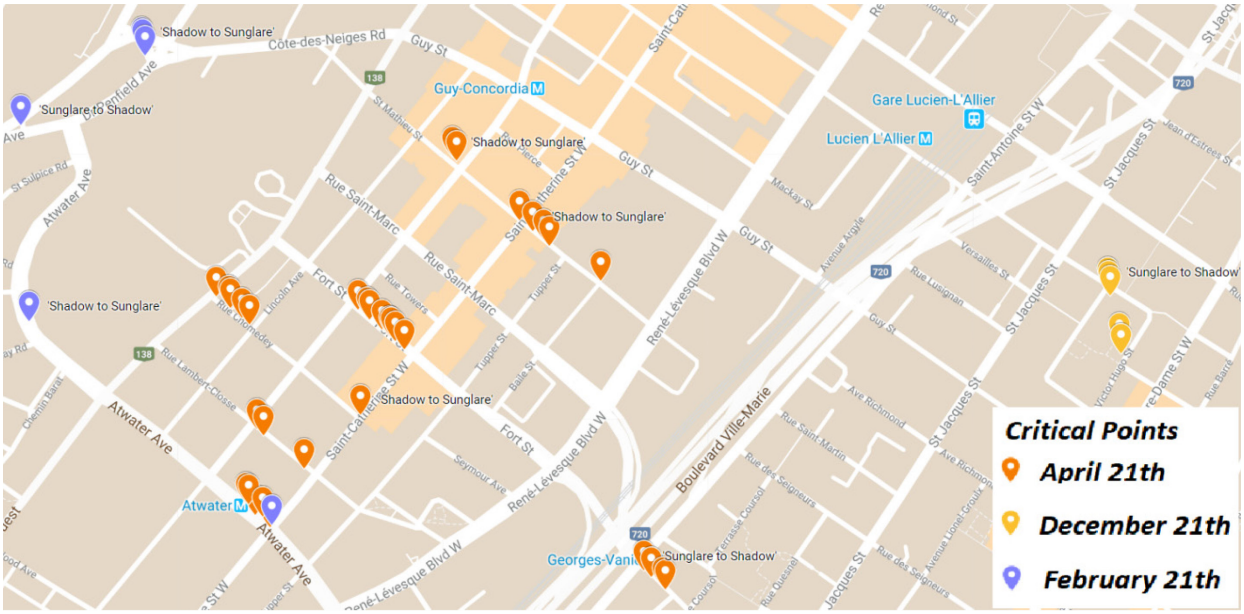


Figure 22. Annual critical points on road network of downtown Montreal (Reference: Map data ©2019 Google)

This study included an empirical verification of the application outcomes by studying glare-conditions along an arbitrarily selected arterial road of Montreal. This arterial in the proximity of downtown Montreal (i.e., street of Chemins de la Côte-Sainte-Catherine) was selected for an analysis of the sun exposure footprint. The occurrence of the sun glare and critical points for the morning periods during the peak hour was evaluated to identify the period and location that the Sun is considered to be at its worst angular position along the road. The application revealed the occurrence of sun glare between junctions with Mc Eachran Ave. and Boulevard Saint Joseph during 6:30 to 7:15 morning time interval on July 18, 2020. Additionally, another occurrence of critical points is near Ave. McCulloch and Ave. Laurier. Fig. 8a shows the outcome of the application on the Google map open-source data visualization tool.

The field investigation was carried out using a Sigma camera to verify the accuracy of the algorithm on detecting vision impairment locations. The camera was adjusted on the mode of taking the best photo shot in different exposure. Figure 23 shows the captured photos for each identified critical point including: 22b) driving under direct sun exposure, 22c & 22d) transition points between sun glare and contrasting shadow (blindness), and 22e) location of transition from shadow to sun glare (dazzling).

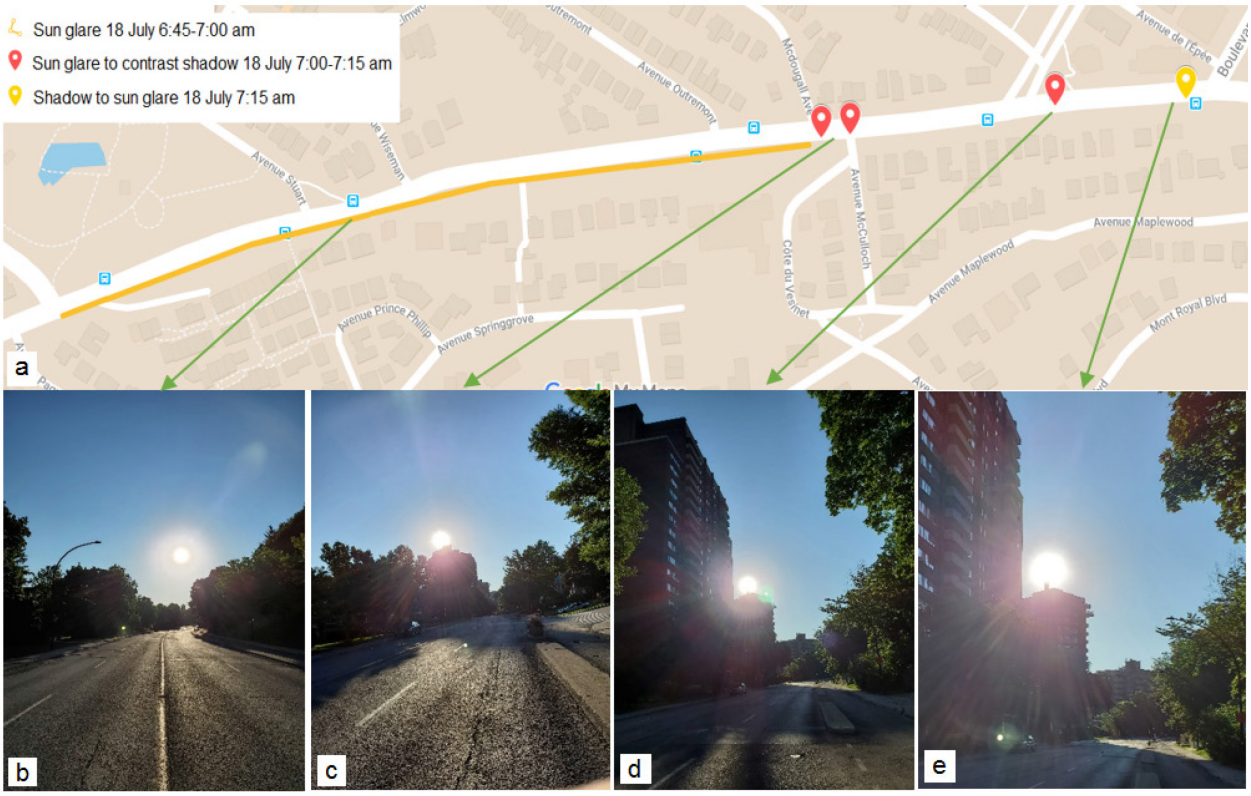


Figure 23. Sun exposure investigation on Chemins de la cote saint Catherine on 18th July (Reference: Map data ©2020 Google)

An accurate glare evaluation of photos in real scenes of contrasting shadows via luminance level requires the use of a wide view of fisheye lens (Inanici, 2006; Jacobs, 2007; Wienold & Christoffersen, 2006). In this study, a Sigma 4,5mm F2,8 EX DC HSM Circular Fisheye lens was used for the investigation of luminance of objects in sun exposure when driver approach to contrasting shadows. The luminance and visual comfort of photos have been analyzed by the Pintools online method. Identification of the blindness impact of sun glare needs to assess the exposure level (i.e., luminance) in the shadow and in the brightness area. A captured photo from a critical point (i.e., blindness situation) using the fisheye lens is illustrated in Fig. 9a, while the

corresponding minimum and maximum exposure of the photo are shown in Figure 24b & 23c, respectively. Both bounds of the exposure level revealed the high saturation of sunlight in photos and impairment of the vision in detecting the front of overtaking bus on the left side of the photo. The comparison of the clearance between the center of the vision and surrounding environment vividly proved that overexposure of the sunlight disables eyes on detecting objects under the contrasting shadow in front of the driver.

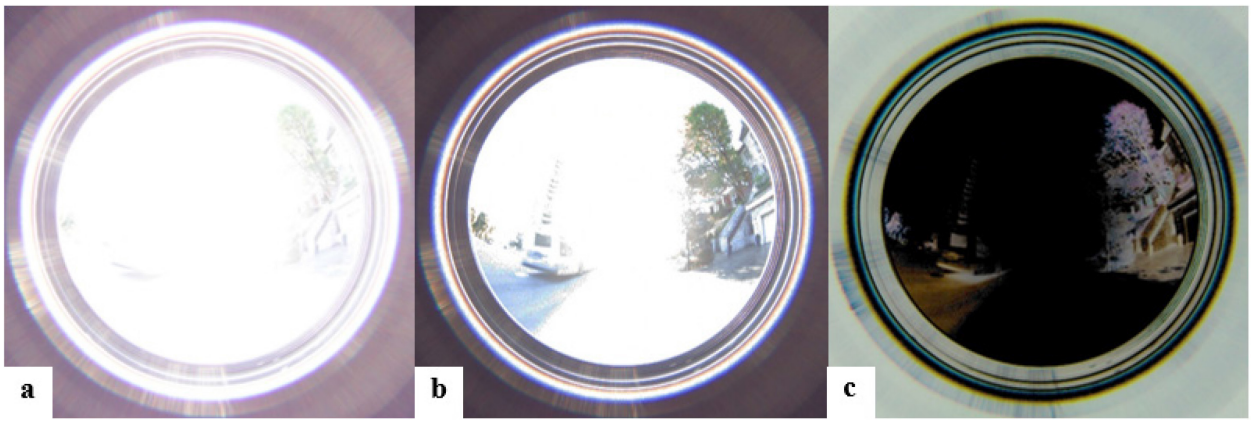


Figure 24. Captured sun exposure by fisheye lens

6.2 A Microsimulation Model for Reduced Visibility Traffic Conditions

The following section presents analysis of numerical results associated with the data from the selected radar detector, weather station, and GIS maps after presenting the methodologies. The case study is a 2.2-km long freeway segment along the TransCanada highway (A20) upstream of the interchange with Rte-132 West. Figure 25 shows the layout of the case study includes the data collection points for traffic flow and weather station. The geometry of the selected highway segment has zero slopes and an azimuth of 164 degrees. Drivers use a two-lane highway approaching the Island of Montreal.

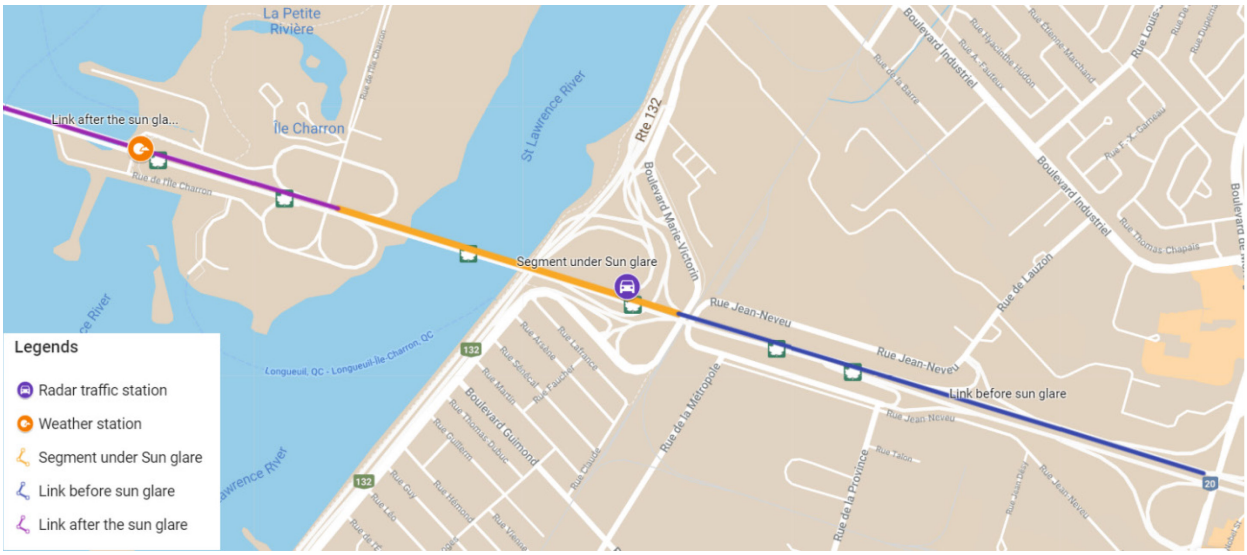


Figure 25. Location of study area – Highway A20 Northbound

6.2.1 ***Data analysis***

This study also tested the sun glare detection algorithm on highways by analysis of data sources, including historical weather data by the La Fontaine tunnel organization, prepared GIS maps by the Canadian Open Government, and traffic flow received by the Ministry of Transportation Quebec (MTQ). The methodology used the combination of all data within the scope of vision impairment to determine changes in driving behavior during the sun glare period in the clear sky condition.

The study's preliminary step investigated the occurrence of sun glare for drivers on the selected highway segment for the entire year. A clear sky cannot exclusively be identified as a perception dilemma for drivers; thus, the sun position and driving direction were investigated to reveal the occurrence of sun glare during each day. Figure 26 illustrates the trend of daily cumulative minutes of sun glare occurrence for the entire year along the study area. The result also shows the intensive situation of having sun glare, approximately 150 minutes per day, during spring and summer seasons. The occurrence of sun glare was also verified by field drive-through data collection from the front view's visibility situation. For traffic safety design, to capture near-worst case scenarios, AASHTO recommends that an average car driver’s eye is equal to 1.08m (T. AASHTO, 2011). However, for the purpose of this study and considering the typical passenger cars in Montreal a different value was selected. Thus, a camera was installed at 1.30m above the road surface (i.e., approximately near to the middle of the windshield) to validate the vision impairment conditions

at random locations as detected by the sun glare algorithm. Figure 27a represents a capture from the video recording of in-vehicle installed cameras on 8th of May, and Figure 27b the corresponding results of sun elevation from the sun glare detection algorithm. The result of the algorithm showed the low elevation of sun at 7:20 pm (i.e., approximately 7° from the horizon) which is visible in the captured photo.

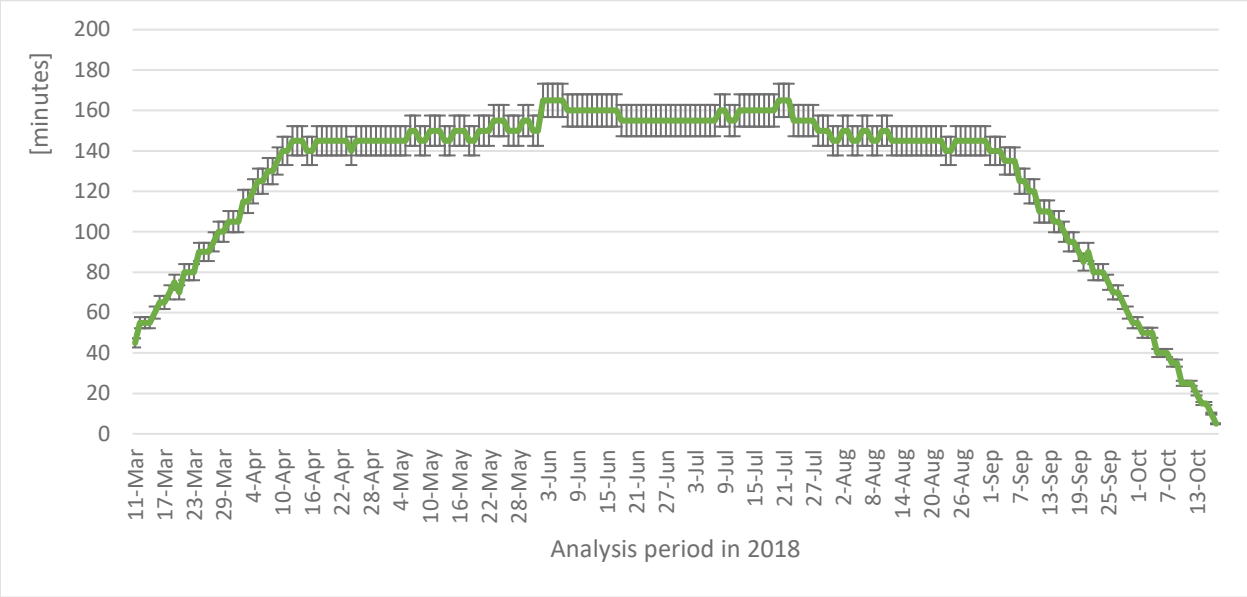


Figure 26. Daily cumulative occurrence of sun glare during the entire year (the Northbound of highway A20)

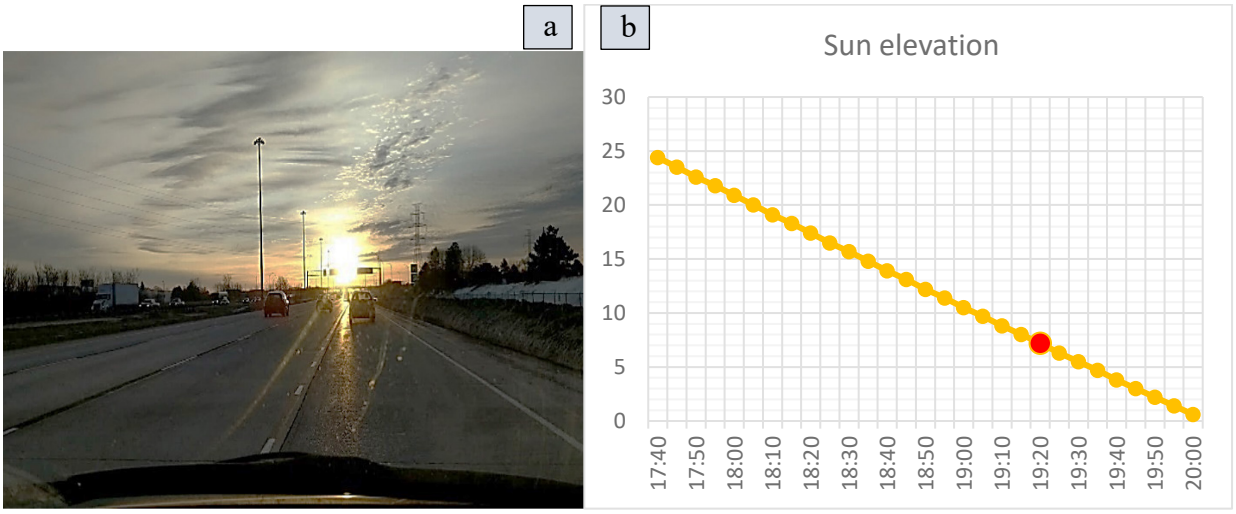


Figure 27. Validation of Sun glare detection algorithm on the highway A20 Highway A20 Northbound on 8th of May at 7:10 pm (sun elevation 9°, azimuth 285°) Elevations of the Sun from the sun glare detection algorithm for 8th of May

6.2.2 *Weather Conditions*

The impact of cloud coverage was assessed for 80 consecutive days, between 11th of March and 30th of May. Weekends and holidays were discarded to ensure consistency of the traffic flow patterns in the simulation analysis. Figure 28 shows the change of Sun irradiance S_i, as defined by equation **[37]**, for 15-minute analysis period, based on the weather data collected from the locally installed weather station. It can be seen that the irradiance and the elevation of the sun are highly correlated with the exception for the sun elevations smaller than 5 and larger than 10 degrees. The deviation of two lines at both ends represents the impact of atmosphere scatters on sun irradiance in a lower and higher sun elevation. For example, the impact of cloud coverage on sun irradiance is illustrated in Figure 29 where the incident irradiance was fluctuated between 10:00 to 14:00 due to the cloud coverage.

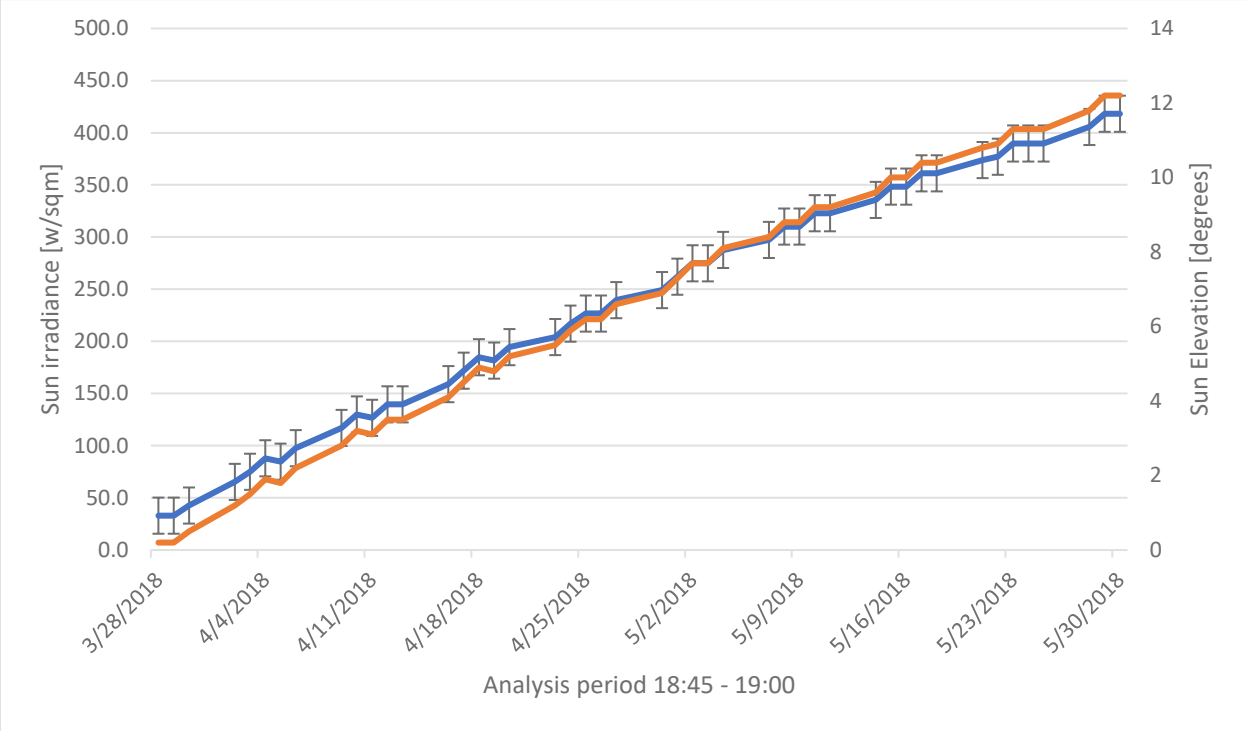


Figure 28. The atmosphere impact on variation of sun irradiance respect to the sun elevation for 15-minute interval of analysis period at the weather station (Blue: Sun irradiance, Orange: Sun elevation)

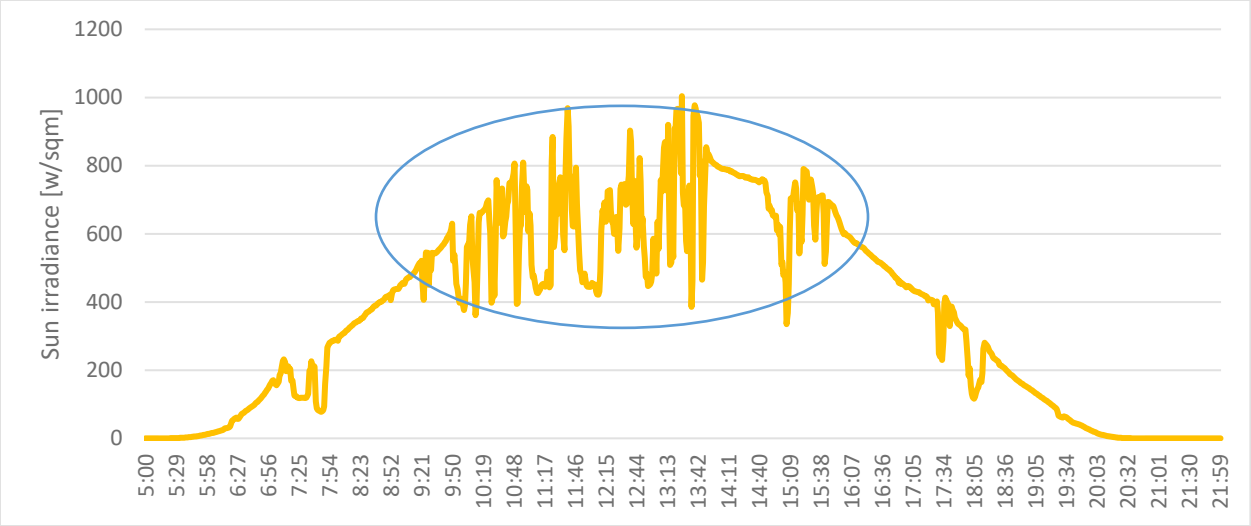


Figure 29. Impact of clouds on receiving sun irradiance on the detector at 17th May

The study's daily analysis period is limited to an hour in the proximity of the evening peak hour (i.e., 6:30 pm to 7:30 pm) to cover moderate traffic flow conditions on highway A20 approaching the Island of Montreal. These conditions are typically associated with relatively high speeds and moderate headways, leading to more critical traffic safety issues when critical conflicts occur. Figure 30 illustrates the interval of sun glare occurrence which occurred approximately between *6 pm* to *7 pm* in evening without considering the impact of weather conditions. Considering the weather impact on vision impairment led to disaggregate the clear sky from the cloudy condition during the daily analysis period. The historical weather data showed 16 days under the cloudy sky (i.e., the overcast sky without precipitation), 16 days under the actual sun glare, and 14 days unstable weather conditions (i.e., include precipitation or being partial cloudy). Table 6 demonstrates the disaggregated sky condition during the analysis period according to the analyzed irradiance at detectors.

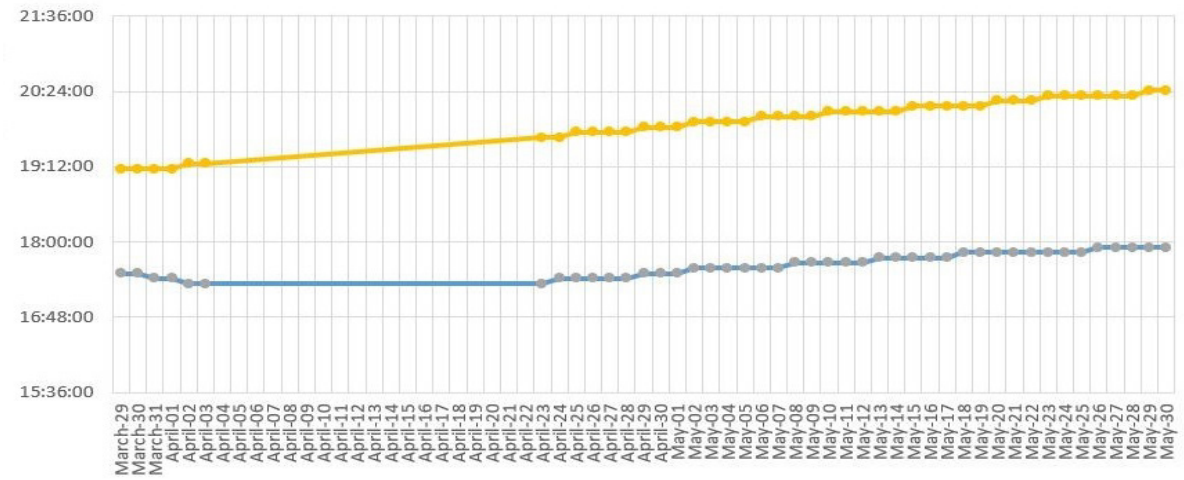


Figure 30. Prediction of sun glare detection algorithm for the occurrence of sun glare at the A20 North band in the evening (without considering the weather condition). Beginning of the sun glare (Blue), End of the sun glare (Yellow)

Table 6. Sky condition during analysis period

Date	Weather	Date	Weather	Date	Weather	Date	Weather	Date	Weather
28/03		10/04		24/04		07/05		18/05	
29/03		11/04		25/04		08/05		21/05	
30/03		12/04		26/04		09/05		22/05	
02/04		13/04		27/04		10/05		23/05	
03/04		16/04		30/04		11/05		24/05	
04/04		18/04		01/05		14/05		25/05	
05/04		19/04		02/05		15/05		28/05	
06/04		20/04		03/05		16/05		29/05	
09/04		23/04		04/05		17/05		30/05	

6.2.3 *Traffic Flow*

The potential impact of sun glare on driving behavior was analyzed using real-world traffic conditions. Traffic flow data was collected via a radar detector during the sun-glare analysis period. It includes the 15-minute average speed values of all vehicles passing the RADAR detector location (Figure 25). Vehicles speed records are aggregated within incremental speed intervals of 10 km/h.

The hypothesis test was obtained for the hypothesis's plausibility between two sample speed data, under the overcast sky and under the sun glare (Table 7). The null hypothesis was stated that the mean speed of each 15 minutes under sun glare is equal to the mean speed of the corresponding time interval on cloudy days. The alternative hypothesis was defined that vehicles' mean speed within each 15-minute interval under the sun glare is higher than the corresponding mean speed in overcast weather. The volume of vehicles had been normalized for each 15-minute interval to have a homological sample size, and the test was evaluated in the analysis period. The rejection of Levene's test indicated that 15-minute vehicle speeds had no equal variance and t-test should rely on the second row of each result. A significant 2-tailed of less than 0.05 in Table 7 was indicated as the rejection of the null hypothesis and passing of alternative hypothesis.

Table 7. Results of hypothesis test on the impact of sun glare for the average speed of vehicles

		Levene's Test		t-test for Equality of Means				
Analysis Period		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference
18:00 18:15	Equal variances assumed	3.675	0.065	-2.3	30	0.044	-4.687	2.038
	Equal variances not assumed			-2.3	18.6	0.049	-4.687	2.038
18:15 18:30	Equal variances assumed	3.52	0.062	-2.1	30	0.028	-4	1.904
	Equal variances not assumed			-2.1	18.3	0.033	-4	1.904
18:30 18:45	Equal variances assumed	3.833	0.069	-2.423	30	0.022	-6.12	2.526
	Equal variances not assumed			-2.423	18.9	0.026	-6.12	2.526
18:45 19:00	Equal variances assumed	4.38	0.083	-2.28	30	0.03	-6.317	2.771
	Equal variances not assumed			-2.28	19.2	0.034	-6.317	2.771

Headway and aggregated speed normalization indicated that the average speed in sun glare event under moderate traffic flow condition was at least 5 km/h higher than in the non-glare event with the same traffic flow (Figure 31). Additionally, the average and variability of 85th percentile headways were lower under the sun glare than the normal visibility condition (i.e., overcast sky). The observed difference in headways and speeds could be explained by the fact that deteriorated visibility conditions reduce the acuity of drivers' perception, and consequently, they tend to drive aggressively by increasing their speed to reduce the duration of vision impairment exposure as much as possible.

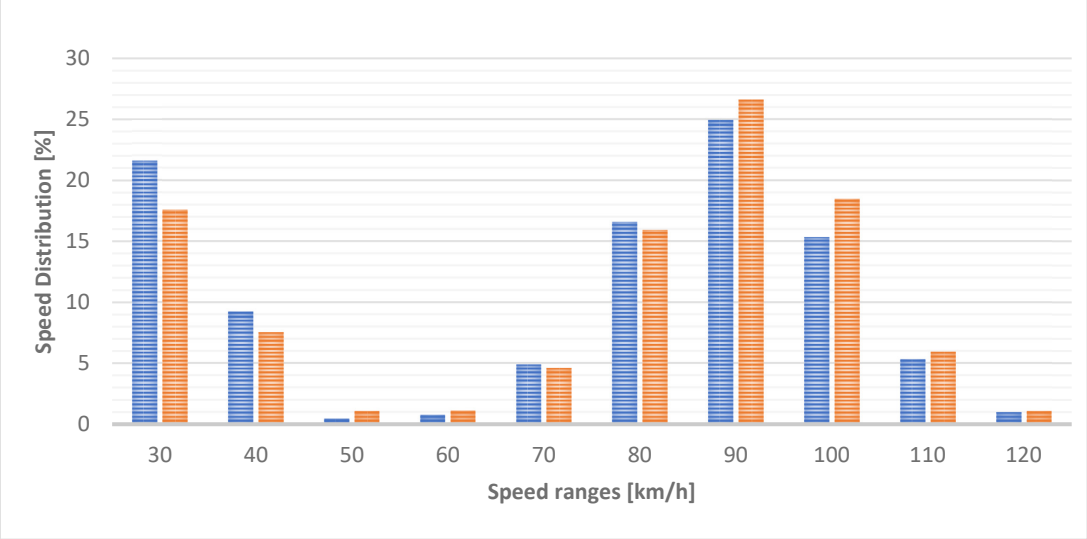


Figure 31. Changes in desired speeds due to sun glare (orange) and overcast sky (blue)

6.2.4 ***Calibration***

All sensitive parameters of driving behavior in the Wiedemann 99 car-following model were calibrated using Particle Swarm Optimization (PSO) algorithm. The calibration method used a combination of PSO algorithm and VISSIM COM scripting to minimize the objective function in 100 iterations, 1100 simulation runs, and validated in 15 different random seeds. The objective was adjusted based on the deviation of headway in simulation and observed data. In each iteration, the algorithm ran the simulation model multiple times and chose a new set of values for the parameters in the next iteration. The simulation runs continued until achieving the objective function closer to the real-world value. Table 8 illustrates the calibrated values of Wiedemann 99 car-following parameters in VISSIM for conventional drivers in two different visibility conditions. While the driving model followed the same values for the average standstill distance ($CC0$) and

additional safety distance ($CC2$), headway distribution parameter ($CC1$) fell from 1.3 seconds in overcast weather to 1.1 seconds during sun glare occurrence. Also, driving under the sun glare applied smaller Maximum Trailing Deceleration (MTD) value due to the aggressive driving behavior.

Table 8. VISSIM optimum parameters by Particle Swarm Optimization (PSO)

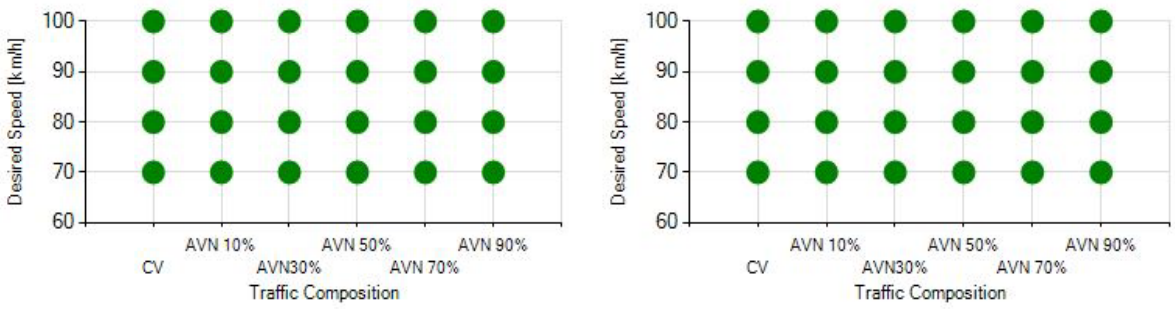
	CC0 [meter]	*CC1 [second]*	*CC2 [meter]*	*MTD [m/s^2]*
Overcast sky	1.52	1.3	4.7	-2.71
Sun glare	1.49	1.1	3.7	-2.88

The VISSIM car-following parameters were calibrated for both driving conditions under the overcast weather and sun glare to obtain the base models and investigate conventional vehicles' interaction at different desired speeds. Moreover, the development of scenarios required further analysis of mixed traffic includes various combination of conventional and autonomous vehicles.

6.2.5 ***Traffic Operations***

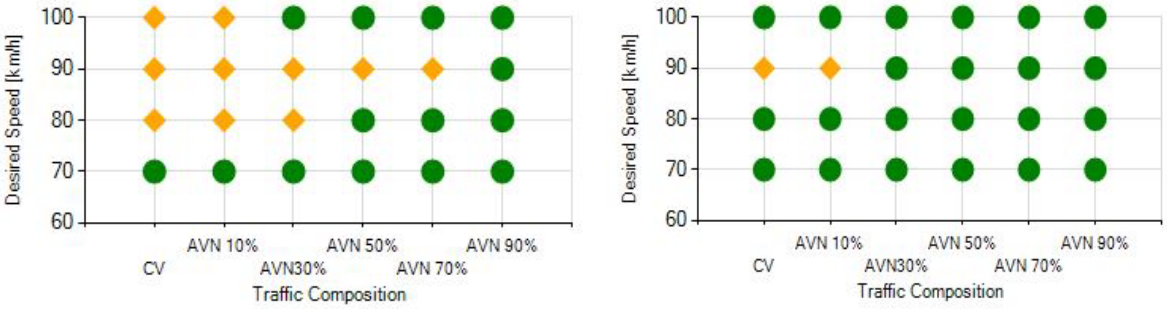
Both calibrated models for different visibility conditions were examined for different traffic flow and speed choices (i.e., 100 to 70 with 10 km/h hierarchical reduction) to evaluate the impact of transition to the use of different type of AVs (i.e., normal and all-knowing) on traffic operation. In this regard, the link behavior function of the segment under the sun glare and connected links were defined based on the calibrated parameters of driving behavior under the sun glare and overcast sky, respectively (Table 8). The model layout of sun glare scenarios was created for a 2200-meter segment of the highway, including 1000 meters without sun glare and a continuous length of 1200 meters under the sun glare.

The traffic operations were evaluated based on the average delays of vehicles along the segment of the case study. The free flow travel time of the study segment was 50, 55, 62, 71 seconds, for 100, 90, 80, and 70 km/h, respectively. Figure (32 - 34) show the average delays in different desired speeds and traffic flow when AVNs penetrate the mixed traffic. While penetration of AVN illustrated a slight improvement in traffic operations, a reduction in desired speed caused a significant reduction in a traffic delay, particularly in near congestion flow on the case study.



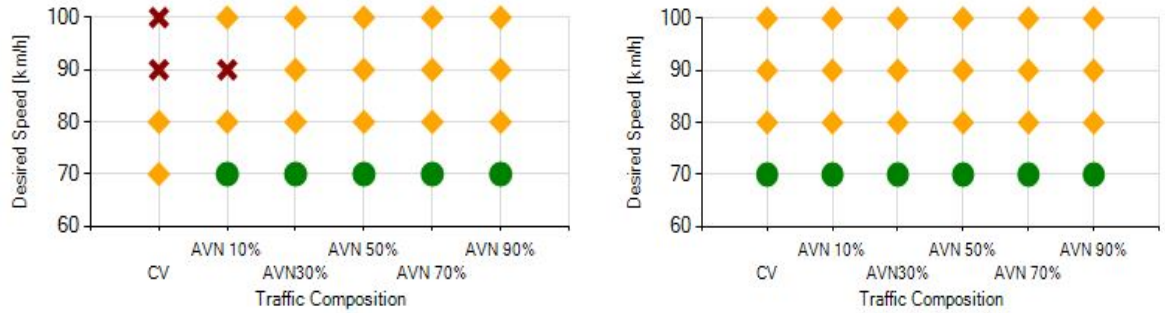
(a) Overcast sky *(b) Sun glare*

Figure 32. Traffic delays of AVNs in light traffic flow and various operational speed (delays $<$ 2: green, delays $<$ 5 seconds: yellow, delays $\geq$ 5 seconds: red)



(a) Overcast sky *(b) Sun glare*

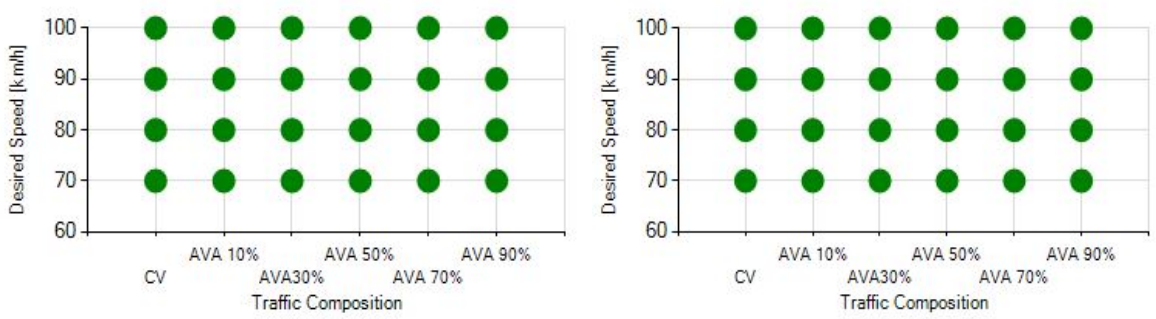
Figure 33. Traffic delays of AVNs in moderate traffic flow and various operational speed (delays $<$ 2: green, delays $<$ 5 seconds: yellow, delays $\geq$ 5 seconds: red)



(a) Overcast sky *(b) Sun glare*

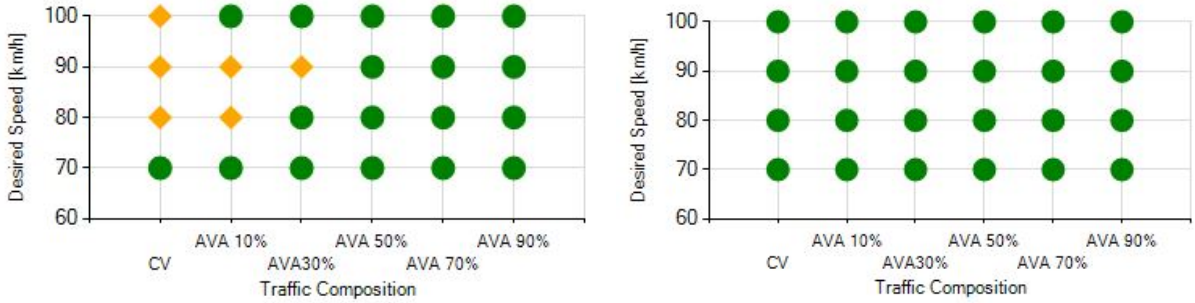
Figure 34. Traffic delays of AVNs in near congestion traffic flow and various operational speed (delays $<$ 2: green, delays $<$ 5 seconds: yellow, delays $\geq$ 5 seconds: red)

Figure (35 - 38) show average delays when AVAs penetrated the mixed traffic flow. Penetration of AVAs into the network showed the same improvement as AVNs. Therefore, compared to the base scenario (i.e., the conventional vehicle with 100 km/h speed), transition to the different AV levels and reduce their speeds greatly benefit the traffic operations.



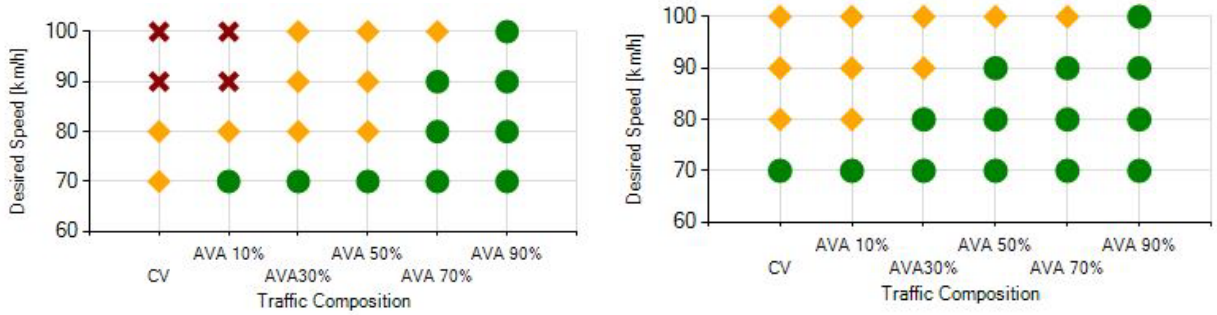
(a) Overcast sky *(b) Sun glare*

Figure 35. Traffic delays of AVAs in light traffic flow and various operational speed (delays $<$ 2: green, delays $<$ 5 seconds: yellow, delays $\geq$ 5 seconds: red)



(a) Overcast sky *(b) Sun glare*

Figure 36. Traffic delays of AVAs in moderate traffic flow and various operational speed (delays $<$ 2: green, delays $<$ 5 seconds: yellow, delays $\geq$ 5 seconds: red)



(a) Overcast sky *(b) Sun glare*

Figure 37. Traffic delays of AVAs in near congestion traffic flow and various operational speed (delays $<$ 2: green, delays $<$ 5 seconds: yellow, delays $\geq$ 5 seconds: red)

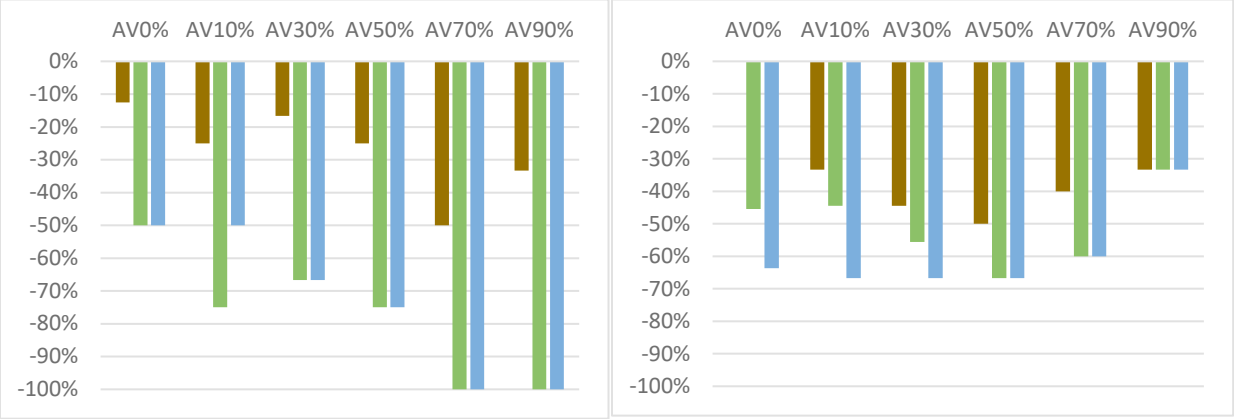
6.2.6 ***Assessment for Rear-End Collisions***

According to the time-to-collision indicator, trajectory files of "Vehicle in Network" were used to analyze vehicles' interaction. Particular TTC_t was selected based on the sum of regular headway for maximum capacity (i.e., 1.5 seconds at 2400 veh/h) and utilized headway for each vehicle type ($CC1$) in the car-following model. Table 9 shows the difference in following distance ($CC0$), headways ($CC1$), and corresponding time-to-collision threshold (TTC_t).

Table 9. Changes in maximum of critical TTC based on vehicle type

Following vehicle type	**CC0** [meters]	**CC1** [seconds]	Capacity headway [seconds]	$\boldsymbol{TTC_t}$ [seconds]
Conventional	1.5	1.3	1.5	2.8
AV Normal	1.5	0.9	1.5	2.4
AV All-knowing	1.0	0.6	1.5	2.1

The conflict analysis results indicated that a 10 km/h decreased AVN vehicles' operational speed eliminates approximately 20% of critical conflict. However, further decrease in the desired speed resulted in significant improvement in the network, more than 40%. Figure (38 – 41) illustrate the percentage reduction in the number of conflicts for different traffic flows and adjusted speeds. The result shows a consistent safety improvement pattern for all AVN ratios on reduced speed scenarios more than 20 km/h.

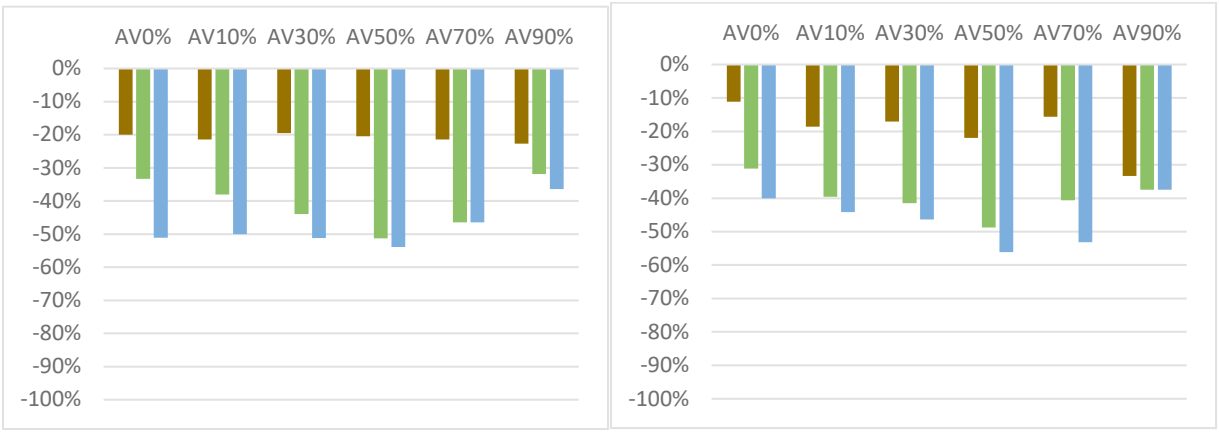


(a) Overcast sky *(b) Sun glare*

Figure 38. Relative change in the number of conflicts in light traffic flow with different speed reduction and AVN penetration ratio
(brown: $-10km/h$, green: $-20km/h$, blue: $-30km/h$)

(a) Overcast sky *(b) Sun glare*

Figure 39. Relative change in the number of conflicts in moderate traffic flow with different speed reduction and AVN penetration ratio
(brown: $-10km/h$, green: $-20km/h$, blue: $-30km/h$)



(a) Overcast sky *(b) Sun glare*

Figure 40. Relative change in the number of conflicts in near congestion traffic flow with different speed reduction and AVN penetration ratio
(brown: $-10km/h$, green: $-20km/h$, blue: $-30km/h$)

On the other hand, the penetration of AVA into the system was evaluated to determine the safety improvement under the reduced operational speed. Figure 41 and 42 show a significant reduction in the number of critical conflicts when vehicles drive with 20 km/h less operational speed in light and moderate traffic flow, respectively. However, reducing operational speed further than 10 km/h shows the same level of reduction in the number of conflicts, approximately 20%, in the near congestion traffic flow (Figure 43). Therefore, reducing the speed of AVNs to 20km/h may be the best optimal safe distance for driving under the sun glare. Although AVAs can select a 20 km/h

less operational speed for light and moderate traffic flow, a reduction of 10 km/h for the near congestion traffic was suitable enough to provide the maximum safety improvement.

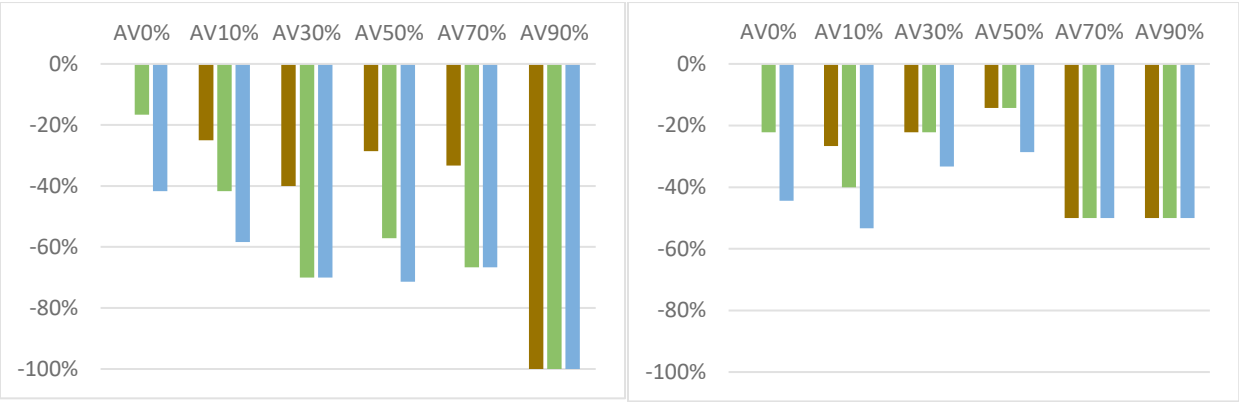
In conclusion, the effective reduction in desired speed of different AVs identified and represents in the table according to changes in the traffic flow.

Table 10. Optimal speed reduction for different level of AVs in the mixed traffic flow

Level of AVs	Light traffic flow	Moderate traffic flow	Near congestion traffic flow
AV Normal	20 km/h	20 km/h	20 km/h
AV All-knowing	20 km/h	20 km/h	10 km/h

(a) Overcast sky *(b) Sun glare*

Figure 41. Relative change in the number of conflicts in light traffic flow with different speed reduction and AVA penetration ratio
(brown: $-10km/h$, green: $-20km/h$, blue: $-30km/h$)



(a) Overcast sky *(b) Sun glare*

Figure 42. Relative change in the number of conflicts in moderate traffic flow with different speed reduction and AVA penetration ratio
(brown: $-10km/h$, green: $-20km/h$, blue: $-30km/h$)

(a) Overcast sky *(b) Sun glare*

Figure 43. Relative change in the number of conflicts in near congestion traffic flow with different speed reduction and AVA penetration ratio (brown: $-10km/h$, green: $-20km/h$, blue: $-30km/h$)

6.3 Collision Avoidance System Under Sun-Glare

The implemented algorithm in the simulation model validated the application's functionality on the case study highway A20 and determined traffic flow metrics. CASUS advisory application works under a three-stage hierarchical framework as a plugin for any car-following models. In the first stage, the application detected the real-time location, speed, and vehicle type to evaluate the occurrence of vision impairment on the A20. In the second stage, the warning message for the beginning of the sun glare and the optimal-safe speed (following the table 10) for passing the segment was transmitted to the vehicle. In the final stage, the safe distance for having a smooth deceleration has been analyzed to enforce the optimal-safe speed to the vehicle with the highest comfort level. Using the same case study in the microsimulation model led us to evaluate the performance and efficiency of CASUS.

6.3.1 *Advance Collision Warning Message*

The script of the CASUS was developed in the VISSIM model, and vehicles followed the same car-following model that has been used in chapter 4. The data and functions contained in the VISSIM application was accessed externally through the COM-Interface. Each vehicle's real-time situation in the network was evaluated through the "Vehicle in Network" attributes in every single time step. CASUS script read the position of each vehicle to identify the location of vision impairment **[45]** in the following road. CASUS also used further attributes of vehicle, such as

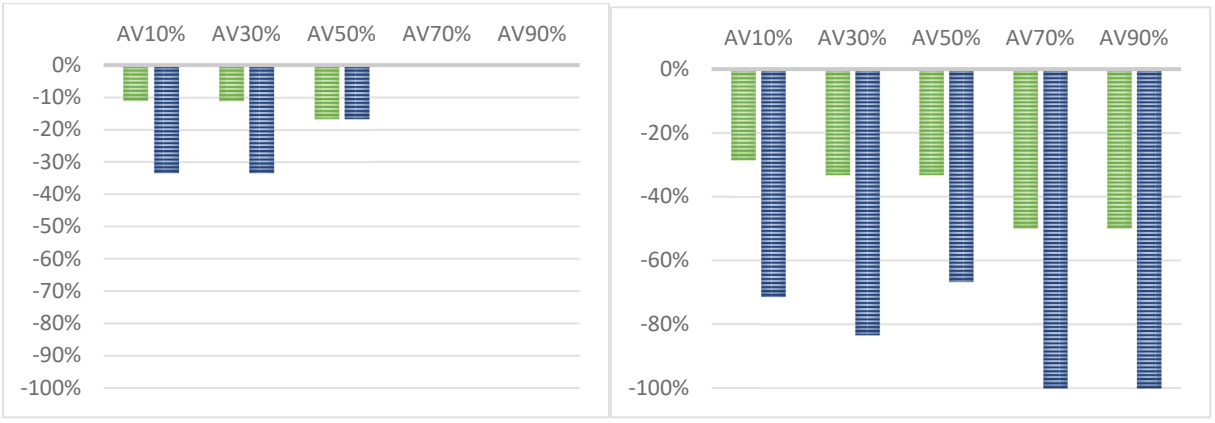
vehicles' position, operational speed, and acceleration to calculate the remaining distance to the incident point.

In terms of detecting the segment under vision impairment, four signal heads were placed in the VISSIM model at the sun glare's beginning and end location on each lane. Figure 44 shows the placement of signal heads on the simulated network when ACWS detected the occurrence of vision impairment on the following road. First, a pair of signal heads at the beginning of the sun glare sent a message to the upstream with the specific range corresponding to each vehicles' velocity (Table 4). AVs were equipped to detect the signal heads and receive the corresponding warning message. The message has remained active until the vehicle enters the segment under sun glare. Then, a new message was sent to the driver to terminate the vision impairment, R_e. All calculation has been done through a script file (see Appendix 9.2.1) in the VISSIM model to simulate receiving warning message by target vehicles in the network.

Figure 44. Positioning vision impairment points on VISSIM model

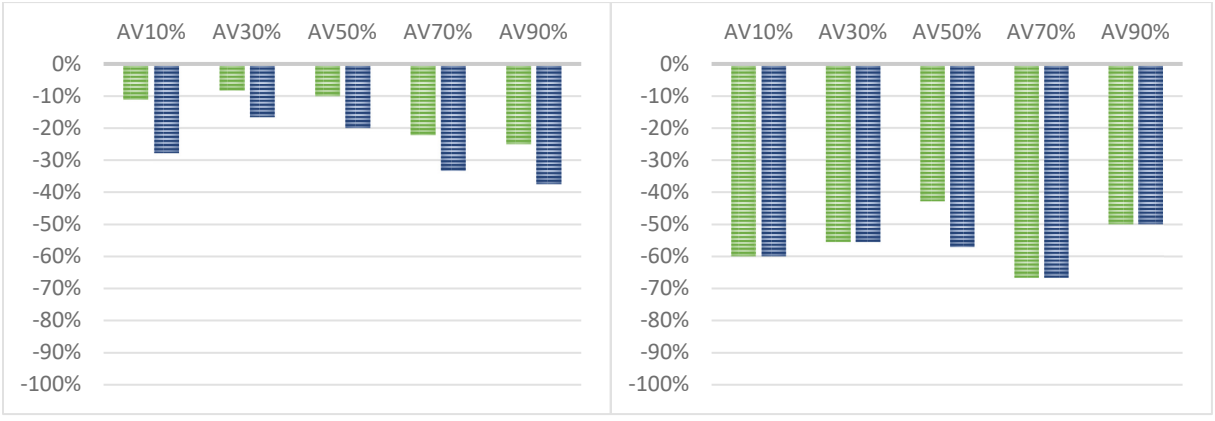
The compliance level of 100% was considered for applying optimal-safe speed during vision impairment. Thus, all AVs began deceleration when receive the warning to continue with the maximum proposed speed. Figure (45-47) show improvement in traffic safety when either AVNs or AVAs chose to accept the new speed limit and decelerate with a standard rate (i.e., 2.5 m/s^2). AVAs provided a significant reduction in the number of conflicts, more than 50%, in light and moderate traffic flow. The penetration of AVAs into the moderate traffic flow provided the same improvement for the reduction of 10km/h (green bars) and 20km/h (blue bars). Thus, speed reduction of 10 km/h is sufficient for AVAs to improve traffic safety.

On the other hand, speed control of AVNs in various traffic flow (Figure 45 - 47) show slight improvement in traffic safety when reducing 20 km/h from the desired speed of AVNs. Therefore, the results revealed while AVNs drive similar to conventional vehicles by providing smaller gaps, using ACWS with larger reduction in their speeds (i.e., 20 km/h) provided a better improvement in road safety. Also, full compliance of AVAs with ACWS provided significant safety improvement in light and moderate traffic flow due to using higher level of autonomy in AVAs for connectivity and visibility of further distances. This improvement is less in near congestion flow because of high density of vehicles on the network and less maneuver of AVAs. However, drivers may reject applying the new speed and keep their previous desired speed. Therefore, a control measure of IBA was tested as the second alternative for safe driving under the sun glare.



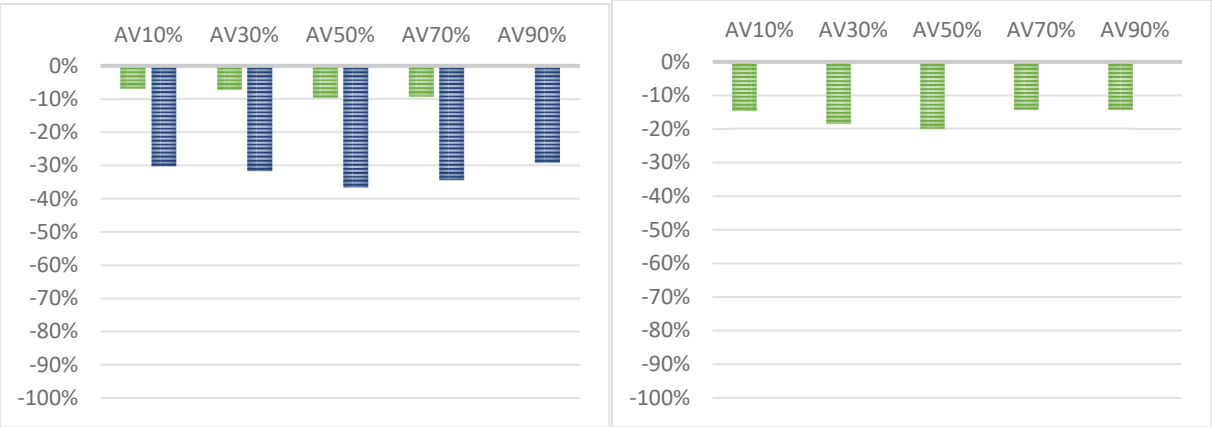
(a) AV Normal *(b) AV All-knowing*

Figure 45. Relative change in the number of conflicts in light traffic flow by applying ACWS with different speed reduction (Green: -10 km/h, Blue: -20 km/h) and AV penetration ratio



(a) AV Normal *(b) AV All-knowing*

Figure 46. Relative change in the number of conflicts in moderate traffic flow by applying ACWS with different speed reduction (Green: -10 km/h, Blue: -20 km/h) and AV penetration ratio



(a) AV Normal *(b) AV All-knowing*

Figure 47. Relative change in the number of conflicts in near congestion traffic flow by applying ACWS with different speed reduction (Green: -10 km/h, Blue: -20 km/h) and AV penetration ratio

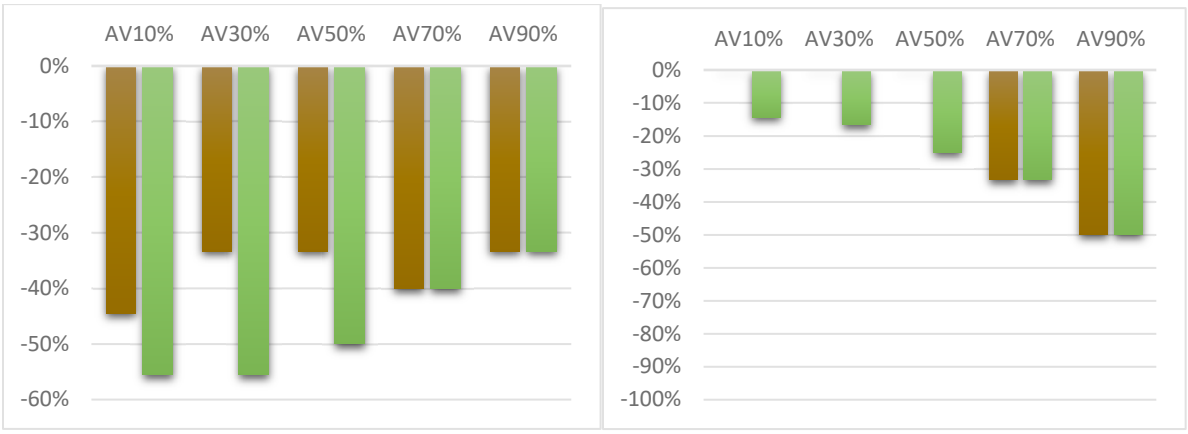
6.3.2 ***Intelligent Braking Assistance***

IBA works as a complementary system for the noncompliance of ACWS warning to mitigate rear end collisions. CASUS supervised operational speed of each equipped vehicle to enforce the safe speed if the vehicle was driven with higher than the optima-safe speed. The minimum distance for the beginning of deceleration was identified according to the calculated distance in equation **[46]**. AVs reduced their speeds with comfortable deceleration rate 2.5 m/s^2 to continue their movement based on applied optimal-safe speed.

In order to simulate the IBA, the COM script of VISSIM detected the signal head at vision impairment location and reduced the speed of vehicle when the vehicle has higher speed than the optimal-safe speed. In this regard, reducing 20 km/h in speed may lead to an abrupt deceleration for followers and increase the risk of rear-end collisions. Thus, a hierarchical speed reduction of 10 km/h was considered when the deviation of vehicle speed and the optimal-safe speed was larger than 10km/h.

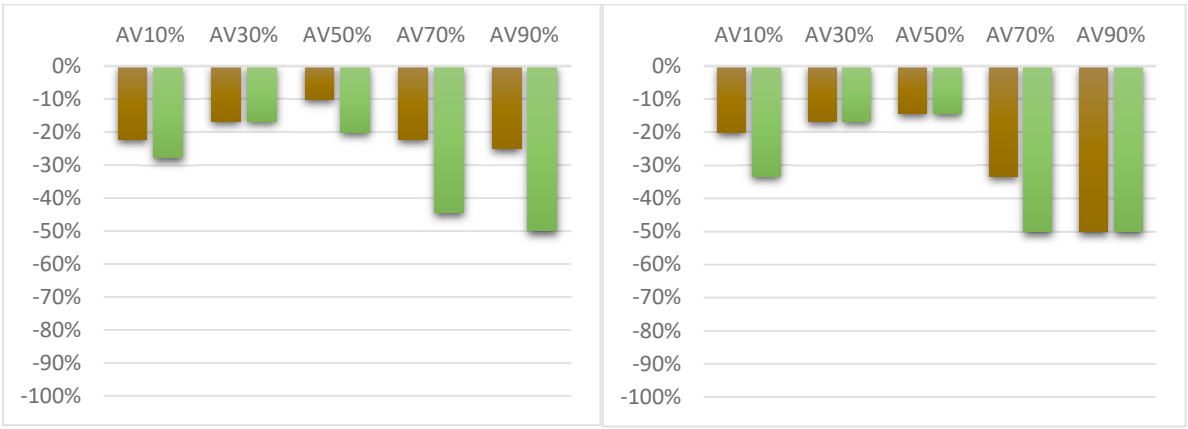
Figure (48 - 50) show the safety improvement in the mixed traffic flow when IBA controlled speeds of AVs. The speed control of AVNs shows a better improvement in traffic safety when vehicles enforced reducing 20 km/h (green bars) from their desired speeds. Vehicles in moderate traffic flow experienced better safety by penetrating more AVNs into the network. As it was expected the crash potential reduction was higher for smaller speeds because reducing in speed of AVNs increased TTC which was exceeded from the threshold, TTC_t . However, the rate of safety

improvement is consistent with more penetration of AVNs into the near congestion traffic flow. This improvement was significant for AVAs when they had more transition into the mixed traffic (Figure 50). AVAs provided an acceptable response to the road safety by enforcing even 10km/h (brown bar) reduction in their desired speed.



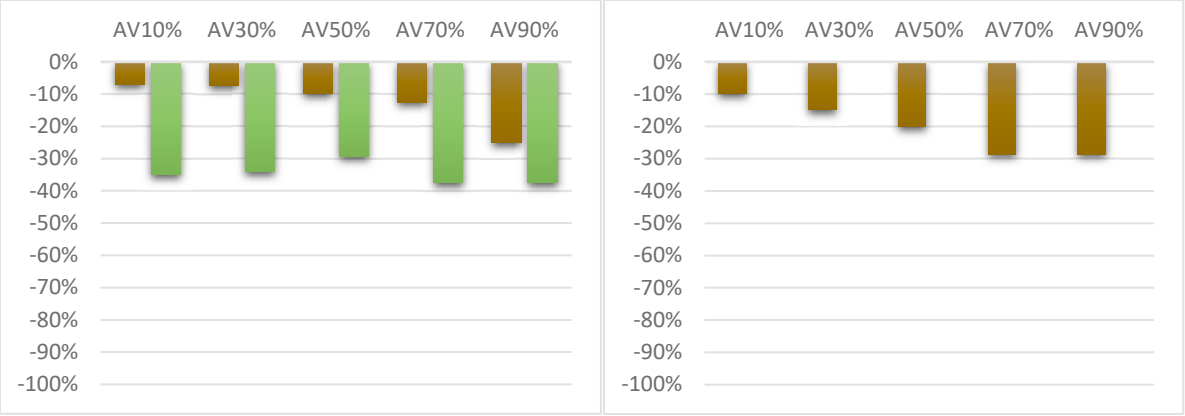
(a) AV Normal *(b) AV All-knowing*

Figure 48.Relative change in the number of conflicts in light traffic flow by applying IBA with different speed reduction (Brown: -10 km/h, Green: -20 km/h) and AV penetration ratio



(a) AV Normal *(b) AV All-knowing*

Figure 49. Relative change in the number of conflicts in moderate traffic flow by applying IBA with different speed reduction (Brown: -10 km/h, Green: -20 km/h) and AV penetration ratio



(a) AV Normal *(b) AV All-knowing*

Figure 50. Relative change in the number of conflicts in near congestion traffic flow by applying IBA with different speed reduction (Brown: -10 km/h, Green: -20 km/h) and AV penetration ratio

6.3.3 ***Concluding Remarks***

The section examined the solution of using the proposed speed control methodology and collision avoidance system, CASUS, to improve traffic safety under the impaired visibility of drivers in the highway case study. An integrated algorithm of the ACWS and IBA in the simulation environment was developed to control speed for different AVs in mixed traffic. Trajectories of vehicles were analyzed to identify the number of critical rear-end conflicts in the network.

The results show a moderate improvement in traffic safety, approximately 30% when normal AVs accepted the proposed 20 km/h reduction in their speed by ACWS. However, AVAs used 10 km/h speed reduction by ACWS to provide the best traffic safety under the sun glare, and only during light traffic preferred to reduce speed 20 km/h.

The results also indicated improved safety by using IBA in terms of noncompliance of drivers from the ACWS warning. This last-minute control of the vehicle before approaching the sun glare segment showed the same improvement, approximately 30%, for the moderate and near congestion traffic flow of AVNs compared with the use of ACWS. However, a small speed reduction of AVNs, 10 km/h, in light traffic improved safety significantly due to having fewer conflicts in low density on roadways. On the other hand, reducing the desired speed of AVAs improved traffic safety; however, it was half of the improvement that ACWS provided in light and moderate traffic flow for AVAs.

Chapter 7 CONCLUSION AND FUTURE DIRECTIONS

This dissertation's main goal was to build a novel advisory system for AVs to understand and predict driver's vision impairment on different types of roadways. The CASUS was developed as a study prototype to determine the impaired visibility locations on urban roadways. Sun glare and contrasting shadows were considered as the common types of vision impairment. The contrasting shadow was clustered based on sunlight's occlusion during a sun glare event by surface objects. The case study of the urban roadways in Montreal was used as a pilot project. The results revealed a dynamic situation of sun glare occurrence within the urban area. The most occurrence of sun glare frequently occurred at the middle of the roadways and inside the wide intersections.

This study also investigated the impact of sun glare on driving behaviors. A homogenous traffic flow on a straight freeway was selected to limit vehicles' interaction with the environment and only consider the impact of vision impairment. A driving simulator study was conducted to model driving under sun glare and investigated car-following parameters' sensitivity. The traffic simulator generated the trajectory files for three different vehicle automation levels (i.e., conventional, normal AV, all-knowing AV) and various desired speed reductions to determine the conflict frequencies. A safety indicator of TTC was defined to determine critical rear-end conflicts by evaluating the difference in velocity and acceleration for each interaction of vehicles. The results indicated a proposed speed adjustment for driving a different type of AVs under the sun glare occurrence in highways. The conflict analysis of the simulation results revealed a significant improvement in road safety by using optimal-safe speed for each type of AVs.

The CASUS was introduced as an innovative method to reduce rear-end crashes from the threat of the temporary blindness of drivers due to sun exposure. Instead of focusing on traditional speed limit signs, the CASUS provides an integrated combination of two different driving assistance conditions, ACWS and IBA, and acts based on microsimulation for safe stop driving under the sun glare. The result revealed a moderate improvement in traffic safety by ensuring AVNs' drivers using the proposed speed reduction by ACWS. However, AVAs obtained greater improvement in road safety when vehicles comply 100% with the proposed speed of ACWS. The last-minute control of AVAs with IBA provided half of the improvement that ACWS obtained. However, AVNs responded better to road safety when the IBA controlled their speed through the sun glare segment.

The Presented CASUS can help road designers determine the intensity of vision impairment occurrence on predesigned road's alignment. This study's most significant contribution is to improve safety on the existing roads with the highest number of vision impairments. The study also helps researchers and car manufacturers identify real-time hazardous traffic locations and mitigate crash potentials by investing in the transition to different AVs' autonomous levels.

This study considered the threat of vision impairment based on human drivers and evaluated the transition of AVs into the network for safe driving under the sun glare exposure. In the future, traffic composition may comprise only fully connected AVs and sensory system be a replacement to eliminate the deficiency of vision impairment on the road. In this regard, the defectiveness of the sensory system for bicycles and pedestrians needs to be investigated by using the CASUS application to provide the optimal speed based on the type and quality of on-vehicle sensors.

Implementing the advisory system in the simulation environment is transferable to using the model for any other types of uncertain incidents on the road. Also, different car-following models can be applied through the created COM script to optimize the vehicle's maneuver and consider the passengers' comfort.

In this thesis, it was not possible to conduct a real-world experimental test for the implementation of CASUS due to the limited availability of autonomous vehicles. Therefore, further research is recommended for implementing CASUS in an advanced driving simulator to analyze AVs' safety improvement over human drivers. Moreover, the methodology investigated the rear-end critical conflicts, and it could be developed and validated for a case study with a large number of lateral movements.

This thesis considered a fully- or non-compliance of drivers with the proposed speed of ACWS due to the focus of the study on vehicle communication and control. Further compliance levels will be analyzed to replicate the responses of drivers to the system. Considering different compliance level may develop the length of initial warning range in the application of ACWS.

Additionally, the control system of IBA was used to enforce speed reduction when AVs were approaching the sun glare segment. The future direction of IBA modeling is to investigate the traffic safety effects of incorrect estimation of safe following speed and distraction of drivers when

approaching the contrasting shadow. Further control measures can be added to this model by introducing a combination of different vision impairment on the roadways.

Chapter 8 References

AASHTO, A. (2001). Policy on geometric design of highways and streets. *American Association of State Highway and Transportation Officials, Washington, DC*, *1*(990), 158.

AASHTO, T. (2011). 342-11. 2011. *Standard Method of Test for Determining Dynamic Modulus of Hot Mix Asphalt (HMA)*.

Abdelhamid, S., Hassanein, H. S., & Takahara, G. (2014). Vehicle as a mobile sensor. *Procedia Computer Science*, *34*, 286–295.

Administration, N. H. T. S. (2008). National motor vehicle crash causation survey: Report to congress. *National Highway Traffic Safety Administration Technical Report DOT HS*, *811*, 59.

Aghabayk, K., Sarvi, M., Young, W., & Kautzsch, L. (2013). A novel methodology for evolutionary calibration of Vissim by multi-threading. In *Australasian Transport Research Forum* (pp. 1–15).

Ahmed, M., Ghasemzadeh, A., Hammit, B. E., Khan, N., Das, A., Ali, E., … Eldeeb, H. (2018). *Driver performance and behavior in adverse weather conditions: An investigation using the SHRP2 naturalistic driving study data—Phase 2*. Wyoming. Dept. of Transportation.

Ahn, Y., Haban, P., Gross, J. S., Shane, A., De Moe, J., & Julian, B. W. (2016, November 1). Autonomous vehicle overall interior. Google Patents.

Andrey, J. C., Mills, B. E., & Vandermolen, J. (2001). Weather information and road safety.

Barbaresso, J., Cordahi, G., Garcia, D., Hill, C., Jendzejec, A., Wright, K., & Hamilton, B. A. (2014). *USDOT's Intelligent Transportation Systems (ITS) ITS strategic plan, 2015-2019.* United States. Department of Transportation. Intelligent Transportation ….

Borrough, P. (1997). Variable speed limits reduce crashes significantly in the UK. *The Urban Transportation Monitor*.

Bose, A., & Ioannou, P. A. (2003). Analysis of traffic flow with mixed manual and semiautomated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *4*(4), 173–188.

Brackstone, M., & McDonald, M. (1999). Car-following: a historical review. *Transportation*

Research Part F: Traffic Psychology and Behaviour, *2*(4), 181–196.

Browman, C. P., & Goldstein, L. (1989). Articulatory gestures as phonological units. *Phonology*, 201–251.

Brunson, S. J., Kyle, E. M., Phamdo, N. C., & Preziotti, G. R. (2002). *Alert algorithm development program: NHTSA rear-end collision alert algorithm*.

Burg, A. (1966). Visual acuity as measured by dynamic and static tests: A comparative evaluation. *Journal of Applied Psychology*, *50*(6), 460.

Burgett, A. L., Carter, A., Miller, R. J., Najm, W. G., & Smith, D. L. (1998). A collision warning algorithm for rear-end collisions. In *16th International Technical Conference on Enhanced Safety of Vehicles (ESV), Windsor, Canada*.

Chandler, R. E., Herman, R., & Montroll, E. W. (1958). Traffic dynamics: studies in car following. *Operations Research*, *6*(2), 165–184.

Charette, R. N. (2009). This car runs on code. *IEEE Spectrum*, *46*(3), 3.

Chen, C., Liu, L., Qiu, T., Ren, Z., Hu, J., & Ti, F. (2018). Driver's intention identification and risk evaluation at intersections in the Internet of vehicles. *IEEE Internet of Things Journal*, *5*(3), 1575–1587.

Chrysler, S. T., Danielson, S. M., & Kirby, V. M. (1996). Age differences in visual abilities in nighttime driving field conditions. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 40, pp. 923–927). SAGE Publications Sage CA: Los Angeles, CA.

Churchill, A. M., Tripodis, Y., & Lovell, D. J. (2012). Sun glare impacts on freeway congestion: geometric model and empirical analysis. *Journal of Transportation Engineering*, *138*(10), 1196–1204. https://doi.org/http://dx.doi.org/10.1061/(ASCE)TE.1943-5436.0000418

Committee, S. A. E. O.-R. A. V. S. (2014). Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems. *SAE Standard J*, *3016*, 1–16.

Cornwall, C., Horiuchi, A., Lehman, C. 2011. S. calculator. N. E. S. R. L. (n.d.). No Title. Retrieved from https://www.esrl.noaa.gov/gmd/grad/solcalc/azel.html

De Charette, R. (2012). Vision Algorithms for Rain and Traffic Lights in Driver Assistance Systems.

De Charette, R., Wiedemann, R., Barbaresso, J., Cordahi, G., Garcia, D., Hill, C., … Helbing, D. (2013). Energy and emission benefit comparison of stationary and in-vehicle advanced driving alert systems. *Transportation Research Record*, *2122*(1), 143–163.

Delphi, L., & Sepehri, H. (2016). Apple pectin: A natural source for cancer suppression in 4T1 breast cancer cells in vitro and express p53 in mouse bearing 4T1 cancer tumors, in vivo. *Biomedicine & Pharmacotherapy*, *84*, 637–644.

Department for Transport, G. M. H. (DTGMH). (2012). *Road Accidents Great Britain 2012*. London.

Dixit, V. V, Chand, S., & Nair, D. J. (2016). Autonomous vehicles: disengagements, accidents and reaction times. *PLoS One*, *11*(12), e0168054.

Dowling, R., Skabardonis, A., & Alexiadis, V. (2004). *Traffic analysis toolbox, volume III: Guidelines for applying traffic microsimulation modeling software*. United States. Federal Highway Administration. Office of Operations.

Drake, J. S. (1967). A statistical analysis of speed density hypothesis. *HRR*, *154*, 53–87.

Duran, D. R., Robinson, E., Kornecki, A. J., & Zalewski, J. (2013). Safety analysis of Autonomous Ground Vehicle optical systems: Bayesian belief networks approach. In *2013 Federated Conference on Computer Science and Information Systems* (pp. 1419–1425). IEEE.

Edie, L. C., Gazis, D. C., Helly, W., Herman, R., & Rothery, R. (1965). Letter to the Editor—Third International Symposium on the Theory of Traffic Flow. *Operations Research*, *13*(6), 1045–1051.

Elliott, D., Whitaker, D., & MacVeigh, D. (1990). Neural contribution to spatiotemporal contrast sensitivity decline in healthy ageing eyes. *Vision Research*, *30*(4), 541–547.

Fraczek, W. (2003). Mean sea level, GPS, and the geoid. *ArcUsers Online*.

Fritzsche, H.-T., & Ag, D. (1994). A model for traffic simulation. *Traffic Engineering+ Control*, *35*(5), 317–321.

Fry, G. A. (1955). *Physiological bases of disability glare*.

Garber, N. J., Miller, J. S., Sun, X., & Yuan, B. (2006). Safety impacts of differential speed limits for trucks and passenger cars on rural interstate highways: A modified empirical Bayes approach. *Journal of Transportation Engineering*, *132*(1), 19–29.

Gazis, D. C., Herman, R., & Rothery, R. W. (1961). Nonlinear follow-the-leader models of traffic flow. *Operations Research*, *9*(4), 545–567.

Genders, W., & Razavi, S. (2016a). Using a deep reinforcement learning agent for traffic signal control. *ArXiv Preprint ArXiv:1611.01142*.

Genders, W., & Razavi, S. N. (2016b). Impact of connected vehicle on work zone network safety through dynamic route guidance. *Journal of Computing in Civil Engineering*, *30*(2), 4015020.

Gettman, D., & Head, L. (2003). Surrogate safety measures from traffic simulation models. *Transportation Research Record*, *1840*(1), 104–115.

Ghanim, M. S., & Shaaban, K. (2019). A case study for surrogate safety assessment model in predicting real-life conflicts. *Arabian Journal for Science and Engineering*, *44*(5), 4225–4231.

Gipps, P. G. (1981). A behavioural car-following model for computer simulation. *Transportation Research Part B: Methodological*, *15*(2), 105–111.

Gkikas, N. (2016). *Automotive ergonomics: driver-vehicle interaction*. CRC Press.

Goel, P., Dedeoglu, G., Roumeliotis, S. I., & Sukhatme, G. S. (2000). Fault detection and identification in a mobile robot using multiple model estimation and neural network. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)* (Vol. 3, pp. 2302–2309). IEEE.

Gong, H., Walker, H. K., Hall, W. D., & Hurst, J. W. (1990). Clinical Methods: The History. *Physical, and Laboratory Examinations*, *3*, 60.

Goodwin, L. C. (2002). Weather impacts on arterial traffic flow. *Federal Highway Administration, Washington, Prepared for Road Weather Management Program*.

Goudie, D. W., Bowler, J. J., Brown, C. A., Heinrichs, B. E., & Siegmund, G. P. (2000). Tire friction during locked wheel braking. *SAE Transactions*, 1780–1789.

Graham, R. (1999). Use of auditory icons as emergency warnings: evaluation within a vehicle collision avoidance application. *Ergonomics*, *42*(9), 1233–1248.

Greenshields, B. D., Bibbins, J. R., Channing, W. S., & Miller, H. H. (1935). A study of traffic capacity. In *Highway research board proceedings* (Vol. 1935). National Research Council (USA), Highway Research Board.

Grimm, R., & Langenbeck, B. (2008, June 3). Brake assembly, particularly for a hydraulic wheel drive. Google Patents.

Guerrero-Ibáñez, J., Zeadally, S., & Contreras-Castillo, J. (2018). Sensor technologies for intelligent transportation systems. *Sensors*, *18*(4), 1212.

Habtemichael, F. G., & de Picado Santos, L. (2013). Safety and operational benefits of variable speed limits under different traffic conditions and driver compliance levels. *Transportation Research Record*, *2386*(1), 7–15.

Hagita, K., & Mori, K. (2014). The effect of sun glare on traffic accidents in Chiba prefecture, Japan. *Asian Transport Studies*, *3*(2), 205–219.

Hall, F. L. (2001). Chapter 2: Traffic stream characteristics. *Traffic Flow Theory: A State of the Art Report, Edited by: Gartner, N., Messer, C., and Rathi, A., Available at: Https://Www. Academia. Edu/25432198/Traffic_Flow_Theory_A_State_of_the_Art_Report (Last Access: March 2019)*.

Hamdar, S. H., Qin, L., & Talebpour, A. (2016). Weather and road geometry impact on longitudinal driving behavior: Exploratory analysis using an empirically supported acceleration modeling framework. *Transportation Research Part C: Emerging Technologies*, *67*, 193–213.

Hammond Jr, B. R., Wooten, B. R., & Curran-Celentano, J. (2001). Carotenoids in the retina and lens: possible acute and chronic effects on human visual performance. *Archives of Biochemistry and Biophysics*, *385*(1), 41–46.

Hanken, A., & Rockwell, T. H. (1967). A MODEL OF CAR FOLLOWING DERIVED EMPIRICALLY BY PIECE-WISE REGRESSION ANALYSIS. IN VEHICULAR TRAFFIC SCIENCE. In *Proceedings of the Third International Symposium on the Theory of Traffic FlowOperations Research Society of America*.

Happian-Smith, J. (2001). *An introduction to modern vehicle design*. Elsevier.

Hawkins, R. K. (1988). Motorway traffic behaviour in reduced visibility conditions. In *Vision in Vehicles II. Second International Conference on Vision in VehiclesApplied Vision AssociationErgonomics SocietyAssociation of Optometrists*.

Henclewood, D., Suh, W., Rodgers, M. O., Fujimoto, R., & Hunter, M. P. (2017). A calibration procedure for increasing the accuracy of microscopic traffic simulation models. *Simulation*, *93*(1), 35–47.

Ho, C. K., Ghanbari, C. M., & Diver, R. B. (2011). Methodology to assess potential glint and glare hazards from concentrating solar power plants: analytical models and experimental validation. *Journal of Solar Energy Engineering*, *133*(3), 31021. https://doi.org/http://dx.doi.org/10.1115/1.4004349

Hoedemaeker, M., & Brookhuis, K. A. (1998). Behavioural adaptation to driving with an adaptive cruise control (ACC). *Transportation Research Part F: Traffic Psychology and Behaviour*, *1*(2), 95–106.

Hoffman, J., Lee, J. D., & Hayes, E. M. (2003). Driver preference of collision warning strategy and modality.

Hogema, J. H., & Janssen, W. H. (1996). *Effects of intelligent cruise control on driving behaviour: a simulator study*. TNO.

Homburger, W. S., Keefer, L. E., & Mcgrath, W. R. (1982). *Transportation and traffic engineering handbook*.

Hoogendoorn, S. P., & Hoogendoorn, R. (2010). Generic calibration framework for joint estimation of car-following models by using microscopic data. *Transportation Research Record*, *2188*(1), 37–45.

Hörl, S., Ciari, F., & Axhausen, K. W. (2016). Recent perspectives on the impact of autonomous vehicles. *Arbeitsberichte Verkehrs-Und Raumplanung*, *1216*.

Horowitz, A. D., & Dingus, T. A. (1992). Warning signal design: A key human factors issue in an in-vehicle front-to-rear-end collision warning system. In *Proceedings of the Human Factors Society Annual Meeting* (Vol. 36, pp. 1011–1013). SAGE Publications Sage CA: Los Angeles, CA.

Hourdakis, J., Michalopoulos, P. G., & Kottommannil, J. (2003). Practical procedure for calibrating microscopic traffic simulation models. *Transportation Research Record*, *1852*(1), 130–139.

Hurwitz, J. B., & Wheatley, D. J. (2001). Driver choice of headway with auditory warnings. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 45, pp. 1637–1640). SAGE Publications Sage CA: Los Angeles, CA.

Ibrahim, A. T., & Hall, F. L. (1994). *Effect of adverse weather conditions on speed-flow-occupancy relationships*.

Inanici, M. N. (2006). Evaluation of high dynamic range photography as a luminance data acquisition system. *Lighting Research & Technology*, *38*(2), 123–134.

Islam, M. T., Hadiuzzaman, M., Fang, J., Qiu, T. Z., & El-Basyouny, K. (2013). Assessing mobility and safety impacts of a variable speed limit control strategy. *Transportation Research Record*, *2364*(1), 1–11.

Jacobs, A. (2007). High dynamic range imaging and its application in building research. *Advances in Building Energy Research*, *1*(1), 177–202.

Jakobsson, L. (2004). Evaluation of impact severity measures for AIS 1 neck injuries in frontal impacts using crash recorder data. *International Journal of Crashworthiness*, *9*(1), 105–111.

Jeong, E., Oh, C., & Lee, S. (2017). Is vehicle automation enough to prevent crashes? Role of traffic operations in automated driving environments for traffic safety. *Accident Analysis & Prevention*, *104*, 115–124.

Jurado-Piña, R., & Mayora, J. M. P. (2009). Methodology to predict driver vision impairment

situations caused by sun glare. *Transportation Research Record*, *2120*(1), 12–17. https://doi.org/http://dx.doi.org/10.3141/2120-02

Karimi, M., Miriestahbanati, M., Esmaeeli, H., & Alecsandru, C. (2019). *Multi-Objective Stochastic Optimization Algorithms to Calibrate Microsimulation Models*. *Transportation Research Record* (Vol. 2673). https://doi.org/10.1177/0361198119838260

Karimi, Mohammad, Miriestahbanati, M., Esmaeeli, H., & Alecsandru, C. (2019). Multi-Objective Stochastic Optimization Algorithms to Calibrate Microsimulation Models. *Transportation Research Record*, 0361198119838260.

Kato, S., Tsugawa, S., Tokuda, K., Matsui, T., & Fujii, H. (2002). Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications. *IEEE Transactions on Intelligent Transportation Systems*, *3*(3), 155–161.

Kesting, A., Treiber, M., & Helbing, D. (2009). Agents for traffic simulation. *Multi-Agent Systems: Simulation and Applications*, 325–356.

Kiefer, R J, Cassar, M. T., Flannagan, C. A., LeBlanc, D. J., Palmer, M. D., Deering, R. K., & Shulman, M. A. (2003). *Forward collision warning requirements project: refining the CAMP crash alert timing approach by examining" last second" braking and lane change maneuvers under various kinematic conditions*. United States. National Highway Traffic Safety Administration.

Kiefer, R J, LeBlanc, D., Palmer, M. D., Salinger, J., Deering, R. K., & Shulman, M. (1999). *Development and validation of functional definitions and evaluation procedures for collision warning/avoidance systems*. United States. Department of Transportation. National Highway Traffic Safety ….

Kiefer, Raymond J, LeBlanc, D. J., & Flannagan, C. A. (2005). Developing an inverse time-to-collision crash alert timing approach based on drivers' last-second braking and steering judgments. *Accident Analysis & Prevention*, *37*(2), 295–303.

Kim, J., & Mahmassani, H. S. (2011). Correlated parameters in driving behavior models: Car-following example and implications for traffic microsimulation. *Transportation Research Record*, *2249*(1), 62–77.

Kockelman, K., Avery, P., Bansal, P., Boyles, S. D., Bujanovic, P., Choudhary, T., … Helsel, J. (2016). *Implications of connected and automated vehicles on the safety and operations of roadway networks: A final report*.

Kouchaki, S., Roshani, H., Prozzi, J. A., Garcia, N. Z., & Hernandez, J. B. (2018). Field investigation of relationship between pavement surface texture and friction. *Transportation Research Record*, *2672*(40), 395–407.

Krauß, S., Wagner, P., & Gawron, C. (1997). Metastable states in a microscopic model of traffic flow. *Physical Review E*, *55*(5), 5597.

Krishnan, H., Gibb, S., Steinfeld, A., & Shladover, S. (2001). Rear-End Collision–Warning System: Design and Evaluation via Simulation. *Transportation Research Record*, *1759*(1), 52–60.

Kuang, Y., Qu, X., & Wang, S. (2015). A tree-structured crash surrogate measure for freeways. *Accident Analysis & Prevention*, *77*, 137–148.

Kurker, M., Fournier, C., Zhao, Q., Hakimi, S., Qi, Y., Tang, S., … Machemehl, R. (2014). *Minimizing user delay and crash potential through highway work zone planning.* Texas. Dept. of Transportation. Research and Technology Implementation Office.

Kuusniemi, H., Lachapelle, G., & Takala, J. H. (2004). Position and velocity reliability testing in degraded GPS signal environments. *GPS Solutions*, *8*(4), 226–237.

Laboratories, E. S. R., & Laboratory, G. M. (n.d.). noaa sun position algorithm. Retrieved from https://www.esrl.noaa.gov/gmd/grad/solcalc/solareqns.PDF

Law, A. M. (2015). Statistical analysis of simulation output data: the practical state of the art. In *2015 Winter Simulation Conference (WSC)* (pp. 1810–1819). IEEE.

Lee, C., Hellinga, B., & Saccomanno, F. (2003). Real-time crash prediction model for application to crash prevention in freeway traffic. *Transportation Research Record: Journal of the Transportation Research Board*, (1840), 67–77.

Lee, H., & Mousa, A. M. (1996). GPS travelling wave fault locator systems: investigation into the anomalous measurements related to lightning strikes. *IEEE Transactions on Power Delivery*,

11(3), 1214–1223.

Lee, J. D., Hoffman, J. D., & Hayes, E. (2004). Collision warning design to mitigate driver distraction. In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems* (pp. 65–72).

Li, Q., Qiao, F., & Yu, L. (2016). Implications of advanced warning messages on eliminating sun glare disturbances at signalized intersections. *Journal of Traffic and Transportation Engineering (English Edition)*, *3*(4), 296–307.

Lidbe, A. D., Hainen, A. M., & Jones, S. L. (2017). Comparative study of simulated annealing, tabu search, and the genetic algorithm for calibration of the microsimulation model. *Simulation*, *93*(1), 21–33.

Lieberman, E., & Rathi, A. K. (2001). Traffic Simulation in Traffic Flow Theory A State-Of-The-Art Report. *Oak Ridge National Laboratory, Oak Ridge, Tn*.

Litman, T. (2017). *Autonomous vehicle implementation predictions*. Victoria Transport Policy Institute Victoria, Canada.

Littlefair, P. J. (1985). The luminous efficacy of daylight: a review. *Lighting Research & Technology*, *17*(4), 162–182.

Littlefair, P. J. (1988). Measurements of the luminous efficacy of daylight. *Lighting Research & Technology*, *20*(4), 177–188.

Macario, G., Torchiano, M., & Violante, M. (2009). An in-vehicle infotainment software architecture based on google android. In *2009 IEEE International Symposium on Industrial Embedded Systems* (pp. 257–260). IEEE.

Mace, D., Garvey, P., Porter, R. J., Schwab, R., & Adrian, W. (2001). Countermeasures for reducing the effects of headlight glare. *American Automobile Association Foundation for Traffic Safety*.

Maerivoet, S., & Moor, B. D. (2008). Traffic theory'. *ArXiv Preprint Physics/0507126*.

Manual, H. C. (2000). Highway capacity manual. *Washington, DC*, *2*, 1.

Maria, A. (1997). Introduction to modeling and simulation. In *Proceedings of the 29th conference*

on Winter simulation (pp. 7–13).

Martin, P. T., Kalyani, V. C., & Stavanovic, A. (2003). *Evaluation of advance warning signals on high speed signalized intersections*. Mountain-Plains Consortium.

Meeus, J. (1988). Astronomical Formulae For Calculation. Willmann-Bell. *Inc., Printed in the United States of America*.

Merat, N., & Lee, J. D. (2012). Preface to the special section on human factors and automation in vehicles: Designing highly automated vehicles with the driver in mind. *Human Factors*, *54*(5), 681–686.

Michalsky, J. J. (1988). The astronomical almanac's algorithm for approximate solar position (1950–2050). *Solar Energy*, *40*(3), 227–235. https://doi.org/https://doi.org/10.1016/0038-092X(88)90045-X

Minderhoud, M. M., & Bovy, P. H. L. (2001). Extended time-to-collision measures for road traffic safety assessment. *Accident Analysis & Prevention*, *33*(1), 89–97.

Montreal, V. de. (2014). [dataset] Géobase double - côtés de rue du réseau routier. Retrieved August 20, 2010, from http://donnees.ville.montreal.qc.ca/dataset/geobase-double

Montreal, V. de. (2015). [dataset] Modèle numérique de terrain. Retrieved September 20, 2001, from http://donnees.ville.montreal.qc.ca/dataset/modele-numerique-de-terrain-mnt

Montreal, V. de. (2019a). [dataset] LiDAR aérien 2015. Retrieved September 20, 2002, from http://donnees.ville.montreal.qc.ca/dataset/lidar-aerien-2015

Montreal, V. de. (2019b). [dataset] Modèle numérique de surface (MNS). Retrieved September 20, 2001, from http://donnees.ville.montreal.qc.ca/dataset/modele-numerique-de-surface-mns

Mössinger, J. (2010). Software in automotive systems. *IEEE Software*, *27*(2), 92–94.

Mössinger, J., Macario, G., Torchiano, M., Violante, M., Litman, T., Saffarian, M., … Bertini, R. L. (2013). Vehicle dynamics and control. *Transportation Research Record*, *4*(1), 31–44.

Muley, D., Ghanim, M., & Kharbeche, M. (2018). Prediction of Traffic Conflicts at Signalized Intersections using SSAM. *Procedia Computer Science*, *130*, 255–262.

Muneer, T. (1995). Solar irradiance and illuminance models for Japan II: luminous efficacies. *International Journal of Lighting Research and Technology*, *27*(4), 223–230.

Muneer, Tariq. (2007). *Solar radiation and daylight models*. Routledge.

Najm, W., Stearns, M., Howarth, H., Koopmann, J., & Hitz, J. S. (2006). *Evaluation of an automotive rear-end collision avoidance system*. United States. Department of Transportation. National Highway Traffic Safety ….

National renewable Energy Laboratory. (2013). System Advisor Model. Retrieved from https://sam.nrel.gov/node/76071

Nilsson, L. (1993). Behavioural research in an advanced driving simulator-experiences of the VTI system. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 37, pp. 612–616). SAGE Publications Sage CA: Los Angeles, CA.

Olia, A., Abdelgawad, H., Abdulhai, B., & Razavi, S. N. (2016). Assessing the potential impacts of connected vehicles: mobility, environmental, and safety perspectives. *Journal of Intelligent Transportation Systems*, *20*(3), 229–243.

Olson, P. L., & Sivak, M. (1986). Perception-response time to unexpected roadway hazards. *Human Factors*, *28*(1), 91–96.

Olstam, J. J., & Tapani, A. (2004). *Comparison of Car-following models* (Vol. 960). Swedish National Road and Transport Research Institute Linköping.

Ossen, S., & Hoogendoorn, S. P. (2005). Car-following behavior analysis from microscopic trajectory data. *Transportation Research Record*, *1934*(1), 13–21.

Ossen, S., & Hoogendoorn, S. P. (2008). Validity of trajectory-based calibration approach of car-following models in presence of measurement errors. *Transportation Research Record*, *2088*(1), 117–125.

Ossen, S., & Hoogendoorn, S. P. (2011). Heterogeneity in car-following behavior: Theory and empirics. *Transportation Research Part C: Emerging Technologies*, *19*(2), 182–195.

Papadimitriou, S., & Choudhury, C. F. (2017). Transferability of car-following models between driving simulator and field traffic. *Transportation Research Record*, *2623*(1), 60–72.

Pauwelussen, J., & Minderhoud, M. (2008). The effects of deactivation and (re) activation of ACC on driver behaviour analyzed in real traffic. In *2008 IEEE Intelligent Vehicles Symposium* (pp. 257–262). IEEE.

Perez, R., Ineichen, P., Seals, R., Michalsky, J., & Stewart, R. (1990). Modeling daylight availability and irradiance components from direct and global irradiance. *Solar Energy*, *44*(5), 271–289.

Pipes, L. A. (1953). An operational analysis of traffic dynamics. *Journal of Applied Physics*, *24*(3), 274–281.

Pisano, P. A., Goodwin, L. C., & Rossetti, M. A. (2008). US highway crashes in adverse road weather conditions. In *24th Conference on International Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology, New Orleans, LA*. https://doi.org/https://www.fhwa.dot.gov/exit.cfm?link=http://ams.confex.com/ams/pdfpapers/133554.pdf

Poczter, S. L., & Jankovic, L. M. (2014). The Google Car: driving toward a better future? *Journal of Business Case Studies (JBCS)*, *10*(1), 7–14.

Punzo, V., Ciuffo, B., & Montanino, M. (2012). Can results of car-following model calibration based on trajectory data be trusted? *Transportation Research Record*, *2315*(1), 11–24.

Punzo, V., & Simonelli, F. (2005). Analysis and comparison of microscopic traffic flow models with real traffic microscopic data. *Transportation Research Record*, *1934*(1), 53–63.

Rajamani, R., Tan, H.-S., Law, B. K., & Zhang, W.-B. (2000). Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons. *IEEE Transactions on Control Systems Technology*, *8*(4), 695–708.

Ranney, T. A., Garrott, W. R., & Goodman, M. J. (2001). *NHTSA driver distraction research: Past, present, and future*. SAE Technical Paper.

Reidenbach, H.-D. (2009). Local susceptibility of the retina, formation and duration of afterimages in the case of Class 1 laser products, and disability glare arising from high-brightness light emitting diodes. *Journal of Laser Applications*, *21*(1), 46–56. https://doi.org/http://dx.doi.org/10.2351/1.3071565

Reuschel, A. (1950). Vehicle movements in a platoon with uniform acceleration or deceleration of the lead vehicle. *Zeitschrift Des Oesterreichischen Ingenieur-Und Architekten-Vereines*, *95*, 50–62.

Rockwell, T. H., Ernst, R. L., & Hanken, A. (1968). A sensitivity analysis of empirically derived car-following models. *Transportation Research/UK/*.

Rothery, P., & Roy, D. B. (2001). Application of generalized additive models to butterfly transect count data. *Journal of Applied Statistics*, *28*(7), 897–909.

Rudin-Brown, C. M., & Parker, H. A. (2004). Behavioural adaptation to adaptive cruise control (ACC): implications for preventive strategies. *Transportation Research Part F: Traffic Psychology and Behaviour*, *7*(2), 59–76.

Saemundsson, T. (1986). Atmospheric refraction. *Sky and Telescope*, *72*, 70.

Saffarian, M., de Winter, J. C. F., & Happee, R. (2012). Automated driving: human-factors issues and design solutions. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 56, pp. 2296–2300). Sage Publications Sage CA: Los Angeles, CA.

Saifuzzaman, M., & Zheng, Z. (2014). Incorporating human-factors in car-following models: a review of recent developments and research needs. *Transportation Research Part C: Emerging Technologies*, *48*, 379–403.

Schultz, G. G., & Talbot, E. S. (2009). Advance warning signals: long-term monitoring results. *Transportation Research Record*, *2122*(1), 27–35.

Seiler, P., Song, B., & Hedrick, J. K. (1998). Development of a collision avoidance system. *SAE Transactions*, 1334–1340.

Shepard, F. D. (1996). *Reduced visibility due to fog on the highway* (Vol. 228). Transportation Research Board.

Shladover, S. E. (2012). Highway capacity increases from automated driving. *California PATH Program*.

Siddharth, S. M. P., & Ramadurai, G. (2013). Calibration of VISSIM for Indian heterogeneous traffic conditions. *Procedia-Social and Behavioral Sciences*, *104*, 380–389.

Souleyrette, R., & Hochstein, J. (2012). *Development of a conflict analysis methodology using SSAM*.

Spencer, J. W. (1971). Fourier series representation of the position of the sun. *Search*, *2*(5), 172. https://doi.org/https://doi.org/10.1109/VR.2006.19

Stringham, J. M., Garcia, P. V, Smith, P. A., McLin, L. N., & Foutch, B. K. (2011). Macular pigment and visual performance in glare: benefits for photostress recovery, disability glare, and visual discomfort. *Investigative Ophthalmology & Visual Science*, *52*(10), 7406–7415.

Summala, H., Nieminen, T., & Punto, M. (1996). Maintaining lane position with peripheral vision during in-vehicle tasks. *Human Factors*, *38*(3), 442–451.

Svensson, Å., & Hydén, C. (2006). Estimating the severity of safety related behaviour. *Accident Analysis & Prevention*, *38*(2), 379–385.

Swerling, P. (1997). Radar probability of detection for some additional fluctuating target cases. *IEEE Transactions on Aerospace and Electronic Systems*, *33*(2), 698–709.

Takamatsu, Y. (2016, November 29). Vehicle control system and method. Google Patents.

Talebpour, A., & Mahmassani, H. S. (2016). Influence of connected and autonomous vehicles on traffic flow stability and throughput. *Transportation Research Part C: Emerging Technologies*, *71*, 143–163.

Theeuwes, J., Alferdinck, J. W. A. M., & Perel, M. (2002). Relation between glare and driving performance. *Human Factors*, *44*(1), 95–107.

Toledo, T. (2007). Driving behaviour: models and challenges. *Transport Reviews*, *27*(1), 65–84.

Tonnis, M., Broy, V., & Klinker, G. (2006). A survey of challenges related to the design of 3d user interfaces for car drivers. In *3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on* (pp. 127–134). IEEE.

Treiber, M., Kesting, A., & Helbing, D. (2007). Influence of reaction times and anticipation on stability of vehicular traffic flow. *Transportation Research Record*, *1999*(1), 23–29.

TROPPMANN, R., & HOEGER, A. (2005). Driver assistance systems-an introduction to Adaptive Cruise Control (ACC). *AUTOMOTIVE ELECTRONICS*, *1*(1).

Ullah, M. B. (1996). International daylighting measurement programme—Singapore data II: luminous efficacy for the tropics. *International Journal of Lighting Research and Technology*, *28*(2), 75–81.

Van Arem, B., Van Driel, C. J. G., & Visser, R. (2006). The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, *7*(4), 429–436.

Van der Horst, A. R. A. (1991). A time-based analysis of road user behaviour in normal and critical encounters.

Van Winsum, W. (1999). The human element in car following models. *Transportation Research Part F: Traffic Psychology and Behaviour*, *2*(4), 207–211.

Vashitz, G., Shinar, D., & Blum, Y. (2008). In-vehicle information systems to improve traffic safety in road tunnels. *Transportation Research Part F: Traffic Psychology and Behaviour*, *11*(1), 61–74.

Vos, J. J. (2003). On the cause of disability glare and its dependence on glare angle, age and ocular pigmentation. *Clinical and Experimental Optometry*, *86*(6), 363–370. https://doi.org/http://dx.doi.org/10.1111/j.1444-0938.2003.tb03080.x

Walch, M., Mühl, K., Kraus, J., Stoll, T., Baumann, M., & Weber, M. (2017). From car-driver-handovers to cooperative interfaces: Visions for driver–vehicle interaction in automated driving. In *Automotive User Interfaces* (pp. 273–294). Springer.

Waldron, K. (2013). Sun glare can be deadly for drivers. *Retrieved from the New*, *20*.

Wang, C.-C., Thorpe, C., & Suppe, A. (2003). Ladar-based detection and tracking of moving objects from a ground vehicle at high speeds. In *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No. 03TH8683)* (pp. 416–421). IEEE.

Wang, J., Kong, Y., Fu, T., & Stipancic, J. (2017). The impact of vehicle moving violations and freeway traffic flow on crash risk: An application of plugin development for microsimulation. *PLoS One*, *12*(9), e0184564.

Wang, L., Abdel-Aty, M., & Lee, J. (2017). Implementation of Active Traffic Management

Strategies for Safety on Congested Expressway Weaving Segments. *Transportation Research Record: Journal of the Transportation Research Board*, (2635), 28–35.

Wang, Xu, Seraj, M., Bie, Y., Qiu, T. Z., & Niu, L. (2016). Implementation of variable speed limits: Preliminary test on Whitemud Drive, Edmonton, Canada. *Journal of Transportation Engineering*, *142*(12), 5016007.

Wang, Xuesong, Chen, M., Zhu, M., & Tremont, P. (2016). Development of a kinematic-based forward collision warning algorithm using an advanced driving simulator. *IEEE Transactions on Intelligent Transportation Systems*, *17*(9), 2583–2591.

Watzenig, D., & Horn, M. (2017). Introduction to automated driving. In *Automated Driving* (pp. 3–16). Springer.

Wiedemann, H., & Stüve, A. (1996). Stüve-Wiedemann syndrome: update and historical footnote. *American Journal of Medical Genetics*, *63*(1), 12–16.

Wiedemann, R. (1994). Simulation des straßenverkehrsflusses. schriftenreihe heft 8. *Institute for Transportation Science, University of Karlsruhe, Germany*.

Wiedemann, Rainer. (1974). Simulation des Strassenverkehrsflusses.

Wienold, J., & Christoffersen, J. (2006). Evaluation methods and development of a new glare prediction model for daylight environments with the use of CCD cameras. *Energy and Buildings*, *38*(7), 743–757.

Wilson, B. H. (2001). *How soon to brake and how hard to brake: Unified analysis of the envelope of opportunity for rear-end collision warnings*. SAE Technical Paper.

Wu, G., Boriboonsomsin, K., Zhang, W.-B., Li, M., & Barth, M. (2010). Energy and emission benefit comparison of stationary and in-vehicle advanced driving alert systems. *Transportation Research Record*, *2189*(1), 98–106.

Xiao, L., Wang, M., Schakel, W. J., Shladover, S. E., & van Arem, B. (2017). *Modeling Lane Change Behavior on a Highway with a High Occupancy Vehicle Lane with Continuous Access and Egress*.

Yan, C., Xu, W., & Liu, J. (2016). Can you trust autonomous vehicles: Contactless attacks against

sensors of self-driving vehicle. *Def Con*, *24*(8), 109.

Young, M. S., & Stanton, N. A. (2007). Back to the future: Brake reaction times for manual and automated vehicles. *Ergonomics*, *50*(1), 46–58.

Zhao, L., & Sun, J. (2013). Simulation framework for vehicle platooning and car-following behaviors under connected-vehicle environment. *Procedia-Social and Behavioral Sciences*, *96*, 914–924.

Zhao, M. (2015). Advanced driver assistant system, threats, requirements, security solutions. *Intel Labs*.

Zheng, B., & Lachapelle, G. (2004). Acquisition schemes for a GPS L5 software receiver. In *Proceedings of the 17th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2004)* (pp. 1035–1040).

Zheng, N.-N., Tang, S., Cheng, H., Li, Q., Lai, G., & Wang, F.-W. (2004). Toward intelligent driver-assistance and safety warning system. *IEEE Intelligent Systems*, *19*(2), 8–11.

Zhizhou, W., Jian, S., & Xiaoguang, Y. (2005). Calibration of VISSIM for shanghai expressway using genetic algorithm. In *Proceedings of the Winter Simulation Conference, 2005.* (pp. 4-pp). IEEE.

Ziraknejad, N., Lawrence, P. D., & Romilly, D. P. (2014). Vehicle occupant head position quantification using an array of capacitive proximity sensors. *IEEE Transactions on Vehicular Technology*, *64*(6), 2274–2287.

Zwahlen, H. T. (1989). Conspicuity of suprathreshold reflective targets in a driver's peripheral visual field at night. *Transportation Research Record*, (1213). Retrieved from https://www.safetylit.org/citations/index.php?fuseaction=citations.viewdetails&citationIds[]=citjournalarticle_604257_38

APPENDICES

8.1 Appendices A: Vision Impairment Detection Python Code

8.1.1 *Reading GIS data*

```
# Converting radians to degrees
def angle_coverter(*args):
    deg = degrees(*args)
    return deg

def get_directory():
    global directory
    directory = ""
    # Find the directory of the file
    source_file = filedialog.askopenfile()
    if source_file:
        directory = source_file.name
    return directory

def get_directory_building():
    global directory_building
    directory_building = ""
    # Find the directory of the file
    source_file = filedialog.askopenfile()
    if source_file:
        directory_building = source_file.name
    return directory_building

def sunboard_gui():
    parent = Tk()
    parent.title("SunBoard - Ver 1.01")
    parent.geometry('300x100')
    directory_button = Button(parent, text="Choose the Road File",
command=get_directory).pack()
    directory_button_2 = Button(parent, text="Choose the Building File",
command=get_directory_building).pack()
    exit_button = Button(parent, text="Exit", command=parent.destroy).pack()
    parent.mainloop()
sunboard_gui()

# Making workbook from 'sample data'
workbook = openpyxl.load_workbook(directory)
road_data_sheet = workbook["Road data"]
```

```
# Making workbook from 'Building Coordination'
workbook_building = openpyxl.load_workbook(directory_building)
building_data_sheet = workbook_building["Buildings"]

# Indicating dimension of the table of data in excel (e.g., how many rows and columns are in the table)
# Extracting data from 'sample_data'
min_row = road_data_sheet.min_row
max_row = road_data_sheet.max_row
min_column = road_data_sheet.min_column
max_column = road_data_sheet.max_column

# 'Extracting data form 'Building Coordinations'
max_building_row = building_data_sheet.max_row
max_building_column = building_data_sheet.max_column

# Step 1 : Extracting data for each row (segment) for "road data"
def segment_info(road_data_sheet, max_row, max_column):
    # Indicating index of 'headers_coords' elements, 'headers_long_lat' elements and 'header_street_id'

    headers = ['VOIE', 'SEG', 'LENGTH', 'XS', 'YS', 'XE', 'YE', 'LONG_S', 'LAT_S', 'LONG_E', 'LAT_E', 'X_EMIN', 'Y_EMIN', 'X_EMAX', 'Y_EMAX', 'EMIN', 'EMAX']  # 'row' address will be added to this list later

    headers_index = []
    segment_info_database = []  # List of segments with all required information (coordination, id)

    # In this section 'r' stands for ROW and 'c' stands for COLUMN
    # Step 2: Finding index of required columns (e.g. X,Y,Z, Elev,..)
    n = 0
    while n < 17:
        for c in range(1, max_column + 1):
            if road_data_sheet.cell(row=1, column=c).value == headers[n]:
                headers_index.append(c)
        n += 1

    for r in range(2, max_row + 1):
        temp = []
        for index in headers_index:
            temp.append(road_data_sheet.cell(row=r, column=index).value)
        temp.append(r)  # Address of row for selected segment
        segment_info_database.append(temp)      #   [[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18], [...], ...]
    return segment_info_database
```

```
segment_info_database = segment_info(road_data_sheet, max_row, max_column)

# Step 3: Extracting data for each row (segment) for "Building Coordination"
def building_info(building_data_sheet, max_building_row, max_building_column):
    headers = ['ELEV', 'Xcoord', 'Ycoord']
    headers_index = []
    building_info_database = []  # List of segments with all required information (coordination, id)

 # In this section 'r' stands for ROW and 'c' stands for COLUMN
 # Finding index of required columns ( e.g., X,Y,Z, Elev,..)
    n = 0
    while n < 3:
        for c in range(1, max_building_column + 1):
            if building_data_sheet.cell(row=1, column=c).value == headers[n]:
                headers_index.append(c)
        n += 1
    for r in range(2, max_building_row + 1):
        temp = []
        for index in headers_index:
            temp.append(round(building_data_sheet.cell(row=r, column=index).value))
        building_info_database.append(temp)      #   [[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18],
[...], ...]
    return building_info_database
building_info_database = building_info(building_data_sheet, max_building_row,
max_building_column)
# building_info_database[0] : [[elev, xcoord, ycoord],...]

# Step 4: Making categories based on the 'VOIE'
def road_sections(segment_info_database):
    road_section = []  # List of roads
    temp_road_section = []

    temp = segment_info_database[:]  # Will be used to check the loop (Stopping the loop)
    while len(temp) != 0:
        temp_road_section = []
        start_point = temp[0][0]  # First random point (VOIE) e.g., 300207

        for i in segment_info_database:
            if i[0] == start_point:
                temp_road_section.append(i)
        road_section.append(temp_road_section)
        for j in temp_road_section:
            temp.remove(j)  # temp list is being updating (Removing points which we found in latest
loop. Then we use updated 'temp' to extract the next 'start_point)
    return road_section
road_category = road_sections(segment_info_database)
```

```
def update_segment_info_database(segment_info_database):
    for element in segment_info_database:
        element_index = segment_info_database.index(element)
        if element[3] == element[11]:
            EMIN = element[15]
            EMAX = element[16]
        else:
            EMIN = element[16]  # 44.6
            EMAX = element[15]  # 44.2
        segment_info_database[element_index].remove(element[15])  # removed : 44.2
        segment_info_database[element_index].remove(element[15])  # removed : 61
        segment_info_database[element_index].insert(5, EMIN)
        segment_info_database[element_index].insert(8, EMAX)
    return segment_info_database
segment_info_database = update_segment_info_database(segment_info_database)

def ascend_descend(element):
    if element[len(element) - 1][6] > element[0][3]:
        return 1
    else:
        return 0

def temp_ends(segment_info_database):  # to do correction of start and end point for each segment
    temp_test = []
    temp_all_segs = []
    temp_north = []
    temp_south = []
    temp_voie = []

    for i in segment_info_database:
        temp_voie.append(i[0])
    temp_voie = list(set(temp_voie))

    for voie in temp_voie:  # temp_voie :
        # In this case there are 11 categories which will be saved separately in 'temp' list.
        temp = [i for i in segment_info_database if i[0] == voie]
        temp_test.append(temp)
        temp_south = []
        temp_north = []
        temp_north_south = []
        n = 0
        for i in temp:
            if n % 2 == 0:
                temp_north.append(temp[n])
            else:
```

```
            temp_south.append(temp[n])
        n += 1
    if ascend_descend(temp_north):    # If    'north' is ascending, therefor, 'south' will be 'descending'.
        for i in temp_north:
            ref = deepcopy(i)
            if ref[3] < ref[6]:
                pass
            else:
                # Replacing XS with XE
                temp_north[temp_north.index(i)][3] = ref[6]
                temp_north[temp_north.index(i)][6] = ref[3]
                # Replacing YS with YE
                temp_north[temp_north.index(i)][4] = ref[7]
                temp_north[temp_north.index(i)][7] = ref[4]
                # Replacing LongS with LongE
                temp_north[temp_north.index(i)][9] = ref[11]
                temp_north[temp_north.index(i)][11] = ref[9]
                # Replacing LatS with LatE
                temp_north[temp_north.index(i)][10] = ref[12]
                temp_north[temp_north.index(i)][12] = ref[10]
                # Replacing EMIN with EMAX
                temp_north[temp_north.index(i)][5] = ref[8]
                temp_north[temp_north.index(i)][8] = ref[5]

        for j in temp_south:
            ref = deepcopy(j)
            if ref[3] > ref[6]:
                pass
            else:
                # Replacing XS with XE
                temp_south[temp_south.index(j)][3] = ref[6]
                temp_south[temp_south.index(j)][6] = ref[3]
                # Replacing YS with YE
                temp_south[temp_south.index(j)][4] = ref[7]
                temp_south[temp_south.index(j)][7] = ref[4]
                # Replacing LongS with LongE
                temp_south[temp_south.index(j)][9] = ref[11]
                temp_south[temp_south.index(j)][11] = ref[9]
                # Replacing LatS with LatE
                temp_south[temp_south.index(j)][10] = ref[12]
                temp_south[temp_south.index(j)][12] = ref[10]
                # Replacing EMIN with EMAX
                temp_south[temp_south.index(j)][5] = ref[8]
                temp_south[temp_south.index(j)][8] = ref[5]
    else:
```

```
            for i in temp_north:
                ref = deepcopy(i)
                if ref[3] > ref[6]:
                    pass
                else:
                    # Replacing XS with XE
                    temp_north[temp_north.index(i)][3] = ref[6]
                    temp_north[temp_north.index(i)][6] = ref[3]
                    # Replacing YS with YE
                    temp_north[temp_north.index(i)][4] = ref[7]
                    temp_north[temp_north.index(i)][7] = ref[4]
                    # Replacing LongS with LongE
                    temp_north[temp_north.index(i)][9] = ref[11]
                    temp_north[temp_north.index(i)][11] = ref[9]
                    # Replacing LatS with LatE
                    temp_north[temp_north.index(i)][10] = ref[12]
                    temp_north[temp_north.index(i)][12] = ref[10]
                    # Replacing EMIN with EMAX
                    temp_north[temp_north.index(i)][5] = ref[8]
                    temp_north[temp_north.index(i)][8] = ref[5]

            for j in temp_south:
                ref = deepcopy(j)
                if ref[3] < ref[6]:
                    pass
                else:
                    # Replacing XS with XE
                    temp_south[temp_south.index(j)][3] = ref[6]
                    temp_south[temp_south.index(j)][6] = ref[3]
                    # Replacing YS with YE
                    temp_south[temp_south.index(j)][4] = ref[7]
                    temp_south[temp_south.index(j)][7] = ref[4]
                    # Replacing LongS with LongE
                    temp_south[temp_south.index(j)][9] = ref[11]
                    temp_south[temp_south.index(j)][11] = ref[9]
                    # Replacing LatS with LatE
                    temp_south[temp_south.index(j)][10] = ref[12]
                    temp_south[temp_south.index(j)][12] = ref[10]
                    # Replacing EMIN with EMAX
                    temp_south[temp_south.index(j)][5] = ref[8]
                    temp_south[temp_south.index(j)][8] = ref[5]
        temp_north_south.append(temp_north)
        temp_north_south.append(temp_south)
        temp_all_segs.append(temp_north_south)
    return temp_all_segs, temp_north, temp_south
all_segments, temp_north, temp_south = temp_ends(segment_info_database)
```

```
# Removing overlapped points
for i in all_segments:
    for j in i:
        for k in j:
            if k[3] == k[6] and k[4] == k[7]:
                del all_segments[all_segments.index(i)][i.index(j)][j.index(k)]
                # or i.remove(j) >>>> updating i, therefor, updating temp_all_segs

# Step 5: Making 'sub-segments'
sub_segments_coords = []
sub_segments_long_lat = []
step = 2.5
sum_length = 0
for i in all_segments:
    for j in i:
        if len(j) != 1:  # IF len(j) <=1,  "j" should be ignored from calculation because in this case there is one segment, therefor it is not possible to distinguish the driver direction.
            for segment in j:
                sum_length += segment[2]
                n = 0
                temp_long_lat = []
                sub_seg_id = 0
                temp_coordination = []
                length = segment[2]
                y_end = segment[7]
                y_start = segment[4]
                x_end = segment[6]
                x_start = segment[3]
                dx = x_end - x_start
                dy = y_end - y_start
                y_lat_end = segment[12]
                y_lat_start = segment[10]
                x_long_end = segment[11]
                x_long_start = segment[9]
                elev_start = segment[5]
                elev_end = segment[8]
                # rs stands for "road slope"
                rs = (elev_end - elev_start) / sqrt(pow(dx, 2) + pow(dy, 2))
                road_slope = atan(rs)
                road_slope = round(angle_coverter(road_slope), 1)

                # Driving PHI
                driver_direction = atan2(dy, dx)
                # Converting the angle from 'Radian' to 'Degree'.
                driver_direction = round(angle_coverter(driver_direction), 1)
```

```
            if -180 < driver_direction < 0:
                driver_direction += 360
            else:
                pass

            # Making sub_segments
            while 5 * n + step < length:
                sub_seg_id += 1  # Making unique id for each 'sub_segment', will be used in 'critical point' analyzing.
                # Number 20 should be updated to 2.5 for real result
                y_mid = int((((n * 5) + step) / length) * (y_end - y_start) + y_start)
                x_mid = int((((n * 5) + step) / length) * (x_end - x_start) + x_start)
                elev_mid = round(((((n * 5) + step) / length) * (elev_end - elev_start) + elev_start), 2)
                lat_mid = round((((n * 5) + step) / length) * (y_lat_end - y_lat_start) + y_lat_start, 5)
                long_mid = round((((n * 5) + step) / length) * (x_long_end - x_long_start) + x_long_start, 5)

                temp_coordination.append([segment[0], segment[1], x_mid, y_mid, elev_mid, road_slope, driver_direction, long_mid, lat_mid, sub_seg_id])
                n += 1
            sub_segments_coords.append(temp_coordination)
```

8.1.2 *Sun positioning algorithm*

```
start_hour = int(input("Start Time 'HOUR' : "))
start_minute = int(input("Start Time 'MINUTE' : "))
finish_hour = int(input("Finish Time 'HOUR' : "))
finish_minute = int(input("Finish Time 'MINUTE' : "))

# Indicating a date in a year
year = int(input("Year : "))
start_day = int(input("Start Day : "))
start_month = int(input("Start Month (input should be in number format) : "))
end_day = int(input("End Day : "))
end_month = int(input("End Month (input should be in number format) : "))

# Making a list >>> Foramt : [hour, mintue, second]
def hour_minute(start_hour, start_minute, finish_hour, finish_minute):
    temp_hour = start_hour
    time_steps = []
    time_steps.append([start_hour, start_minute, 0])
    while (temp_hour * 60) + start_minute < (finish_hour * 60) + finish_minute:
        if start_minute + 5 < 60:
            start_minute += 5
            time_steps.append([temp_hour, start_minute, 0])
        if start_minute + 5 >= 60:
            start_minute = start_minute + 5 - 60
```

```
            temp_hour += 1
            time_steps.append([temp_hour, start_minute, 0])
    return time_steps
time_steps = hour_minute(start_hour, start_minute, finish_hour, finish_minute)

# Make a list of days in a year
def day_in_year(start_day, end_day, start_month, end_month):
    days_in_year = []
    first_day_in_year = date(year, start_month, start_day).timetuple()
    first_day_in_year = first_day_in_year[7]
    last_day_in_year = date(year, end_month, end_day).timetuple()
    last_day_in_year = last_day_in_year[7]
    days_in_year = list(range(first_day_in_year, last_day_in_year + 1))
    return days_in_year

# Check Saving Daylight Time
def dstboundry(year):
    # For March
    march = calendar.monthcalendar(year, 3)
    n = 1
    for i in march:
        if i[6] != 0 and n == 2:  # Second Sunday in March
            march_sunday = i[6]
            break
        else:
            n += 1
    november = calendar.monthcalendar(year, 11)
    november_sunday = november[0][6]
    return march_sunday + 59, november_sunday + 304
march_sunday, november_sunday = dstboundry(year)

# Calculating based on days in year
def time_zone(day):
    if march_sunday <= day <= november_sunday:
        timezone = -4
    else:
        timezone = -5
    return timezone

def sun_position(day, longitude, latitude, hour, minute, second):
    timezone = time_zone(day)
    # or get Hour, Minute and Second automatically from cpu (curent time)
    # now = datetime.now()
    # hour = now.hour
    # minute = now.minute
    # second = now.second
```

```
   fractional_year = round((2 * pi) * ((day - 1 + (hour - 12) / 24) / 365), 2)  # RESULT : "RADIANS"
   equation_of_time = round(229.18 * (0.000075 + 0.001868 * cos(fractional_year) - 0.032077 * sin(fractional_year) - 0.014615 * cos(2 * fractional_year) - 0.040849 * sin(2 * fractional_year)), 2)  # RESULT : MINUTES

   solar_declination_angle = round(0.006918 - 0.399912 * cos(fractional_year) + 0.070257 * sin(fractional_year) - 0.006758 * cos(2 * fractional_year) + 0.000907 * sin(2 * fractional_year) - 0.002697 * cos(3 * fractional_year) + 0.00148 * sin(3 * fractional_year), 2)  # RESULT : "RADIANS"
   time_offset = round(equation_of_time + (4 * longitude) - (60 * timezone), 5)  # RESULT : MINUTES
   true_solar_time = round((hour * 60) + minute + (second / 60) + time_offset, 2)  # RESULT : MINUTES
   if (true_solar_time/4) > 0:
      solar_hour_angle = round((true_solar_time / 4) - 180, 2)  # RESULT : DEGREES
   else:
      solar_hour_angle = round((true_solar_time / 4) + 180, 2)  # RESULT : DEGREES
   latitude_radian = radians(latitude)
   solar_hour_angle_radian = radians(solar_hour_angle)

   # ZENITH ANGLE
   z_angle = sin(latitude_radian) * sin(solar_declination_angle) + cos(latitude_radian) * cos(solar_declination_angle) * cos(solar_hour_angle_radian)
   zenith_angle = round(acos(z_angle), 5)
   elevation_angle = 90 - degrees(zenith_angle)

   # AZIMUTH ANGLE
   az_angle = round((sin(latitude_radian) * cos(zenith_angle) - sin(solar_declination_angle)) / (cos(latitude_radian) * sin(zenith_angle)), 2)
   if az_angle > 1 or az_angle < -1:
      az_angle = 1
   else:
      pass
   if solar_hour_angle > 0:
      azimuth_angle = fmod(degrees(acos(az_angle))+180, 360)
   else:
      azimuth_angle = fmod((540 - degrees(acos(az_angle))), 360)
   return elevation_angle, azimuth_angle
```

8.1.3 *Occlusion test*

```
def occluded_coords(azimuth_angle_compliment_angle, buffer_distance, x_mid, y_mid):
   temp_az = deepcopy(azimuth_angle_compliment_angle) - 360
   occluded_x = x_mid + (cos(radians(temp_az)) * buffer_distance)
   occluded_y = y_mid + (sin(radians(temp_az)) * buffer_distance)
```

```
    return occluded_x, occluded_y, azimuth_angle_compliment_angle

# Defining corners of buffer rectangle
def buffer_corners(x_mid, y_mid, occluded_x, occluded_y, azimuth_angle_compliment_angle):
    temp = []
    buff_corners = []
    modified_sun_phi = deepcopy(azimuth_angle_compliment_angle) - 90
    dx = round(abs(3 * cos(radians(modified_sun_phi))))
    dy = round(abs(3 * sin(radians(modified_sun_phi))))
    temp.append([x_mid + dx, y_mid + dy])
    temp.append([x_mid - dx, y_mid - dy])
    temp.append([round(occluded_x + dx), round(occluded_y + dy)])
    temp.append([round(occluded_x - dx), round(occluded_y - dy)])

    # sorting corners
    temp.sort()
    start_pt = temp.pop(0)
    buff_corners.append(start_pt)
    buff_corners.append(temp[0])
    buff_corners.append(temp[2])
    buff_corners.append(temp[1])
    return buff_corners

city_max_elevation = 314  # max_elevation >>> Maximum Hight in the city
max_occluded = abs(elev_mid - city_max_elevation + 1)  # 1 > hight of driver eyes. !!!!!!!! Need
attention! What is max_occluded?
result = []
sub_seg_index = 1
    for sub_seg in street:  # >>> sub_seg : [300517, 12603751, 298517, 5039572, 51.6, -1.2, 218.3,
-73.58, 45.5, 1]
        latitude = sub_seg[8]
        longitude = sub_seg[7]
        x_mid = sub_seg[2]
        y_mid = sub_seg[3]
        elev_mid = sub_seg[4]
        for time in time_steps:
            hour = time[0]  # 15
            minute = time[1]  # 0
            second = 0
            days_in_year = day_in_year(start_day, end_day, start_month, end_month)
            for day in days_in_year:
                elevation_angle, azimuth_angle = sun_position(day, longitude, latitude, hour, minute,
second)

                # Reading direction from east
                azimuth_angle_compliment_angle = (((90 - azimuth_angle) % 360) + 360) % 360
```

```
driver_direct = sub_seg[6]
```

8.1.4 *Sun glare detection*

```
if abs(driver_direct - azimuth_angle_compliment_angle) <= 30 and elevation_angle > 0 and abs(road_slope - (elevation_angle)) <= 25:
    buffer_distance = round((city_max_elevation - sub_seg[4]) / tan(radians(elevation_angle)), 1)
    occluded_x, occluded_y, azimuth_angle_compliment_angle = occluded_coords(azimuth_angle_compliment_angle, buffer_distance, x_mid, y_mid)
    buff_corners = buffer_corners(x_mid, y_mid, occluded_x, occluded_y, azimuth_angle_compliment_angle)
    # Finding valid points which reside inside of buffer rectangle
    shadow_counter = 0
    shadow_counter_2 = 0
    for point in building_info_database:  # point : [elev, x, y]
      shadow_counter_2 += 1

      #test location of point is inside the buffer area (buff_corners)
      if (point[1] >= buff_corners[0][0] and point[1] <= buff_corners[2][0] and point[2] <= buff_corners[1][1] and point[2] >= buff_corners[3][1]):
        a1 = abs(0.5*(buff_corners[0][1] + buff_corners[1][1])*(buff_corners[0][0] - buff_corners[1][0]) + 0.5*(buff_corners[0][1] + buff_corners[3][1])*(buff_corners[3][0] - buff_corners[0][0])-0.5*(buff_corners[1][1] + buff_corners[3][1])*(buff_corners[3][0] - buff_corners[1][0]))
        a2 = abs(0.5*(buff_corners[1][1] + buff_corners[2][1])*(buff_corners[1][0] - buff_corners[2][0]) + 0.5*(buff_corners[1][1] + buff_corners[3][1])*(buff_corners[3][0] - buff_corners[1][0]) - 0.5*(buff_corners[2][1] + buff_corners[3][1])*(buff_corners[3][0] - buff_corners[2][0]))
        area_buffer = a1 + a2

        triangle_p12 = abs(0.5*(point[2]+buff_corners[0][1])*(point[1]-buff_corners[0][0])+0.5*(point[2]+buff_corners[1][1])*(buff_corners[1][0]-point[1])-0.5*(buff_corners[0][1]+buff_corners[1][1])*(buff_corners[1][0]-buff_corners[0][0]))
        triangle_p23 = abs(0.5 * (point[2] + buff_corners[1][1]) * (point[1] - buff_corners[1][0]) + 0.5 * (point[2] + buff_corners[2][1]) * (buff_corners[2][0] - point[1]) - 0.5 * (buff_corners[1][1] + buff_corners[2][1]) * (buff_corners[2][0] - buff_corners[1][0]))
        triangle_p34 = abs(0.5 * (point[2] + buff_corners[2][1]) * (point[1] - buff_corners[2][0]) + 0.5 * (point[2] + buff_corners[3][1]) * (buff_corners[3][0] - point[1]) - 0.5 * (buff_corners[2][1] + buff_corners[3][1]) * (buff_corners[3][0] - buff_corners[2][0]))
        triangle_p41 = abs(0.5*(point[2]+buff_corners[3][1])*(point[1]-buff_corners[3][0])+0.5*(point[2]+buff_corners[0][1])*(buff_corners[0][0]-point[1])-0.5*(buff_corners[3][1]+buff_corners[0][1])*(buff_corners[0][0]-buff_corners[3][0]))

        if area_buffer == triangle_p12 + triangle_p23 + triangle_p34 + triangle_p41:
          # Calculating distance between observer and selected building
```

```
                    observer_building_distance = sqrt(pow(point[2] - y_mid, 2) + pow(point[1] -
x_mid, 2))
                    if observer_building_distance != 0:

                        # Calculating occluded angle
                        occluded_angle = degrees(atan((point[0] - elev_mid + 1) /
observer_building_distance))
                        # Verifying occlusion
                        if occluded_angle >= elevation_angle:
                            result.append(sub_seg + [round(elevation_angle, 1)] + [hour * 60 +
minute] + [day] + [round(azimuth_angle_compliment_angle)] + ["Shadow"] + [sub_seg_index])
                            shadow_counter += 1
            if shadow_counter == 0:
                result.append(sub_seg + [round(elevation_angle, 1)] + [hour * 60 + minute] + [day]
+ [round(azimuth_angle_compliment_angle)] + ["Sun Glare"] + [sub_seg_index])

    sub_seg_index += 1
```

8.2 Appendices B: Microsimulation Interface

8.2.1 *Speed control – Python code*

```
def toList(NestedTuple):
 #function to convert a nested tuple to a nested list
   return list(map(toList, NestedTuple)) if isinstance(NestedTuple, (list, tuple)) else NestedTuple

def Init():
   #Initialization.
   global vehTypesEquipped
   global vehsAttributes
   global vehsAttNames
   vehsAttributes = []
   vehsAttNames = []
   vehTypesAttributes = Vissim.Net.VehicleTypes.GetMultipleAttributes(['No',
'ReceiveSignalInformation'])
   vehTypesEquipped = [x[0] for x in vehTypesAttributes if x[1]]
   # list of vehicle types which are able to adjust their speed, e.g. [630]
def GetVissimDataVehicles():
 # This function reads vehicle attributes from PTV Vissim
   global vehsAttributes
   global vehsAttNames
   vehsAttributesNames = ['No', 'VehType\\No', 'Lane\\Link\\No', 'Speed', 'DesSpeed',
'OrgDesSpeed', 'DistanceToSigHead', 'Acceleration', 'C2X_HasCurrentMessage']
   vehsAttributes = toList(Vissim.Net.Vehicles.GetMultipleAttributes(vehsAttributesNames))

# create dictionary for the attribute names read from PTV Vissim:
   vehsAttNames = {}
```

```
    cnt = 0
    for att in vehsAttributesNames:
        vehsAttNames.update({att: cnt})
        cnt += 1

def ChangeSpeed():
# read vehicle attributes from PTV Vissim to global variable "vehsAttributes"
    GetVissimDataVehicles()
    # if there are any vehicles in the network
    if len(vehsAttributes) > 1:
        # loop over all vehicles in the network
        for vehAttributes in vehsAttributes:

#optimal speeds based on the analysis of sensitivity analysis [vehicle type, AV penetration rate, volume, optimal speed]
            optspeed = [[620.0, 30.0, 1000.0, 80.0], ….]
            vehtype = vehAttributes[vehsAttNames['VehType\\No']]
            for i in range(1, len(optspeed)):
                volume = Vissim.Net.VehicleInputs.ItemByKey(2).AttValue('Volume(1)')
                if optspeed[i][0] == vehtype and optspeed[i][1] == 50 and optspeed[i][2] == volume:
                    optimalspeed = optspeed[i][3]

 # set easier variables of the current vehicle:
                    Speed = vehAttributes[vehsAttNames['Speed']]
                    DesSpeed = vehAttributes[vehsAttNames['DesSpeed']]
                    OrgDesSpeed = vehAttributes[vehsAttNames['OrgDesSpeed']]
                    DistanceToSigHead = vehAttributes[vehsAttNames['DistanceToSigHead']]
                    Acceleration = vehAttributes[vehsAttNames['Acceleration']]

                    rw = 3.336 * Speed

# if the original desired speed has not yet saved, save it to the UDA "OrgDesSpeed"
                    if OrgDesSpeed is None:
                        OrgDesSpeed = DesSpeed
                        vehAttributes[vehsAttNames['OrgDesSpeed']] = DesSpeed

#Activation of ACWS
                    if DistanceToSigHead < rw:
                            vehAttributes[vehsAttNames['C2X_HasCurrentMessage']] = 1

#activation of IBA
                    if optimalspeed < Speed:
                     optimaldistant = ((Speed ** 2) - (optimalspeed ** 2)) / 64.73

                     if DistanceToSigHead < optimaldistant:
                            vehAttributes[vehsAttNames['DesSpeed']] = optimalspeed
```

8.2.2 *Conflict analysis*

```
def safe_cast(val, to_type, default=None):
    try:
        return to_type(val)
    except (ValueError, TypeError):
        return default

def path_leaf(path):
    head, tail = ntpath.split(path)
    return tail or ntpath.basename(head)

class Vehicle:
    def __init__(self, no, typeNo, second, speed, acceleration, length, coordinationFrontX,
coordinationFrontY, leadTargNo):
        self.no = no
        self.typeNo = typeNo
        self.second = second
        self.speed = speed
        self.acceleration = acceleration
        self.length = length
        self.coordinationFrontX = coordinationFrontX
        self.coordinationFrontY = coordinationFrontY
        self.leadTargNo = leadTargNo

root = tk.Tk()
root.withdraw()

file_path = filedialog.askopenfilename()

input = [line.rstrip('\n') for line in open(file_path, "r")]
i = 0

sequence = {}
start = datetime.datetime.now()
print("Started", start)

while i < 22:
    input.pop(0)
    i += 1

for line in input:
    inputVehicle = line.split(';')

    vehicle = Vehicle(safe_cast(inputVehicle[0], int),
                      safe_cast(inputVehicle[1], float, 0.0),
```

```
                safe_cast(inputVehicle[2], float, 0.0),
                safe_cast(inputVehicle[3], float, 0.0),
                safe_cast(inputVehicle[4], float, 0.0),
                safe_cast(inputVehicle[5], float, 0.0),
                safe_cast(inputVehicle[6], float, 0.0),
                safe_cast(inputVehicle[7], float, 0.0),
                safe_cast(inputVehicle[8], int, 0))
    if not vehicle.second in sequence:
        sequence[vehicle.second] = {}
    sequence[vehicle.second][vehicle.no] = vehicle
result = []

for second in sequence:
    for follower in sequence[second].values():
        if follower.leadTargNo in sequence[second]:
            follower.leader = sequence[second][follower.leadTargNo]
            #print(second, follower.no, "->", follower.leader.no)
            x = follower.coordinationFrontX - follower.leader.coordinationFrontX
            y = follower.coordinationFrontY - follower.leader.coordinationFrontY

            follower.deltaX = round((math.sqrt(x ** 2 + y ** 2)) - follower.length, 2)
            follower.deltaV = round(follower.leader.speed - follower.speed, 2)
            follower.deltaA = round(follower.leader.acceleration - follower.acceleration, 2)

            if follower.deltaV < 0 and follower.deltaA == 0.0:
                ttc = round(-follower.deltaX / follower.deltaV, 2)
            elif follower.deltaV < 0 and follower.deltaA < 0:
                ttc = round((follower.deltaV / follower.deltaA) -
                        ((math.sqrt(follower.deltaV ** 2 - 2 *
                                follower.deltaX * follower.deltaA)) / follower.deltaA), 2)
            elif follower.deltaV >= 0 and follower.deltaA < 0:
                ttc = round((-follower.deltaV / follower.deltaA) -
                        ((math.sqrt(follower.deltaV ** 2 - 2 *
                                follower.deltaX * follower.deltaA)) / follower.deltaA), 2)
            else:
                ttc = 1000
            follower.ttc = round(ttc, 2)
            if follower.typeNo == 100:
                ttct = 2.8
            elif follower.typeNo == 610:
                ttct = 3.0
            elif follower.typeNo == 620:
                ttct = 2.4
            else:
                ttct = 2.1
            follower.ttct = ttct
```

```
            result.append(follower)

outputFullName = path_leaf(file_path)
outputName = outputFullName[0:len(outputFullName)-4]+"_ttc"
outputTxt = filedialog.asksaveasfilename(initialfile=outputName+".txt",title = "Select
file",defaultextension=".txt")
with open(outputTxt, 'w') as output:
    print(['Follower veh ID', 'Vehicle type', 'simulation seconds', 'Speed F', 'Acceleration F',
'Length L',
           'Lead veh ID', 'x follower F', 'y follower F', 'Speed L', 'Acc L', 'x Leader F',
           'y Leader F', 'Delta X', 'Delta V', 'Delta A', 'TTC', 'TTCt'], file=output)
    for follower in result:
        if follower.ttc < 4:
            print([follower.no, follower.typeNo, follower.second, follower.speed,
follower.acceleration, follower.length,
                   follower.leadTargNo, follower.coordinationFrontX, follower.coordinationFrontY,
follower.leader.speed,
                   follower.leader.acceleration, follower.leader.coordinationFrontX,
follower.leader.coordinationFrontY,
                   follower.deltaX, follower.deltaV, follower.deltaA, follower.ttc, follower.ttct],
file=output)

outputCsv = filedialog.asksaveasfilename(initialfile=outputName+".csv",title = "Select
file",defaultextension=".csv")
with open(outputCsv, 'w') as output:
    print('Follower veh ID,Vehicle type,simulation seconds,Speed F,Acceleration F,Length L,'+
          'Lead veh ID,x follower F,y follower F,Speed L,Acc L,x Leader F,'+
          'y Leader F,Delta X,Delta V,Delta A,TTC,TTCt', file=output)
    for follower in result:
        if follower.ttc < follower.ttct:
            print(str(follower.no)+','+ str(follower.typeNo)+','+ str(follower.second)+','+
                  str(follower.speed)+','+ str(follower.acceleration)+','+ str(follower.length)+','+
                  str(follower.leadTargNo)+','+ str(follower.coordinationFrontX)+','+
                  str(follower.coordinationFrontY)+','+ str(follower.leader.speed)+','+
                  str(follower.leader.acceleration)+','+ str(follower.leader.coordinationFrontX)+','+
                  str(follower.leader.coordinationFrontY)+','+
                  str(follower.deltaX)+','+ str(follower.deltaV)+','+ str(follower.deltaA)+','+
                  str(follower.ttc)+','+str(follower.ttct), file=output)

finish = datetime.datetime.now()
print("Finished", finish)
```