Report No. UT-23.16

INTELLIGENT QUEUE LENGTH ESTIMATION ON RAMPS USING COMPUTER VISION AND MACHINE LEARNING ALGORITHMS

Prepared For:

Utah Department of Transportation
Research and Innovation Division

Final Report
October 2023

RESEARCH
UDOT Keeping Utah Moving
Utah Department of Transportation - Research Division
4501 South 2700 West - P.O. Box 148410 - SLC,UT 84114-8410

DISCLAIMER

The authors alone are responsible for the preparation and accuracy of the information, data, analysis, discussions, recommendations, and conclusions presented herein. The contents do not necessarily reflect the views, opinions, endorsements, or policies of the Utah Department of Transportation or the U.S. Department of Transportation. The Utah Department of Transportation makes no representation or warranty of any kind and assumes no liability, therefore.

ACKNOWLEDGMENTS

The authors acknowledge the Utah Department of Transportation (UDOT) for funding this research, and the following individuals from UDOT on the Technical Advisory Committee for helping to guide the research:

- Jamie Mackey (no longer at UDOT)
- Matt Lazanich
- Scott Jones
- Grant Farnsworth
- Dallin Mangels
- K. Colburn
- Scott Stevenson
- Kevin Nichol

TECHNICAL REPORT ABSTRACT

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
UT-23.16	N/A	N/A

4. Title and Subtitle Intelligent Queue Length Estimation on Ramps Using Computer Vision and Machine Learning Algorithms	5. Report Date July 2023
	6. Performing Organization Code
7. Author(s) Sushant Tiwari, Abbas Rashidi, Nikola Markovic'	8. Performing Organization Report No.
9. Performing Organization Name and Address The University of Utah Department of Civil and Environmental Engineering 201 Presidents Circle Salt Lake City, Utah 84112	10. Work Unit No. 5H090 35H
	11. Contract or Grant No. 23-8344
12. Sponsoring Agency Name and Address Utah Department of Transportation 4501 South 2700 West P.O. Box 148410 Salt Lake City, UT 84114-8410	13. Type of Report & Period Covered August 2022 to March 2023
	14. Sponsoring Agency Code PIC No. UT22.302

15. Supplementary Notes

Prepared in cooperation with the Utah Department of Transportation and the U.S. Department of Transportation, Federal Highway Administration

16. Abstract

Traffic sensors are utilized to capture real-time traffic parameters. Many Departments of Transportation in the United States use inductance loops. Induction loops often fail to give accurate results during highly congested periods, which is the most crucial time for ramp metering operations. Image processing and computer vision may provide an alternative with better accuracy. This research employs video captured by existing traffic cameras to measure ramp performance using object detection and tracking in combination with queuing theory.

Deep learning models were constructed and evaluated using data collected from locations on I-15. Four ramps were chosen, considering their height, orientation, and field of view to ensure precise data collection. Additional data was acquired for retraining the base model. Evaluation data helped determine optimal frame size, camera video quality, specifications, and system accuracy, while training data contributed to computer vision framework development.

Computer vision was employed for queue length and queue delay estimation. Under normal conditions, the framework achieved 83% accuracy for queue length and 82% accuracy for queue delay, but these accuracies decreased during sunset/sunrise, night, rain, and snow. Additionally, queuing methods were utilized to enhance the vision-based system's results. A minimal discrepancy of below 1% was observed between the results obtained by the computer vision and queuing methods.

17. Key Words	18. Distribution Statement	23. Registrant's Seal
Ramp performance, queue delay, queue length, Automation, Detection, Tracking, Machine Learning, Deep Learning, Image Processing, Computer Vision	Not restricted. Available through: UDOT Research Division 4501 South 2700 West P.O. Box 148410 Salt Lake City, UT 84114-8410	N/A

19. Security Classification	20. Security Classification	21. No. of Pages	22. Price
(of this report) Unclassified	(of this page) Unclassified		N/A

TABLE OF CONTENTS

EXECUTIVE SUMMARY 10
1. INTRODUCTION 11
1.1 Introduction..11
1.2 Problem Statement..12
1.3 Objectives 13
1.4 Scope...13
1.5 Outline of Report 14
2. LITERATURE REVIEW 15
2.1 Vehicle Detection Techniques 15
2.1.1 In-Roadway Sensors..15
2.1.2 Over-Roadway Sensors 19
2.2 Application of Deep Learning in Transportation..24
2.3 Application of Computer Vision in the Transportation System 26
2.3.1 Object Detection and Recognition 26
2.3.2 Object Tracking 27
3. DATA COLLECTION 29
3.1 Data Collection for Model Evaluation..29
3.2 Data Collection for Model Retraining 32
4. RESEARCH METHOD.. 34
4.1 Data Collection 34
4.2 Data Preprocessing 35
4.3 Object Detection 36
4.4 Object Tracking 38
4.5 Queue Length Detection Using Computer Vision..39
4.6 Queue Delay Calculation Using Computer Vision..40
4.7 Queue Delay Calculation Using Queuing Graphs 41
5. RESULTS AND DISCUSSIONS.. 43
5.1 Object Detection Model Performance Measurements 43
5.2 Retraining Object Detection Model..44

5.3 Computer Vision vs Queue Graph 47
5.4 Image Processing Technique Verification with Ground Truth 47
5.5 Signaling Strategies with Queue Analysis 48
6.0 CONCLUSION 51
6.1 Findings 51
6.2 Limitations 51
6.3 Implementation Plan 52
6.3.1 Camera Functionality 52
6.3.2 Image Processing 54
6.4 Mounting and Orientation Guidelines 55
6.5 Suitability of I-15 Metered Ramps to Computer Vision Deployment 57
7.0 REFERENCES 60

LIST OF TABLES

Table 1: Comparison of Different Sensors for Traffic Data Extraction 24
Table 2: Prevalent Object Detection Algorithms..27
Table 3: Prevalent Tracking Algorithms...28
Table 4: Accuracy of Base YOLOv4 Model 44
Table 5: Accuracy of Retrained YOLOv4 Model...46
Table 6: Queue Delay Comparison Between Algorithms and Queuing Method...........................47
Table 7: Ramp Parameters Variation with Signal Adjustment. 49
Table 8: Calculation of Annual Costs for Cloud Computing 55
Table 9: Qualitative Suitability Assessment of UDOT's I-15 Metered Ramps for CV 57

LIST OF FIGURES

Figure 1: Induction Loop System 16
Figure 2: Inductive Loop Installation 16
Figure 3: Magnetic Sensors Working Mechanisms. 17
Figure 4: Strain Gauge in the Pavement 18
Figure 5: Microwave Radar 20
Figure 6: Infrared Sensors...21
Figure 7: Ultrasonic Waves Sensors 22
Figure 8: Acoustic Sensors 23
Figure 9: 9(a) Camera Locations 10400s 9(b) Sample Frame from UDOT-TOC.........................29
Figure 10: Sample Frame from Location 11400 S - Northbound...30
Figure 11: Sample Frame from Location 500 S - Northbound...30
Figure 12: Sample Frame from Location Beck St - Northbound..31
Figure 13: Different Lighting Conditions and Season Variation in the Evaluation Dataset 31
Figure 14: Lighting Variation in Training Image Dataset 32
Figure 15: Seasonal Variation in Training Image Dataset..33
Figure 16: Flowchart of Methodology Framework 34
Figure 17: Utah Department of Transportation - Traffic Operations Center................................35
Figure 18: (a) Input Frame without Mask (b) Result for Frame without Mask 36
Figure 19: Object Detection and Recognition Using YOLOV4. 37
Figure 20: Object Detection and Tracking..39
Figure 21: Queue Length Detection..40
Figure 22: Queue Delay Calculations 41
Figure 23: Entry and Exit Demarcation 41
Figure 24: Queuing Graph 42
Figure 25: Model Retraining Chart Showing Loss and Average Precision. 45
Figure 26: Comparison of Base YOLO Model and Retrained YOLO Model. 46
Figure 27: Comparison of Ground Truth with Image Processing...48
Figure 28: Ramp Performance Analysis at 15-Minute Intervals 49
Figure 29: Queue Length and Delay Variation in Response to Signal Timing 50

Figure 30: User Interface for Camera System Modeling Software. 56
Figure 31: Installation Drawing for Camera System. 56

UNIT CONVERSION FACTORS

No unit conversions are utilized in this report.

LIST OF ACRONYMS

FPS	Frame Per Second
GPU	Graphics Processing Unit
UDOT	Utah Department of Transportation
YOLO	You Only Look Once
SSD	Single Shot Detector
VPH	Vehicle per hour
UDOT-TOC	Utah Department of Transportation - Traffic Operations Center
FOV	Field of View
WDR	Wide Dynamic Range
LiDAR	Light Detection and Ranging
WIM	Weigh in Motion
RADAR	Radio Detection and Ranging
CNN	Convolution Neural Network
CCTV	Closed-Circuit Television
PTZ	Pan, Tilt and Zoom
CV	Computer Vision
QL	Queue Length
QD	Queue Delay

EXECUTIVE SUMMARY

UDOT's Traffic Management Division sponsored this project to develop an intelligent method of measuring traffic queue length and queue delay using existing infrastructure. This research processes traffic camera video in real time using deep learning and a computer vision framework to measure the ramp performance. Additionally, these metrics will help the traffic management team in decision-making for ramp signal metering. Utah uses ramp metering strategies during peak hours to maintain mainline traffic flow. In this research, four on-ramps were selected in collaboration with the Technical Advisory Committee (TAC) for developing and evaluating the model. The selected locations were on I-15: 10400 S NB, 500 S NB, 11400 S NB, and Beck Street NB.

Video surveillance data was collected with the help of UDOT's Traffic Operations Center (TOC). Four locations that satisfy the conditions of camera orientation, height, video quality, and field of view were chosen. Video surveillance for these chosen locations was recorded for further visual analysis. The field of view was chosen in such a way as to capture the maximum distance of ramp and ramp vehicles. To make the system more robust to external lighting conditions and seasonal variations, video footage under varying light and weather conditions was recorded. A "You Only Look Once" (YOLO) base model was retrained to improve accuracy in all of these conditions to make this technique more dynamic and accurate for real-field applications.

To empower the computer vision system, the research team developed deep learning and computer vision-based software. After collecting the data, videos were preprocessed and passed to the object detection and tracking framework using a combination of state-of-the-art algorithms like YOLOv4 and DeepSORT. YOLOv4 is the state-of-the-art algorithm for object detection and DeepSORT is the state-of-the-art algorithm for object tracking. Python programming was written to combine the framework of these two algorithms to detect the class of an object, count the number of vehicles per lane, and total time spent by each object in the frame. The detailed description and working mechanism of algorithms are explained in this report. The result shows promising accuracy using this method provided that the quality of captured videos is maintained.

1. <u>INTRODUCTION</u>

1.1 Introduction

Ramp metering was introduced in the 1960s on the Eisenhower expressway in Chicago. Subsequently, it was introduced in cities like Detroit, Los Angeles, and Minneapolis/St. Paul. This has led to improved mainline traffic flow, collision reductions, and emission reductions. Since ramp metering has multiple benefits, implementation of ramp metering during peak hours was widely accepted throughout the world and in the United States. Minneapolis has verified that 1160 tons of emissions were prevented from polluting the environment by implementing ramp metering. Moreover, it was efficient in reducing collisions by making the maneuvering process smooth when vehicles from the ramp enter the mainline traffic. Ramp metering also decreases the overall travel time on freeways by preventing shockwaves. (Ramp Metering: A Proven, Cost-Effective Operational Strategy - A Primer: 1. Overview of Ramp Metering, 2014) .

Robust data extraction and analysis are very important to design the ramp metering signal phase. To design the ramp metering signal, data regarding traffic volume and travel time is required. Several sensors like induction loops, radar systems, and acoustic sensors have been employed to extract these data. However, these sensor systems have drawbacks in terms of accuracy, cost, and feasibility. On the other hand, image processing and machine learning can address these problems and have been the core concept of this research effort. Image processing using the video from already existing traffic cameras provides an economical and dynamic data collection framework. Image processing can generate multiple arrays of data like count and speed in a different zone as per the need without demanding new hardware infrastructure, especially for extracting each type of data. Previously, image processing was computationally and operationally expensive. However, recent advancements in artificial intelligence technologies with the advent of machine learning and deep learning algorithms along with the availability of a Graphical Processing Unit (GPU) – a device similar to a Central Processing Unit (CPU) which specializes in graphics and image processing – has made image processing techniques feasible and efficient. The following section describes the problem with existing data collection methods and alternative solutions for addressing those problems.

1.2 Problem Statement

Efficient traffic management strategies at ramps are critical for maintaining optimal flow on highways with ramp metering. Elimination of shockwaves reduces traffic congestion while increasing traffic safety. Ramp metering reduces environmental pollution by lowering fuel consumption and emissions and preventing bottlenecks at the ramp and highway intersections (Wilbur, 2006). To generate insights about the traffic status many sensors are employed, which can be broadly classified as in-roadway sensors and over-roadway sensors. To make the signal coordination efficient and real-time, different sensors such as induction loops are used. However, they are costly and time-consuming during installation and maintenance while also disturbing traffic. Surveillance cameras are another alternative. Many surveillance cameras, such as those installed for traffic incident management, can help reduce the cost of sensors and tools required for signaling any intersection or ramp. Real-time traffic data can be extracted from the video footage recorded in cameras using deep learning and computer vision algorithms (CNN) (Umair et al., 2021).

Vehicle queue length and queue delay are important parameters for designing a ramp signal metering algorithm. Proper design of the ramp metering signal is necessary to maintain the flow of mainline traffic and prevent ramp spillover. Queue length and queue delay can be estimated using image processing with the help of object detection and tracking algorithms. Object detection helps to recognize the vehicle class and spatial location of an object in the frame. Object detection followed by tracking algorithms helps to keep track of the delay faced by an individual vehicle located within the frames. The combination of detection and tracking algorithms can generate information such as average delay, average queue length, and vehicle class. Image processing provides an additional advantage of less interruption to the service of roads and mainlines while installing and maintaining the system. Recognizing the potential of new technology, the Utah Department of Transportation (UDOT) Traffic Management Division is exploring using an existing camera system to facilitate ramp metering that will provide an economical solution with promising accuracy.

1.3 Objectives

The primary objective of this research is to help UDOT identify the potential of using video footage from existing cameras to accurately determine ramp queues and ramp delays.
The secondary objective of this research is to provide general design guidance concerning the mounting height and orientation of cameras, as well as camera technical specifications, for optimal performance.

1.4 Scope

Task 1: Literature review

Task 1 focuses on conducting a literature review on machine learning and computer vision techniques on object detection and tracking. Along with these, additional literature was reviewed on queue length estimation using statistical tools and techniques like queue estimation techniques.

Task 2: Identifying the potentially feasible location and data collection

Task 2 focuses on identifying potential ramp locations with existing cameras based on the height, orientation, and technical specifications of existing cameras. The research team along with the UDOT traffic operations team identified at least four different locations for this research. Recorded video camera data is analyzed using the object detection and tracking framework developed in this research.

Task 3: Methodology framework and algorithm development

This task will develop a framework combining object detection and tracking algorithms to gather traffic information. Additionally, this step will explain in brief the preprocessing and postprocessing of data. Moreover, it will provide details on the queuing methods.

Task 4: System evaluations

This task will assess the system's accuracy and effectiveness in measuring delay and queue length. Manual ground truth will be extracted from small samples of video data that are randomly selected. The outcomes of video processing techniques will be compared to manual ground truth data.

Task 5: Conclusion and key findings

This task will provide a recommendation for the best frame size, ramp performance measuring tools, object detection, and tracking algorithms after evaluating these techniques' ability to accurately determine the queue length and queue delay. In addition, the research team will describe camera specifications and computing devices required to deploy this system.

1.5 Outline of Report

- Introduction
- Literature Review
- Data Collection
- Research Methods
- Evaluation and Analysis
- Conclusions

2. <u>LITERATURE REVIEW</u>

This section describes the advantages and disadvantages of existing sensor technologies. Additionally, it will explain how modern approaches, such as deep learning, have been used in the transportation domain. Finally, it will outline how the ramp performance parameters are extracted employing computer vision and how data is analyzed using vision as well as queuing methods for ramp metering strategies.

2.1 Vehicle Detection Techniques

Vehicle detection techniques are classified into two categories based on the location of the sensors. They can be broadly categorized as in-roadway sensors and over-roadway sensors.

2.1.1 In-Roadway Sensors

The sensors that are embedded in the road pavement to extract data about traffic are called in-roadway sensors. The working mechanism of these sensors is explained in detail in subsequent sections.

2.1.1.a. <u>Induction Loops</u>

The induction loop system is a traffic sensor used to measure the presence, vehicle passage, count, and occupancy of vehicles in a roadway. It is made of a lead-in cable wire loop with a signal range of 10KHz to 50KHz and sends signals to the electronics unit of the controller cabinet via a pull box. Excitation of the loop is due to the increase in oscillation frequency caused by the decrease in inductance when the vehicle stops or passes by the loop system. Speed can be deduced using two induction loop systems or using the single loop system and vehicle length. This method is well understood in practice and suitable for measuring the basic traffic parameters such as headway, volume, presence of vehicles, and the gap between them. Details about the induction loop installation system are shown in Figure 1 and Figure 2. Moreover, its flexible design makes it more suitable for various applications. However, its installation and repair frequently lead to disruption of traffic flow. They need to be reinstalled during each road repair and utility repair, which increases the life cycle cost of the loop. Further, stresses from the vehicle and temperature also can lead to loop failure (Wilbur, 2006).

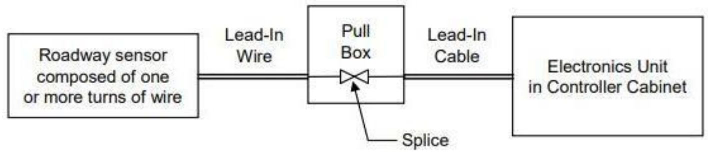


Figure 1: Induction Loop System

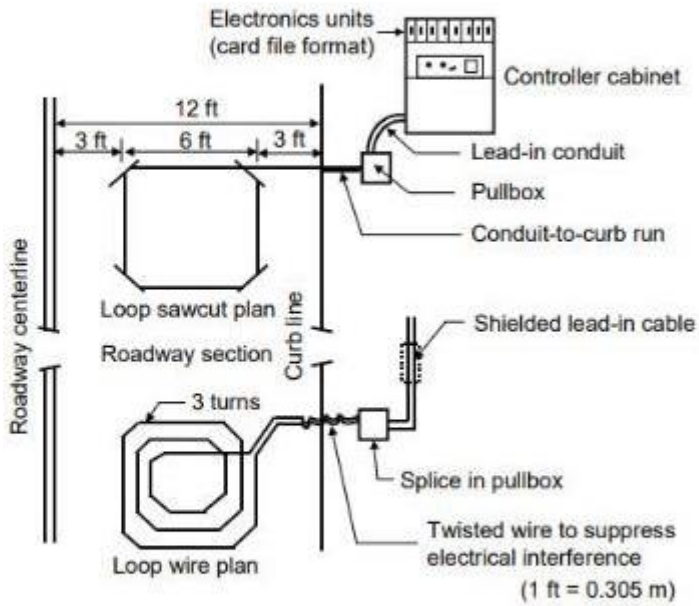


Figure 2: Inductive Loop Installation

Source: Traffic Detector Handbook, Third Edition - Vol I

2.1.1.b. Magnetic Sensors:

Magnetometers can measure the disturbance in the earth's magnetic field caused by metallic objects, which is the basis for this system's working mechanism. Magnetic anomalies can be used to detect the presence of vehicles in a magnetometer's detection zone (Wilbur, 2006). Magnetometers detect the variation in the magnetic field, providing vehicle count, speed, length, lane occupancy, daily and annual average traffic, road surface temperature, and wet-dry conditions (Elena Mimbela Project Manager et al., 2007). Magnetic detectors are less susceptible to pavement stress than induction loops but require pavement cuts and traffic disruptions. The working mechanisms of magnetic sensors are shown below in Figure 3.

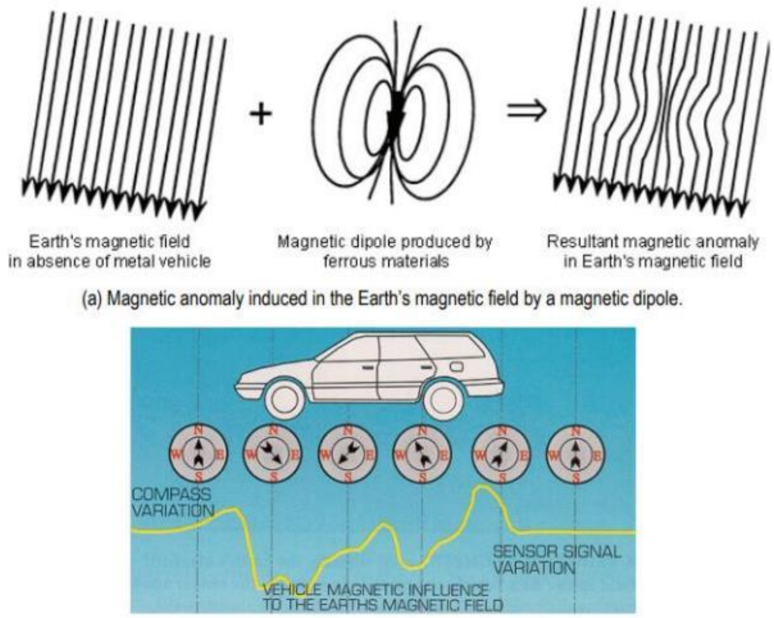


Figure 3: Magnetic Sensors Working Mechanisms

Source: Traffic detector handbook, Third Edition - Vol I

2.1.1.c. <u>Induction Coil:</u>

The working mechanism of this is similar to an induction loop system. This mostly replaced the induction loop system in locations where pavement cuts are not practical (e.g., bridge decks, etc.). But this still demands boring under the pavement. It is resistant to inclement environmental conditions such as rain, snow, and fog. Furthermore, it is less prone to damage under traffic stresses. However, it cannot detect the stopped vehicle unless special sensors and software are used.

2.1.1.d. <u>Piezoelectric Sensors:</u>

Piezoelectric materials are sensors that convert kinetic energy (vibrations or mechanical energy) to electrical energy. For traffic counting and determining axle spacing, it is inserted in each lane and covered with flush epoxy resin. When the same vehicle activates two piezoelectric sensors, the vehicle's speed and axle spacing are computed. The measured voltage is proportional to the weight of the vehicle. These are often deployed in weigh-in-motion (WIM) systems. A junction box located on the roadside collects these signals to provide information about numbers and types of vehicles in digital code format. The small shape and size of these sensors make them easier to handle. Nonetheless, it is sensitive to temperature and sensitive to water infiltration. It can be used for measuring dynamic motions only (ECSTUFF4U for Electronics Engineer: Advantages and Disadvantages of Piezoelectric Transducer).

2.1.1.e. Strain Gauge:

Strain gauges classify the vehicle based on the dynamics strain response pattern generated from a gauge embedded in the pavement that is precise in measurement, easy to maintain, and has a longer operating life. However, strain gauges can be affected by temperature differences which can be easily corrected using a thermistor. It is difficult to place and keep firmly on the pavement throughout its life cycle ((W. Zhang et al., 2008), (Advantages and Disadvantages of Piezoresistive Strain Gauge Sensors - TM Automation Instruments Co.), (Strain Gauge: Principle, Types, Features, and Applications | Encardio Rite))

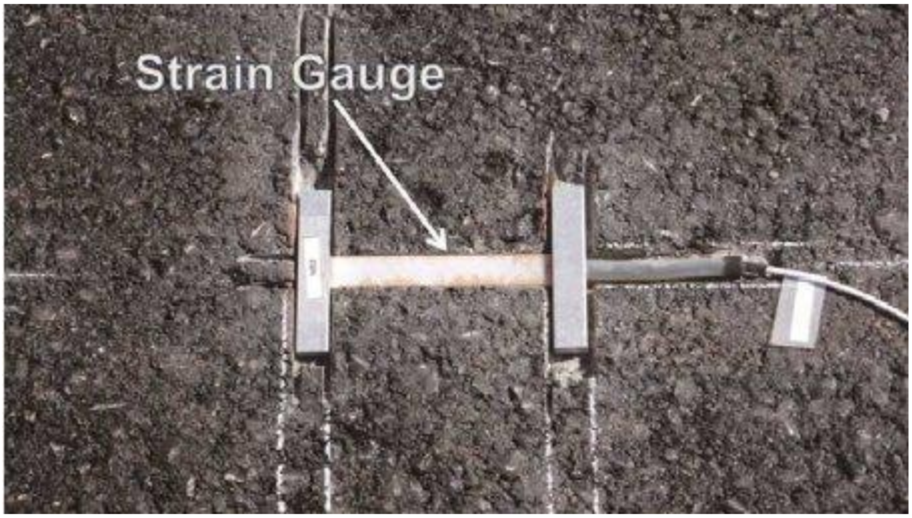


Figure 4: Strain Gauge in the Pavement

Source: (Partl et al., 2015)

2.1.1.f. Seismic Sensors:

The vibrations produced by moving cars can be detected using seismic sensors. A number of these sensors working together can gather information to localize and categorize different kinds of vehicles. Zhou uses the short-time power spectrum density method to extricate features from seismic signals generated by a vehicle. Although seismic sensors have a relatively wide range, they require rigorous calibration before use (Q. Zhou et al., 2013).

2.1.1.g. <u>Pneumatic Tubes:</u>

These are the tubes that send pressurized air through the pipe when a vehicle traverses over the tube. The pressurized air closes an air switch, producing an electrical signal that is later processed by the software. It is commonly used for short-term traffic counting and vehicle classification by axle count and spacing. These sensors are quick to install, consume very low energy to operate, and are easier to maintain. Nonetheless, when the volume of traffic increases, axle counting can be inaccurate. It is sensitive to temperature and prone to damage by wear and tear caused by vehicles (Elena Mimbela Project Manager et al., 2007).

2.1.2 Over-Roadway Sensors

Sensors located above the road pavement or at a certain distance from the pavement are classified as over-roadway sensors. Multiple types of over-roadway sensors are explained in detail below.

2.1.2.a. <u>Microwave Radar:</u>

Microwave Radio Detection and Ranging (Microwave RADAR) transmits electromagnetic energy of almost 10GHz for traffic management. This radar can be continuous wave (CW) Doppler radar or frequency-modulated continuous wave (FMCW). Microwave radar sensors are placed at the overhead antenna that will emit the energy. When a vehicle goes through this energy beam it will reflect the energy beam to the sensor with a different energy level.

Figure 5: Microwave Radar

Source: (Elena Mimbela Project Manager et al., 2007)

This reflected energy can compute the volume, speed, occupancy, and length. Side-mounted FMCW can provide multilane coverage, left turns, and traffic queues. It is resistant to inclement weather conditions while providing direct speed data and multilane traffic flow data. Nevertheless, CW Doppler radar cannot detect stationary vehicles and performs poorly at intersections (*FIELD TEST OF MONITORING OF URBAN VEHICLE OPERATIONS USING NON-INTRUSIVE TECHNOLOGIES Final Report S O N I C*, 1997).

2.1.2.b. Infrared:

Active and passive infrared can be used to monitor approaching and departing vehicles using the top-mounted and side-mounted configurations. It can be used for signal control, speed detection, volume measurements of vehicles and pedestrians, and classification. It operates on the principle that emitted energy is converted to electrical signals which can be taken further into processing and deducing the results. It can be active radar and passive radar.

An active radar transmits energy of its own and receives the reflected energy. It is done using two sets of optics: transmitting optics and receiving optics. Radars can measure distance, speed, queuing, volume, and even the classification of vehicles based on this energy. It can detect up to 11 types of vehicles on toll roads (Elena Mimbela Project Manager et al., 2007). It can be mounted at the top of the road surface.

Passive radar does not emit energy of its own but receives the energy transmitted by the object i.e., road and vehicle. It can detect volume, lane occupancy, and passage. Passive radar receives the graybody emission emitted by the objects. When a vehicle enters the road, the signal is generated which is due to the emissivity difference between the road and vehicles. It is resistant to inclement weather if calibrated carefully.

The major advantage of using infrared sensors is that they don't require pavement disruption which reduces installation and maintenance costs. This increases the operational efficiency of the road. These sensors emit multiple beams for accurate measurements of traffic parameters. Side-mounted models can even facilitate the multi-lane detection system. However, glint from sunlight can cause errors and inclement weather can disperse energy which can affect the energy being received (*FIELD TEST OF MONITORING OF URBAN VEHICLE OPERATIONS USING NON-INTRUSIVE TECHNOLOGIES Final Report S O N I C*, 1997).

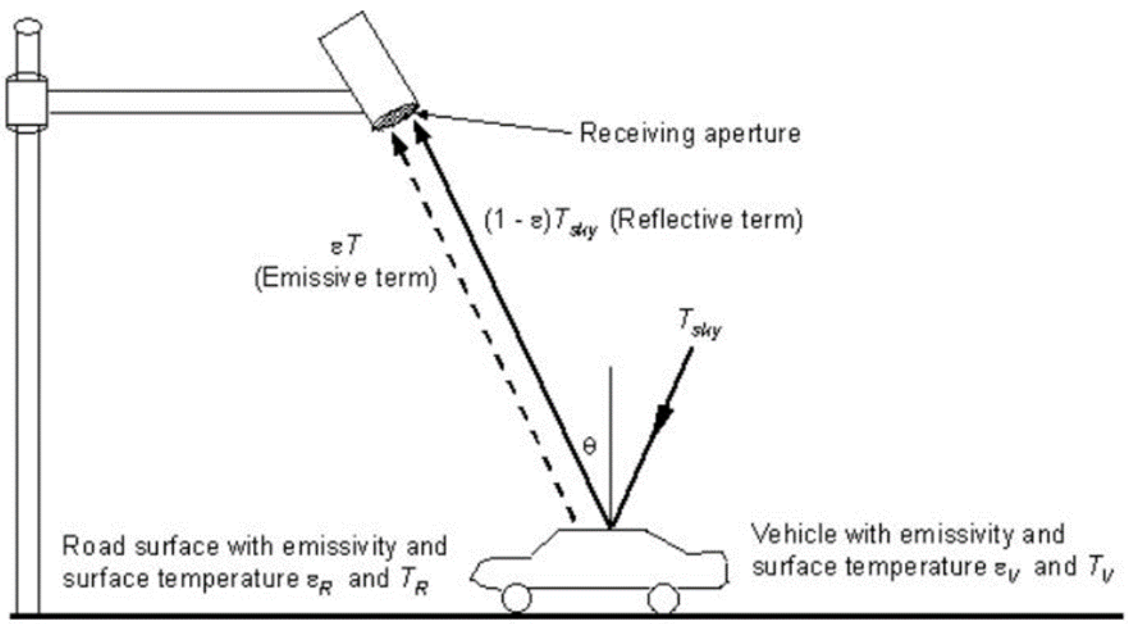


Figure 6: Infrared Sensors

Source: (Elena Mimbela Project Manager et al., 2007)

2.1.2.c. Ultrasonic Waves:

Ultrasonic waves emit sound waves at 25-50 kHz, which is above the audible human range. These waves mostly use pulse waveforms to provide information about vehicle count, presence, and occupancy information. These are mostly configured as overhead mounts and horizontal mounts.

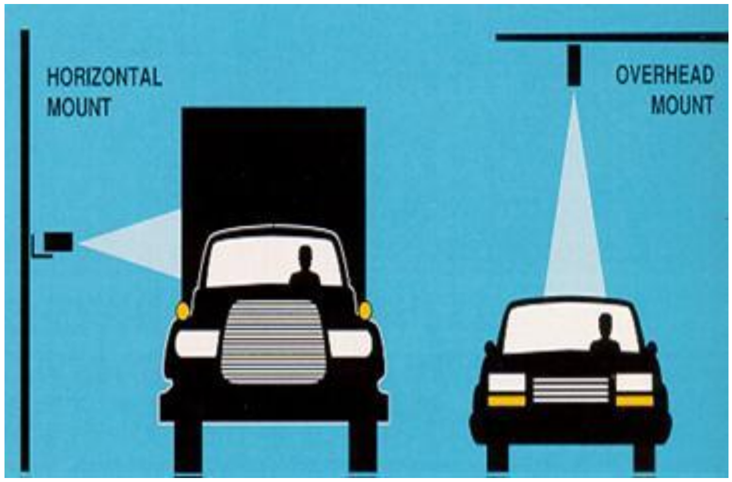


Figure 7: Ultrasonic Waves Sensors

Source: (Elena Mimbela Project Manager et al., 2007)

The waves emitted back can help in differentiation whether it was reflected by road surface or by vehicles. The benefit of using ultrasonic waves is that they don't disrupt the pavement which will ultimately reduce the maintenance and installation cost of the system. Additionally, more advanced ultrasonic waves can be operated on a multilane level. Even so, this technology is susceptible to temperature change and extreme turbulence in the air.

2.1.2.d. Acoustic Waves:

This system detects vehicular movement by capturing the sound produced by a vehicle's tires interacting with the pavement as well as sound from various other sources. It works on the doppler effect, with the approaching vehicle increasing sound energy and the departing vehicle decreasing sound energy. Acoustic waves are extremely beneficial because they are a non-intrusive method of vehicle detection with multilane detection. They are, however, sensitive to cold weather and cars moving slowly in stop-and-go traffic.

Figure 8: Acoustic Sensors

Source: (Elena Mimbela Project Manager et al., 2007)

2.1.2.e. LiDAR:

LiDAR stands for light detection and ranging. It uses ultrasonic, visible, and infrared rays to find the distance of an object. It sends light waves known as emits and absorbs light waves known as pulses. Emits and pulses are used to calculate the distance of an object. It can collect data from the ground, from an airborne vehicle medium, and even from satellites. Yuepeng uses LiDAR technology to track vehicles while developing a connected-vehicle system (Cui et al., 2019). LiDAR was used to detect and track pedestrians (Peng & Shan, 2021) and to estimate the queue length (Wu et al., 2020). LiDAR is useful to collect data both day and night, and is not affected by extreme weather conditions. However, it has a higher operating cost, and is ineffective during heavy rains and cloudy days (Advantages and Disadvantages of LiDAR – LiDAR and RADAR Information).

2.1.2.f. Image Processing Technique:

Image processing algorithms are employed to process the video captured by a surveillance camera. These algorithms process video using frame-by-frame analysis, and try to find the moving object based on background subtraction techniques. Algorithms find the vehicle by searching the unique connected area of pixels. It has been realized that using a video feed from a surveillance camera for traffic flow analysis is more appropriate than using other sensors that require multiple numbers. It can be used to extract additional traffic data from the same video-capturing devices, and the type of data extraction can be easily modified using algorithms without upgrading the system's device requirements.

Table 1: Comparison of Different Sensors for Traffic Data Extraction

sensors		count	speed	occupancy	classification	multiple lane mulitple detection	acceleration	direction	wt	axle configuration	type and model	automatic
In-road sensors	Inductive loop	✓	✓	✓	✓	✗	✓	✓	✗	✗	✗	✓
	Magnetometer	✓	✓	✓	✗	✗	✓	✓	✗	✗	✗	✓
	Induction coil	✓	✓	✓	✗	✗	✓	✓	✗	✗	✗	✓
	Piezoelectric	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
	strain gauge	✓	✓	✗	✓	✗	✓	✓	✗	✓	✗	✓
	seismic	✓	✗	✗	✓	✗	✗	✗	✗	✗	✗	✓
	Pneumatic tube	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
Over-Roadways sensors	microwave radar	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
	active infrared	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
	passive infrared	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	✓
	ultrasonic waves	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✓
	acoustic waves	✓	✓	✓	✗	✓	✓	✓	✗	✗	✗	✓
	LiDAR	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓
	Image processing	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓

To elaborate, the same video can be used to extract information about pedestrians and vehicles, traffic incident management, and to make multi-lane observations without the need of changing the sensor's systems and hardware. It can provide a wide range of data when multiple camera video feeds are linked together. However, these sensors are more prone to lighting conditions, weather conditions, occlusions, and camera motions (Elena Mimbela Project Manager et al., 2007). Recent advancements in deep learning methods and computer vision approaches may be able to overcome these challenges to an acceptable level.

2.2 Application of Deep Learning in Transportation

Deep learning has helped in decision-making in different aspects of human life. Deep learning algorithms usually have many layers of representation created by constructing non-linear modules that improve and abstract the representation at one level, aiding in the learning of complex functions (Lecun et al., 2015).

Deep learning algorithms have major roles in the transportation engineering domain to make the system safe, economical, and efficient. It can be further employed to collect and manage assets and inventories. Mohammadi et al., (2023) used various machine learning as well as deep learning for automating culvert management. Deep learning was applied to represent the transportation network for modeling the spatial and temporal dependencies which helped in proactive congestion mitigation with earlier identification of vulnerable links. To learn the traffic parameters and forecast the traffic situation at time T+1, the model used past data from T intervals (Ma et al., 2015). Further, deep learning is used in traffic flow predictions: predicting the number of vehicles in road segments in different time intervals in the future (Lippi et al., 2013). It is also employed in vehicle classification based on the vehicle's GPS trajectory data (Dabiri et al., 2020). Additionally, deep learning has been used for controlling traffic signals with higher efficiency than traditional methods and reducing emissions and congestion simultaneously (Nguyen et al., 2018). Genders and Razavi (Genders & Razavi, 2016) used a deep and cross-network (DCN) deep learning framework that uses deep reinforcement learning (DRL) techniques to build an adaptive traffic signal control system. Deep learning has reduced the cumulative delay, queue length, and travel time in traffic control systems by 82%, 66%, and 20%, respectively. Deep learning models are used for travel demand modeling for cars (Zhu et al., 2017), mass rapid transit (L. Liu & Chen, 2017), and public transportation (Baek and & Sohn, 2016). Deep learning is used for predicting the passenger demand and respective supply-side infrastructure demand for efficient traffic management using GPS data, traffic sensors, and CCTV. Cheng et al., (2017) used the Deep neural network (DNN), LSTM (long short-term memory), and feature-level data fusion model to forecast day-to-day travel demand in the transportation network of Florida. Deep learning is used to monitor complex driver behavior based on GPS data which can be used for insurance companies and autonomous driving which also enhances road safety (Dong et al., 2016). Further, deep learning is helpful in traffic incident management systems. Q. Chen et al., (2017) implemented a deep stack denoise autoencoder to model human mobility and generate a traffic incident risk map.

Additionally, C. Chen et al., (2018) deployed the deep genetic algorithm-optimized network to develop an automatic avoidance system for rear-end collisions. Deep learning assists in making safer, more economic, and more efficient traffic management strategic decisions. Additionally, another extension of deep learning that uses a convolution neural network (CNN) helps in image processing to further support the optimal decision-making process.

2.3 Application of Computer Vision in the Transportation System

Image processing techniques with combinatory use of deep learning technology have further enhanced the decision-making process in the transportation system. It has helped in developing the driver assistance system, autonomous driving system, traffic inventory management, traffic signal design, etc. Farhadmanesh et al., (2021) used mobile photogrammetry to assess highway asset and pavement monitoring, and Rashidi et al., (2018) utilized videos and point clouds to automatically map the built infrastructure into bridge information models. Likewise, 3-dimensional reconstruction of transportation infrastructure has been done using image and video processing (Brilakis et al., 2011; Dai et al., 2013) and depth mapping using a stereo vision system (Rashidi et al., 2011). Similarly, Hassandokht Mashhadi et al., (2023) used computer vision to provide the safety rating for rural highways. For determining queue lengths, image processing must apply a two-step approach for the complete data extraction needed. The first step is object detection and recognition while the second step is object tracking.

2.3.1 Object Detection and Recognition

Object detection detects the spatial location of an object of interest with the frame while recognition allows us to determine the class of an object after detecting the object. Alzubaidi et al., (2021) used the CNN method to detect the vehicles from UAV images. Y. Zhou et al., (2016) used DNN to detect cars, sedans, and vans. Farhadmanesh et al., (2022) identified light aircraft in non-towered airports using image processing tools. Huval et al., (2015) used CNN for image processing on highway driving and were able to detect the lane and vehicles at a speed acceptable to the real-time systems. Jehad & Rahmat, (2017) were able to detect vehicles with up to 95% accuracy by employing fast vehicle detection methods using background subtraction, shadow removal, and pixel analysis. Yu et al., (2011) were able to capture small vehicles (97.7% accuracy) and large vehicles (97.1% accuracy) using a length-based vehicle classification method that employs background subtraction and threshold-based segmentation. This detected object needs to be recognized into different classes and tracked using the bounding box and tracking ID to find the total delay each vehicle spent in the queue. Table 2 presents the prevalent object detection algorithms. Among them, “You Only Look Once” (YOLO) is found to be the most efficient in terms of both accuracy and speed (Srivastava et al., 2021).

Table 2: Prevalent Object Detection Algorithms

S.N	Object Detection Algorithms	Reference
1	R-CNN	(Girshick et al., 2013)
2	FAST R-CNN	(Girshick, 2015)
3	FASTER R-CNN	(Ren et al., 2015)
4	Cascade R-CNN	(Cai & Vasconcelos, 2017)
5	Single-Shot Multibox Detector (SSD)	(W. Liu et al., 2015)
6	Single-Shot Refinement Neural Network for Object Detection (RefineDet)	(S. Zhang et al., 2017)
7	Retina-Net	(Lin et al., 2017)
8	You Only Look Once (YOLO)	(Bochkovskiy et al., 2020)

2.3.2 Object Tracking

Object tracking is localizing the spatial location of an object in videos frame by frame. In this method, the tracking ID is allocated to objects, and the object is tracked in future frames until the object is still in the frame. Pellegrini et al., (2009) used object tracking to study people's social behavior. Koller et al., (1994) used the active contour model to track multiple vehicles in road traffic. Betke et al., (2000) used a combination of features like color, edge, and motion to track road boundaries, lane marking, and other vehicles. Object tracking can be single object tracking (SOT) and multiple object tracking (MOT). MOT is more complicated than SOT due to its similar appearance, ID-switching problems, and occlusions. Object tracking follows the detection, estimation model, data association, creation, and deletion of track identities. Detection is based on CNN methods to extract features and localize the portion of an image that will be fed for the next step to classify the object. The estimation model tries to predict the target identity in the next frame using the linear constant velocity model. Most estimation models use the Kalman filter to predict the future state of an object in the next frame. The data association process tries to find the correlation between the predicted target from the estimation model and new detection using Intersection over Union (IOU) matching. When there is enough correlation greater than a threshold, then tracking ID is associated with new detection. Creation and deletion of the track identities perform the assigning of the tracking ID when the object first appears in the frame of analysis and deleting the tracking ID when an object goes out of the frame of interest. Table 3 summarizes the most prevalent tracking algorithms. Tarushree (Kumar, 2020) and Mandal (2007) verifies Deepsort is a better tracking algorithm compared to other algorithms. Mandal (2007) asserts that the combination of yolov4 and Deepsort is an optimized combination for multiple object detection algorithms.

Table 3: Prevalent Tracking Algorithms

S. N	Tracking Algorithms	References
1	Simple Online and Real-Time Tracking (SORT)	(Bewley et al., 2016)
2	FairMOT	(Y. Zhang et al., 2020)
3	TransMOT	(Chu et al., 2021)
4	ByteTrack	(Y. Zhang et al., 2022)
5	DeepSORT	(Wojke et al., 2017)

3. DATA COLLECTION

3.1 Data Collection for Model Evaluation

Data collection was conducted with the help of professional staff from the Utah Department of Transportation's Traffic Operations Center. The research team and TAC members have selected four suitable locations for data collection. Owing to the visual system, orientation, visibility, and height of the camera were important factors to consider for obtaining optimal video footage of ramp operations. Four suitable locations which were selected are shown below in Figure 9. To make the model generalization power more robust, data was collected in varying lighting conditions -- morning time, daylight time, evening time, sunset time, and nighttime. Additionally, the computer vision model was strengthened by testing seasonal variations. Video footage during the snow, rain, and rain plus night was taken into consideration to evaluate this model. Four major locations for traffic video collection are:

a. **10400 S NB**

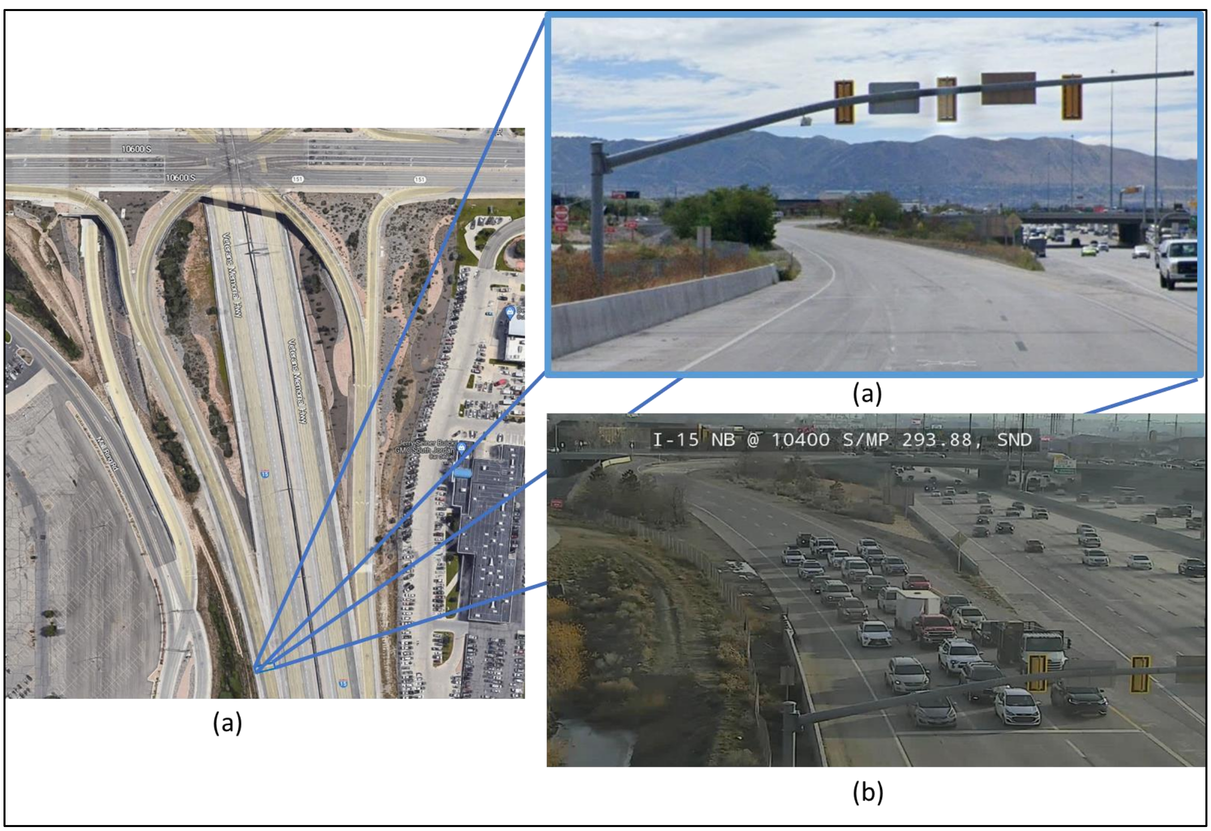


Figure 9: 9(a) Camera Locations 10400 S 9(b) Sample Frame from UDOT-TOC

b. **<u>11400 S NB</u>**

Figure 10: Sample Frame from Location 11400 S - Northbound

c. **<u>500 S NB</u>**

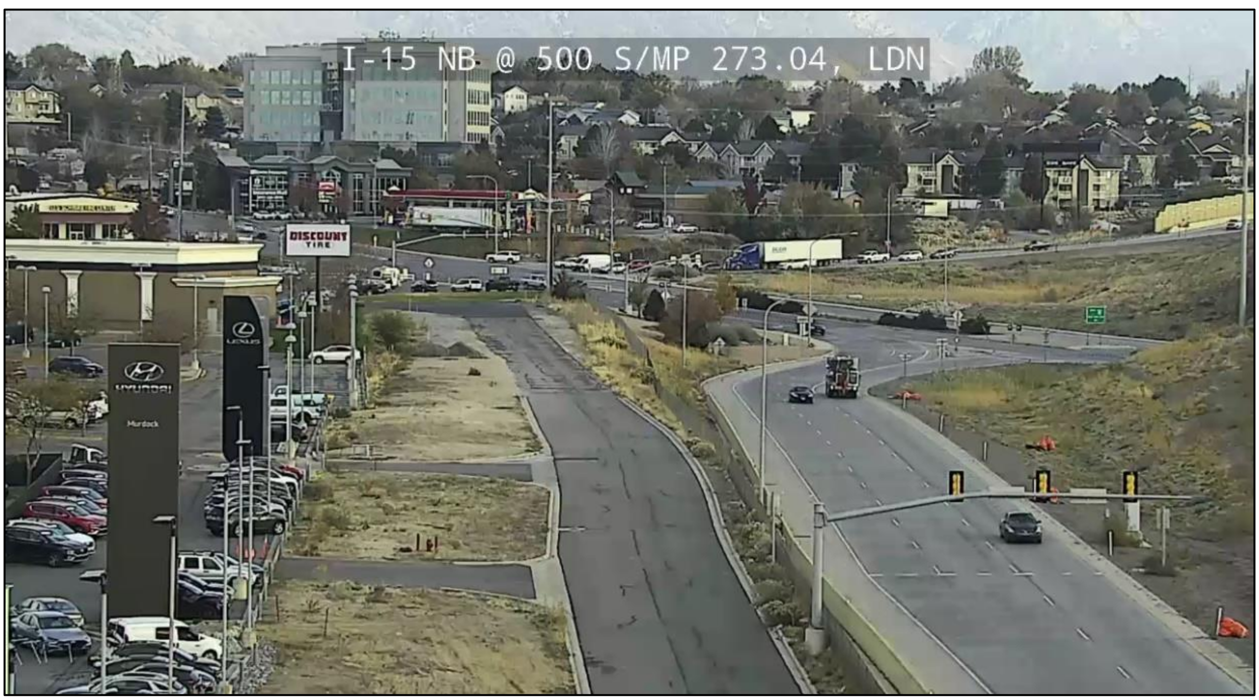


Figure 11: Sample Frame from Location 500 S - Northbound

d. **<u>Beck St NB</u>**

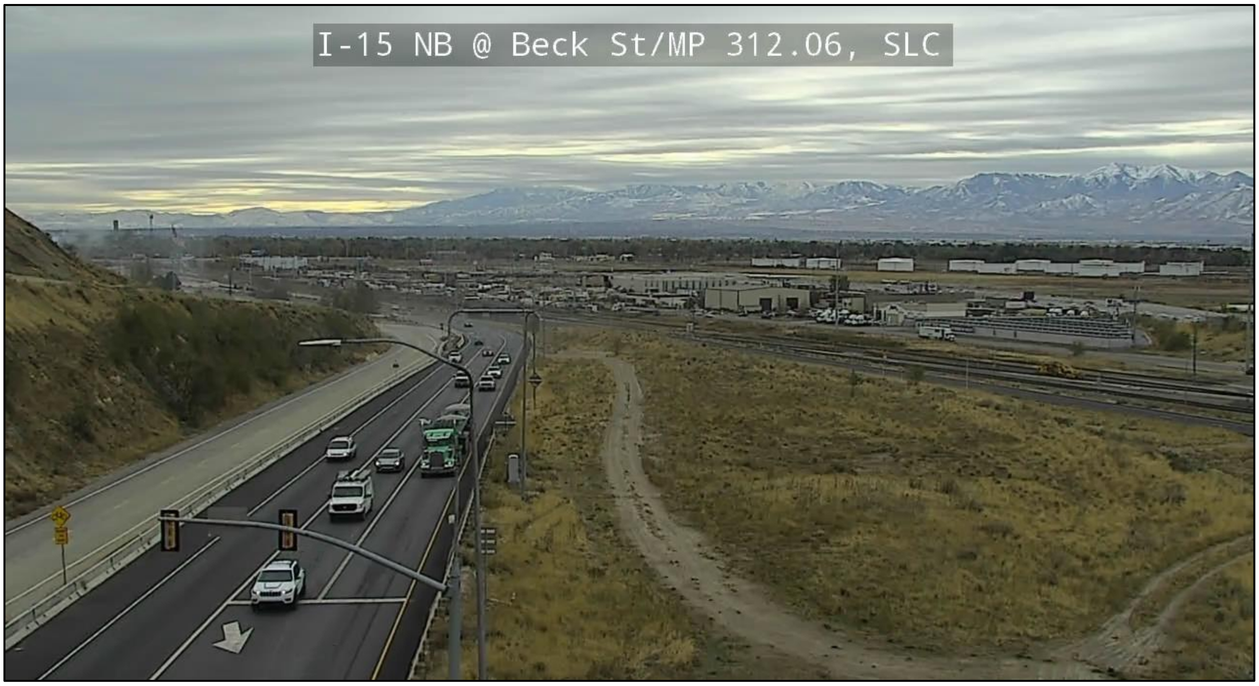


Figure 12: Sample Frame from Location Beck St - Northbound

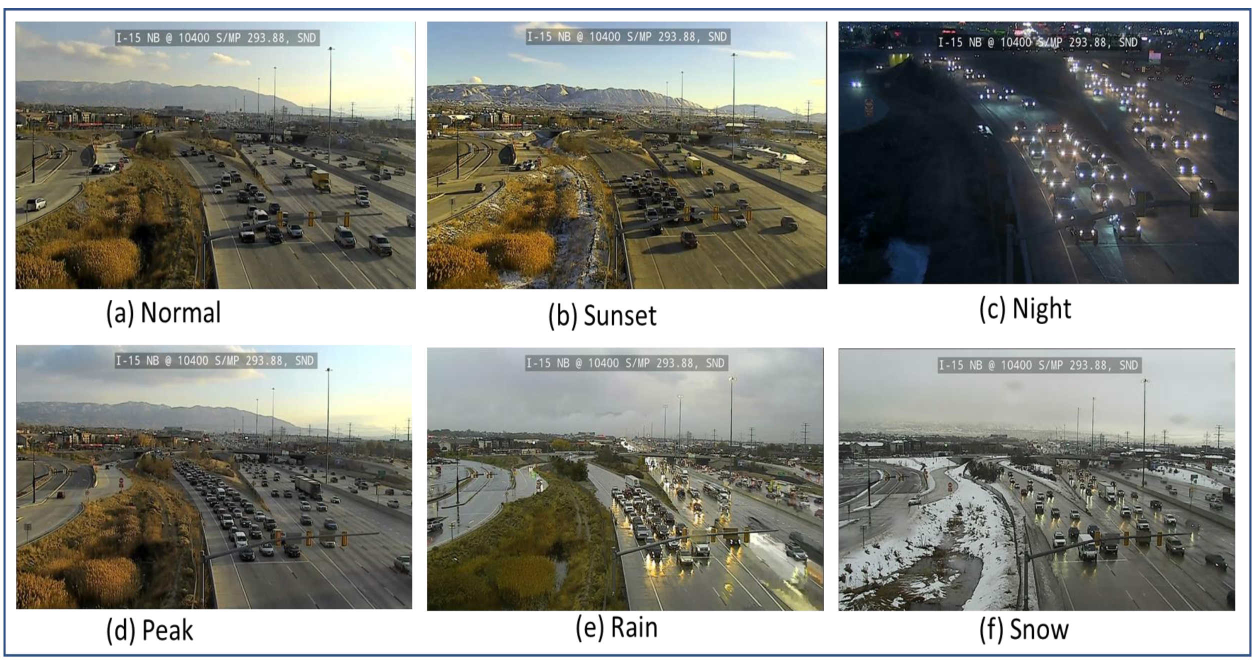


Figure 13: Different Lighting Conditions and Season Variation in the Evaluation Dataset

3.2 Data Collection for Model Retraining

The state-of-the-art object detection model You Only Look Once (YOLO) was used as an object detection algorithm. YOLO uses the COCO dataset – with 80 classes – to train the YOLO model, which consists of the normal condition images for the training dataset. However, the real-life problem poses challenging scenarios such as variations in lighting, and seasonal variations. Additionally, during queue development, there will be occlusion for many cars which makes the object detection task more intricate. To solve this, we must retrain the model using the relevant dataset. Hence, the research team collected more than 8200 images of cars and trucks from open-source platforms like Google under creative commons licenses (not subjected to copyright while using images from Google or any open sources for academic purposes but not for commercial purposes) and hand-labeled them manually to make the model more robust. Various lighting and seasonal conditions were combined in the training image while collecting data. Images were collected, for example, of snow conditions at night and normal light.

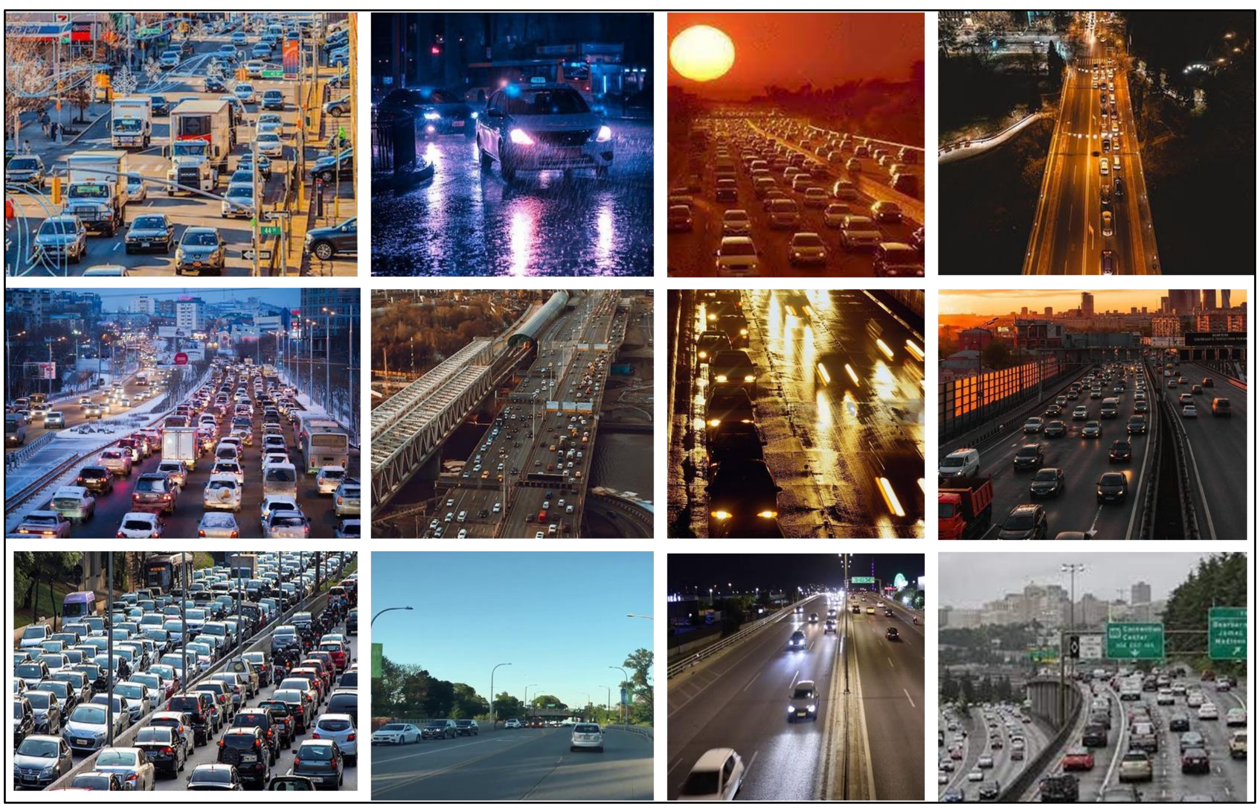


Figure 14: Lighting Variation in Training Image Dataset

Different kinds of cars like SUVs, sedans, etc., were included in the training datasets. Figure 14 and Figure 15 show the sample frame for each challenging case that was labeled manually for model retraining.

Figure 15: Seasonal Variation in Training Image Dataset

4. RESEARCH METHOD

This section describes how object detection and tracking algorithms were used to measure the queue length and queue delay. Figure 16 shows the framework for object detection and queue length detection. Primarily, data collection was done with the help of UDOT-TOC. Preprocessing of data is completed before feeding the frame into the object detection model. Object detection is followed by a tracking process to collect insights about the traffic parameters.

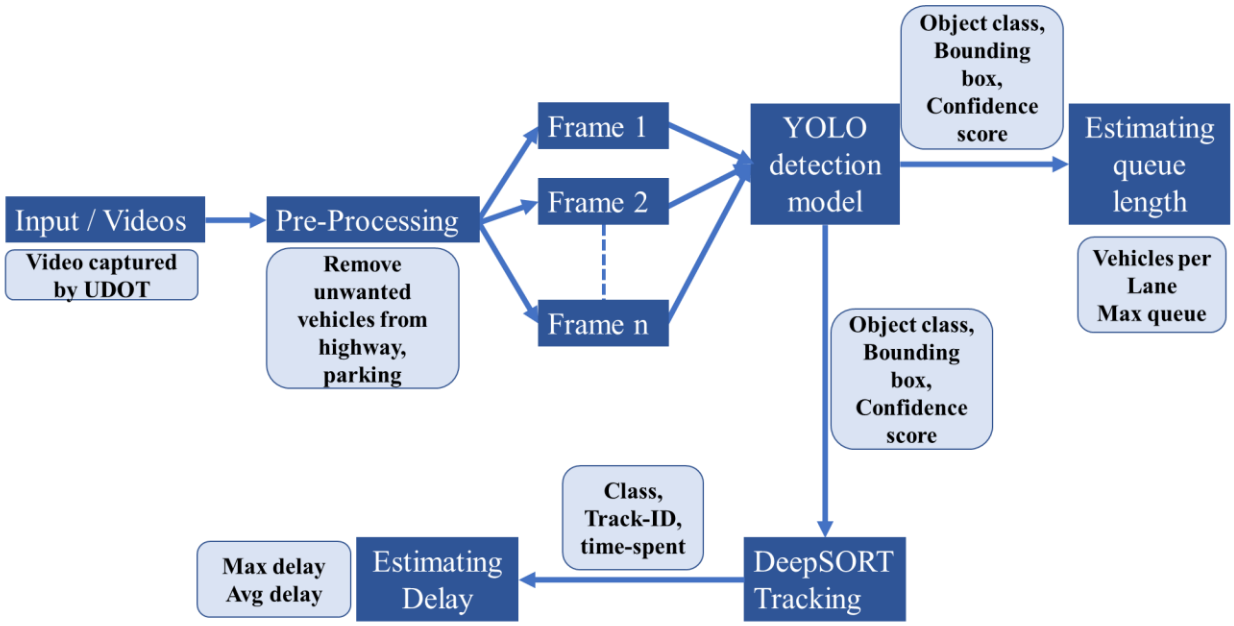


Figure 16: Flowchart of Methodology Framework

4.1 Data Collection

Data capture was done with the help of the UDOT-TOC team from an office in Salt Lake City, shown in Figure 17. Since the traffic camera available at the ramp highway section is allowed to rotate and zoom in and out while capturing the field of view, the research team and TAC members worked in coordination to fix the orientation of cameras, zoom ratio, and quality of the video to capture the reality of the on-ramp's traffic video.

Figure 17: Utah Department of Transportation - Traffic Operations Center

(*Source:* udot.utah.gov)

4.2 Data Preprocessing

Data preprocessing was done to remove vehicles not on ramps to reduce the false positive error. Vehicles from the highway, parking places, and other auxiliary roads were eliminated by masking all other portions of an image except the region of interest by manual coding. Figure 18 shows the difference of preprocessing in obtaining data accuracy. If preprocessing is not done, vehicles located adjacent to, but not on the ramps, will produce a false positive. Input frame to the object detection and tracking framework will be masked, however, the result such as bounding box, vehicle class and tracked time will be displayed in the original frame. Figure 18 shows the input frame with and without masking and the corresponding results.

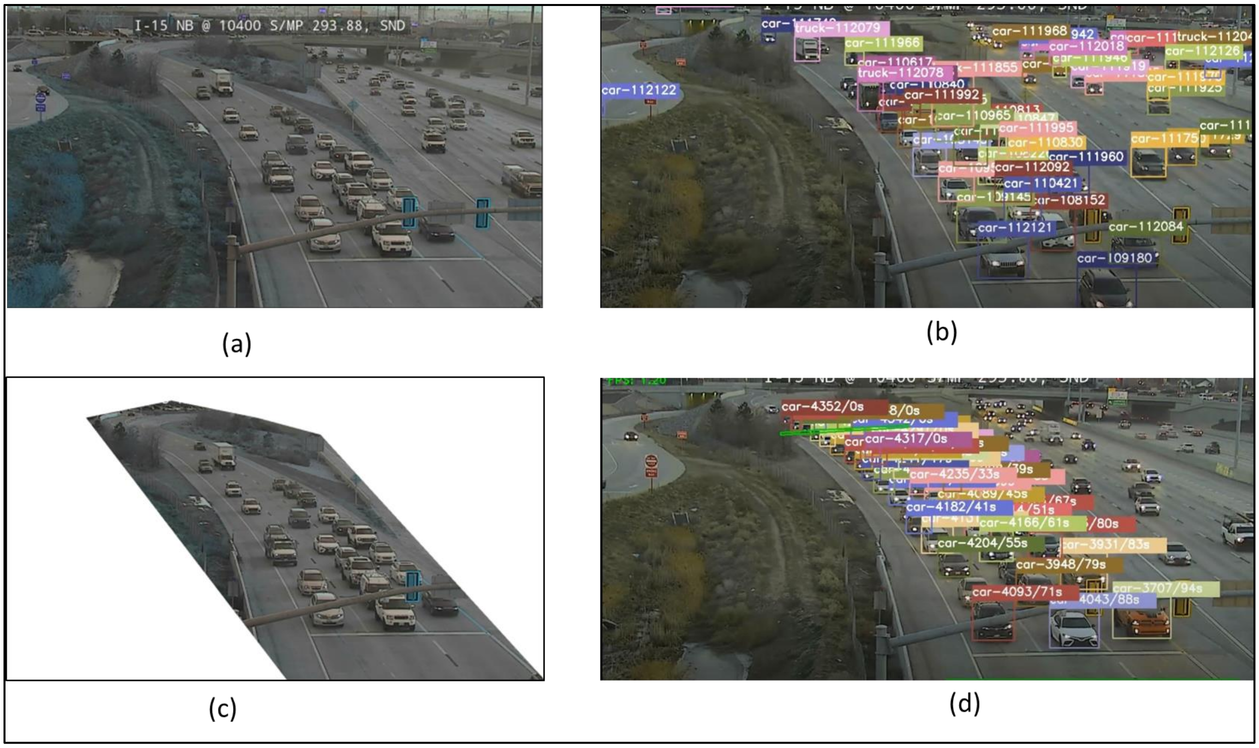


Figure 18: (a) Input Frame without Mask (b) Result for Frame without Mask
(c) Input Frame with Mask (d) Result for Frame with Mask

4.3 Object Detection

Object detection means finding the presence of an object in a given frame from a designated number of classes. Videos are segregated into multiple frames and those frames are passed into the object detection model like YOLO and the result is obtained about the object class and their spatial location in the bounding box. Many prevalent object detections can be found in practice these days; however, YOLO provides real-time performance of object detection and recognition with higher accuracy. YOLO finds the object within the frame if the object falls within 80 classes. Followed by object detection, the next step is object recognition which assigns a class - out of 80 total classes - to a detected object. Figure 19 shows the detection and recognition results obtained from the YOLO basic model. The information in the bounding box is the class of vehicle i.e., car, and confidence score out of 1 formatted as Vehicle class: confidence score (Car: 0.47)

Figure 19: Object Detection and Recognition Using YOLOV4.

Yolov4 consists of the input, backbone, neck, and head block for the object detection model. It was trained using the coco dataset and was able to classify 80 classes of objects. The main advantage of yolov4 over the previous version is that it uses CSPDarknet53 as its backbone. CSPDarknet53 is the cross-stage partial network that utilizes the prior input and combines it with the present input before passing it to the dense layer. This greatly reduces the computational bottleneck present in the previous version making it 20% faster. Head is derived using yolov3 which does the final detection process and provides the bounding box to the objects. The neck gathers the features together from the different stages of the backbone and passes it to the head for the detection process while the backbone is used for the feature extraction process. Additionally, yolov4 has introduced a bag of freebies and a bag of specials. A bag of freebies helps in increasing the performance of the model by enhancing and augmenting the training dataset of the model. Some of the augmentation methods were photometric distortion, geometric distortion, Mixup, and Cutmix. A bag of specials will increase the performance significantly though it also increases the inference time too. The bag of specials implemented in the backbone is mish activation, cross-stage partial connection, and multi-input weighted residual connections. (MiWRC). Likewise, the bag of specials used for the detector is mish activation, SPP-blocks, and the Pan path aggregation block (Bochkovskiy et al., 2020).

However, the original YOLOV4 model has lower accuracy during low lighting conditions and seasonal variation. Hence, the YOLOV4 model was retrained using the image frame consisting of challenging situations. The retraining process of the YOLO model comprises image collection, image labeling, and retraining of a deep learning model.

4.4 Object Tracking

Object tracking keeps the information about the location of an object from one frame to another frame continuously and associates the location of an object from the previous frame to the current frame based on the metrics defined by certain algorithms. There are many prevalent tracking algorithms in use these days, however, DeepSORT is a state-of-the-art tracking algorithm that handles the problems of ID switching and occlusion better than any other tracking algorithm. The information about the bounding box (left top, right bottom, width, and height of the bounding box) from the object detection model is passed to the object tracking algorithms. Object tracking algorithms keep track of the location and calculate the similarity scores of the bounding box between previous frames and current frames and pass the tracking ID from the previous frame to the current frame if the score is above a defined threshold. Every object is provided with a unique ID when they enter the frame which is kept track of until it exits. Figure 20 shows the result of the object detection and tracking algorithms. The information in the bounding box shows the vehicle class, unique ID, and total time duration between when they enter the frame and the current frame. The information is formatted as vehicle class – tracking ID/ time spent as shown in Figure 20 above the bounding box.

DeepSORT is the modification of the SORT algorithm which has handled the problem that arose due to the occlusion of the detected objects. The SORT algorithm uses the bounding box prediction, Kalman filter, and IOU matching techniques. DeepSORT uses the concept of the SORT algorithm with cascade matching, and deep appearance descriptor, along with Hungarian algorithms. The Kalman filter uses linear velocity methods to predict the future position of an object, which is later matched with the real detection in the future frame to calculate the Intersection over Union (IOU) matching. IOU gives a quantitative score of how much two bounding boxes are similar to each other based on location in the image and their corresponding size. If the IOU matching is above the defined threshold, then the tracking ID is transferred and the Kalman filter will update its tracking data based on the current frame. Additionally, deep appearance also provides a similarity score based on cosine distance. DeepSORT averages the scores from IOU matching and cosine distance before finalizing the similarity of a tracked object between the previous frame and the current frame. Cascade matching is the extension to IOU matching. Cascade matching has taken into the temporal dimension. Cascade tries to match the latest detection with later IDs and older with older IDs. This prevents the ID-switching problem between earlier detected objects and later detected objects (Wojke et al., 2017).

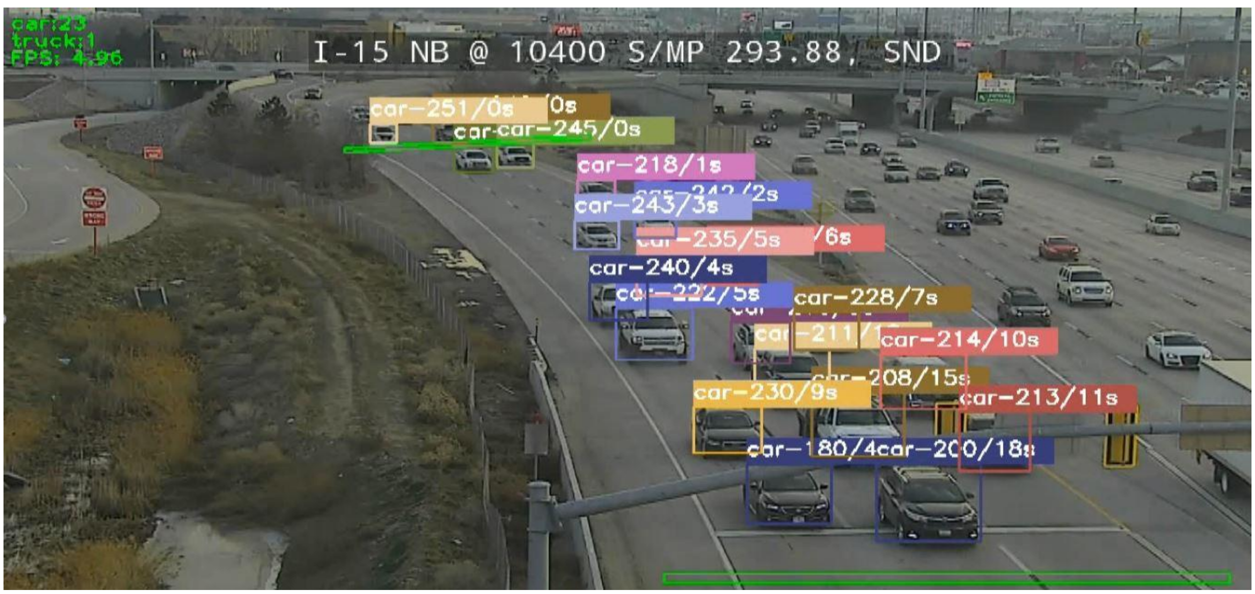


Figure 20: Object Detection and Tracking

4.5 Queue Length Detection Using Computer Vision

Traffic queue length refers to the number of vehicles waiting in a line to move forward on a road or at an intersection. The length of a queue is important information to design a signal timing to reduce vehicle ramp spillover. This aids in adjusting the signal timing so that the queue length does not exceed the ramp's capacity, preventing spillover. When a vehicle passes through an entry zone or an exit zone, the counter increases or decreases respectively giving the total number of vehicles in each lane at present frames formatted as *“lane number: total number of vehicles per lane”* shown in Figure 21.

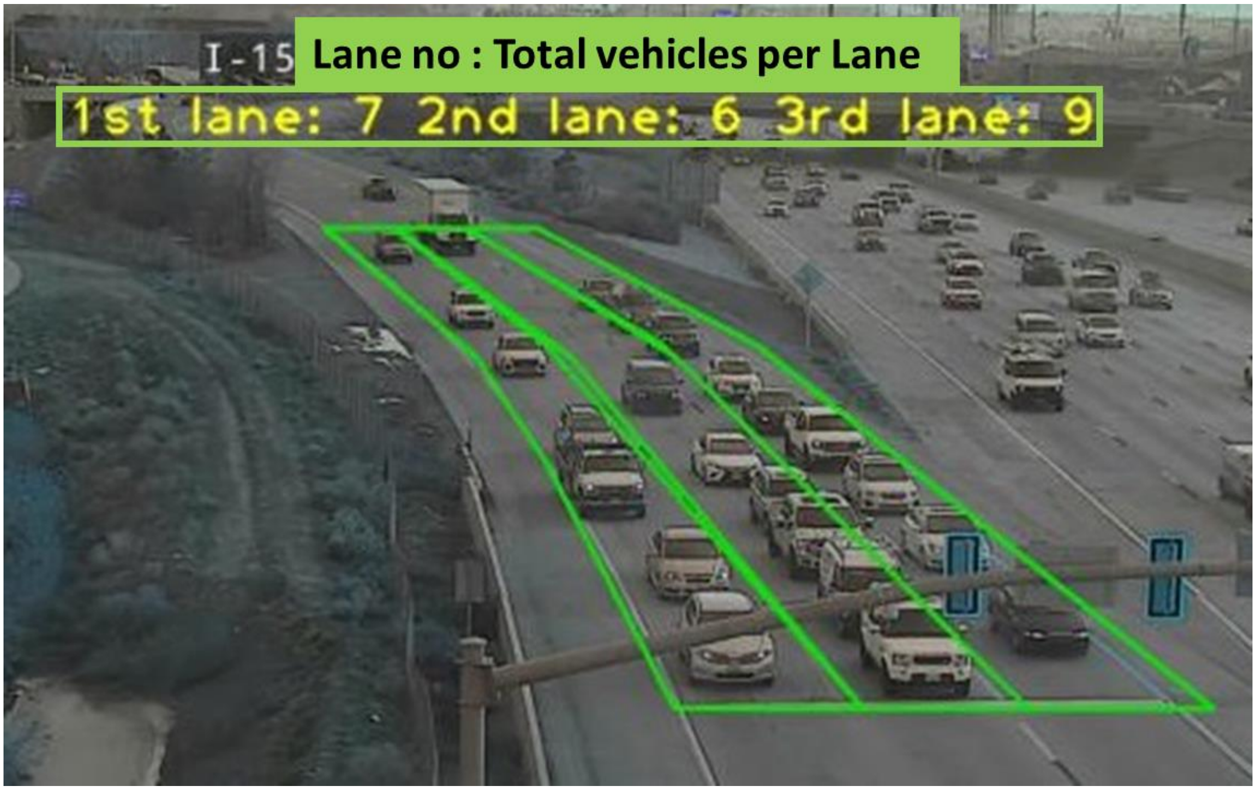


Figure 21: Queue Length Detection

4.6 Queue Delay Calculation Using Computer Vision

Queue delay is the measure of time given in minutes faced by each vehicle from when they enter the ramp until it exits from the traffic signal to merge into the mainline. When a vehicle enters the frame, algorithms assign a unique ID to each object and begin tracking its time until it exits the exit zone. The analytical framework tracks the total delay experienced by vehicles as they pass through ramps and stores vehicle class, tracking ID, and time spent when exiting. The information is displayed on top of the bounding box formatted as vehicle class - Tracking ID/ total time spent as shown in Figure 22. This information is an input to the calculation of the system's maximum delay as well as average delay. Delay calculation is done for each vehicle with the help of tracking ID and frame number. When a vehicle enters the queue, a unique ID is assigned to each vehicle, and the frame number of the video for that particular tracking ID is stored in a dictionary. Using these keys and the value of the python dictionary, the total time spent in the frame is calculated. The total time spent by each vehicle is tracked until the vehicle exits, as shown in Figure 23.

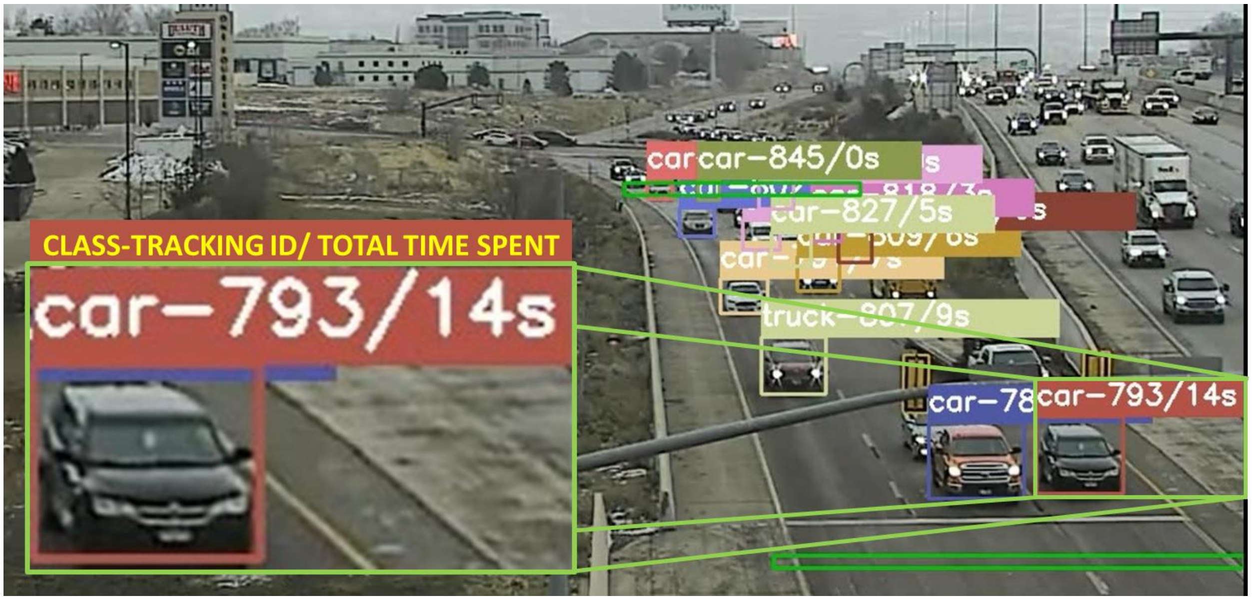


Figure 22: Queue Delay Calculations

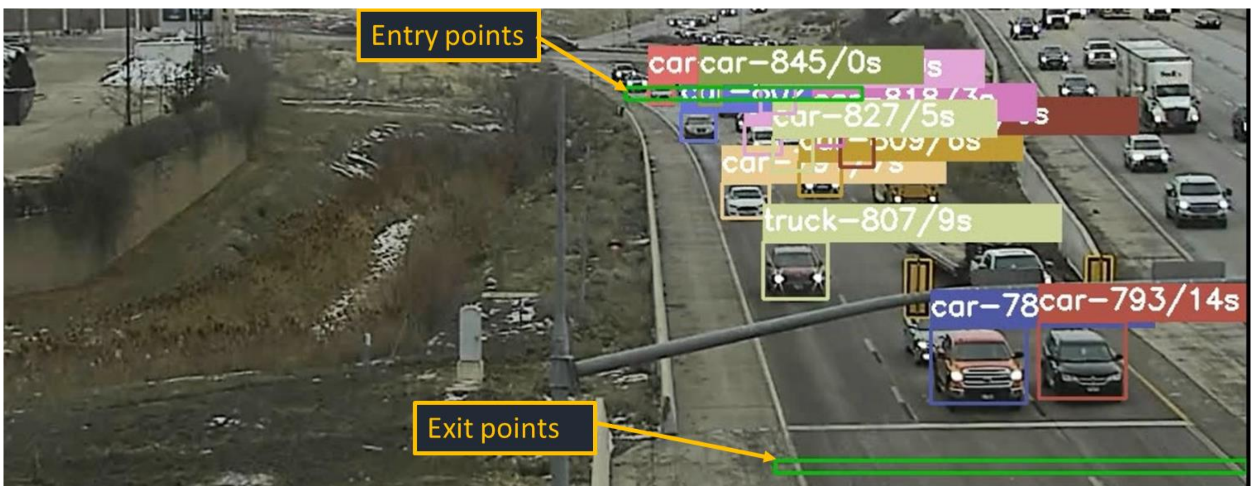


Figure 23: Entry and Exit Demarcation

4.7 Queue Delay Calculation Using Queuing Graphs

Queuing graph is used to calculate the maximum delay, maximum queue, average delay, and average queue using the queuing graphs. The total number of vehicles entering from the entering zone along with their entering time and the number of vehicles exiting from the exit zone along with their exiting time is extracted from the video. The exiting and entering curve are plotted as shown in Figure 24 indicated by a blue solid line and a red dotted line. Time in seconds is plotted on the x-axis, and cumulative entering and exiting is plotted along the y-axis. The vertical distance between the exiting and entering curves gives the value of queue length, and the horizontal distance between the two curves gives the delay of a vehicle. Hence, the maximum horizontal distance gives the maximum delay (99.81 secs in this Figure 24), and the maximum vertical distance provides information about the maximum queue length (35 in this Figure 24). The maximum delay is indicated by the black dotted line while the maximum queue is indicated by a dotted orange line. The total area (plotted in green) gives the values of the cumulative delay of the system.

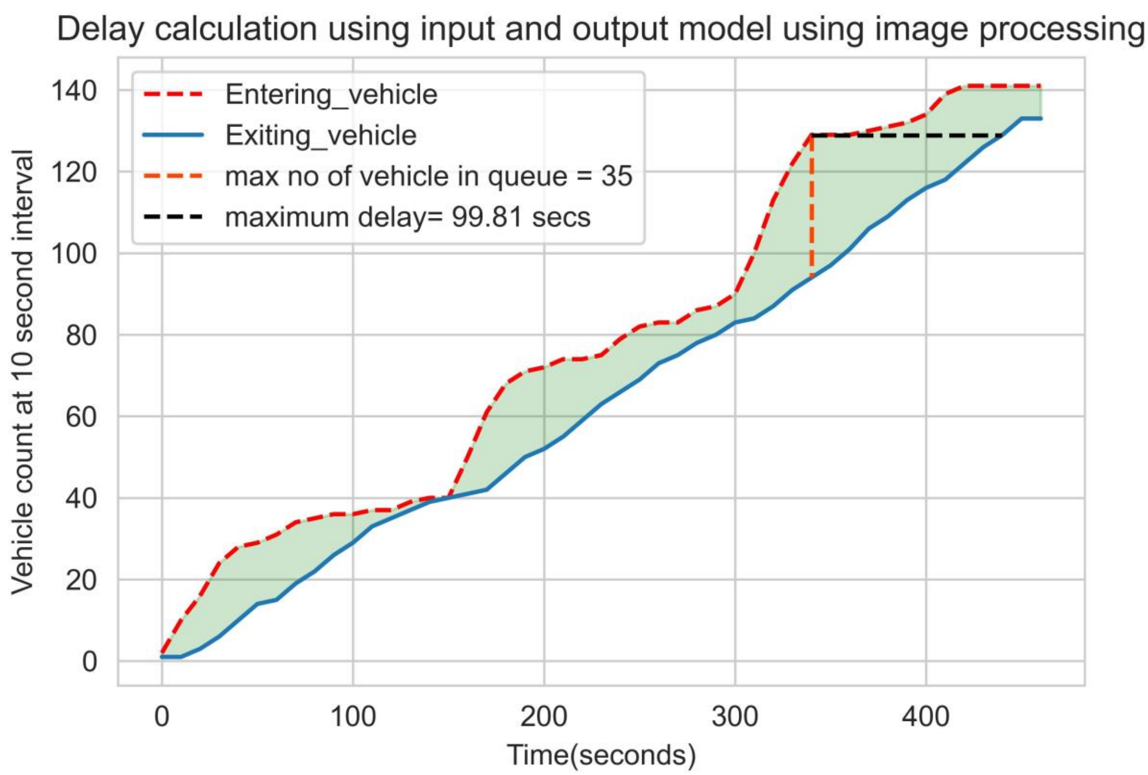


Figure 24: Queuing Graph

5. RESULTS AND DISCUSSIONS

Traffic queue length and queue delay are very important parameters for ramp performance measurements. Queue length and queue delay can be measured from the on-ramps using different sensor technologies. However, this research proposes an innovative technology using the integration of artificial intelligence and computer vision principles to extract the ramp's traffic measurements – queues and delays. Object detection and tracking were used to determine the required parameters. Since the accuracy of a framework for object detection and tracking is highly dependent on the object detection model, it is necessary to develop a robust object detection model with better accuracy.

5.1 Object Detection Model Performance Measurements

Multiple object detection models are used for object detection depending upon requirements. However, for traffic detection and analysis, a real-time detection model is required, and the YOLO object detection model can provide a better result at a faster rate. Although the original YOLO provides a better model with greater speed and accuracy, it falls short when faced with challenging conditions such as lighting issues and seasonal variation. Seasonal variations like snow and rain produce an extra noise in the frame, making object detection difficult and less accurate. Similarly, input frame size plays a significant role in achieving accuracy for object detection models but at the cost of speed. Hence, it is always desirable to find the optimal frame size considering the trade-off between frame size, speed, and accuracy. Video samples of 1 minute, 5 minutes, 10 minutes, and 15 minutes were taken into consideration for analyzing the model. A total video length of 205 minutes and 6150 frames was evaluated manually to find out the ground truth. Table 5 summarizes the accuracy result in different conditions for different frame sizes. Accuracy of detection and tracking increases when there is an increase in input frame size to the model. Additionally, accuracy depends upon external factors such as light and weather conditions. The highest accuracy of 72.56 % was achieved with a frame size of 1024 * 1024 during the normal condition, while the lowest accuracy of 45.01% was during the nighttime. The developed framework is more accurate in determining the queue length in comparison to calculating the queue delay. The speed of processing video frames decreases as the frame size increases. Evaluating speed and frame size in multiple combinations, a frame size of either 960 * 960 or 1024 * 1024 is the optimum frame size as it provides competitive accuracy with justifiable speed.

Table 4: Accuracy of Base YOLOv4 Model

Input Frame Size		**416 * 416**	**512 * 512**	**736 * 736**	**960 * 960**	**1024 * 1024**
Normal daylight	*queue length*	70%	71%	71%	72%	73%
	queue delay	66%	67%	67%	68%	69%
Sunrise/sunset	*queue length*	65%	65%	68%	69%	70%
	queue delay	64%	66%	66%	66%	67%
Nighttime	*queue length*	45%	46%	46%	46%	47%
	queue delay	45%	45%	45%	46%	46%
Rain	*queue length*	55%	56%	57%	57%	57%
	queue delay	51%	52%	52%	52%	53%
Snow	*queue length*	49%	49%	49%	50%	50%
	queue delay	46%	46%	48%	48%	49%

Table 4 shows the accuracy percentile of the base YOLO model concerning changes in frame sizes at different lighting and seasonal variation conditions. Although the base model shows greater accuracy during normal daylight conditions at larger frame sizes, it fails to provide competitive accuracy when there are challenging conditions like poor lighting and external noise such as rain, or snow.

5.2 Retraining Object Detection Model

The YOLO base model provides an accurate and fast result; however, it fails to perform under challenging environmental conditions. The disadvantage of the base YOLO model can be eliminated by retraining the model using manually labeled images under a variety of challenging conditions. The information about the dataset used for depicting challenging conditions was briefed in section 3.2. The model was trained using the YOLOV4 base network with new data. Figure 25 shows the training process of the YOLO model; training loss is the average loss in each training epoch where mAP represents the mean Average Precision. Mean average precision provides a mean precision value considering all the precision of all the classes. Precision gives the ratio of the true positives value over the sum of true positives and false positives. The higher the mAP, the better the model generalization power will be. The final model with superior performance was developed after training the model for 120 hours in 6000 batches.

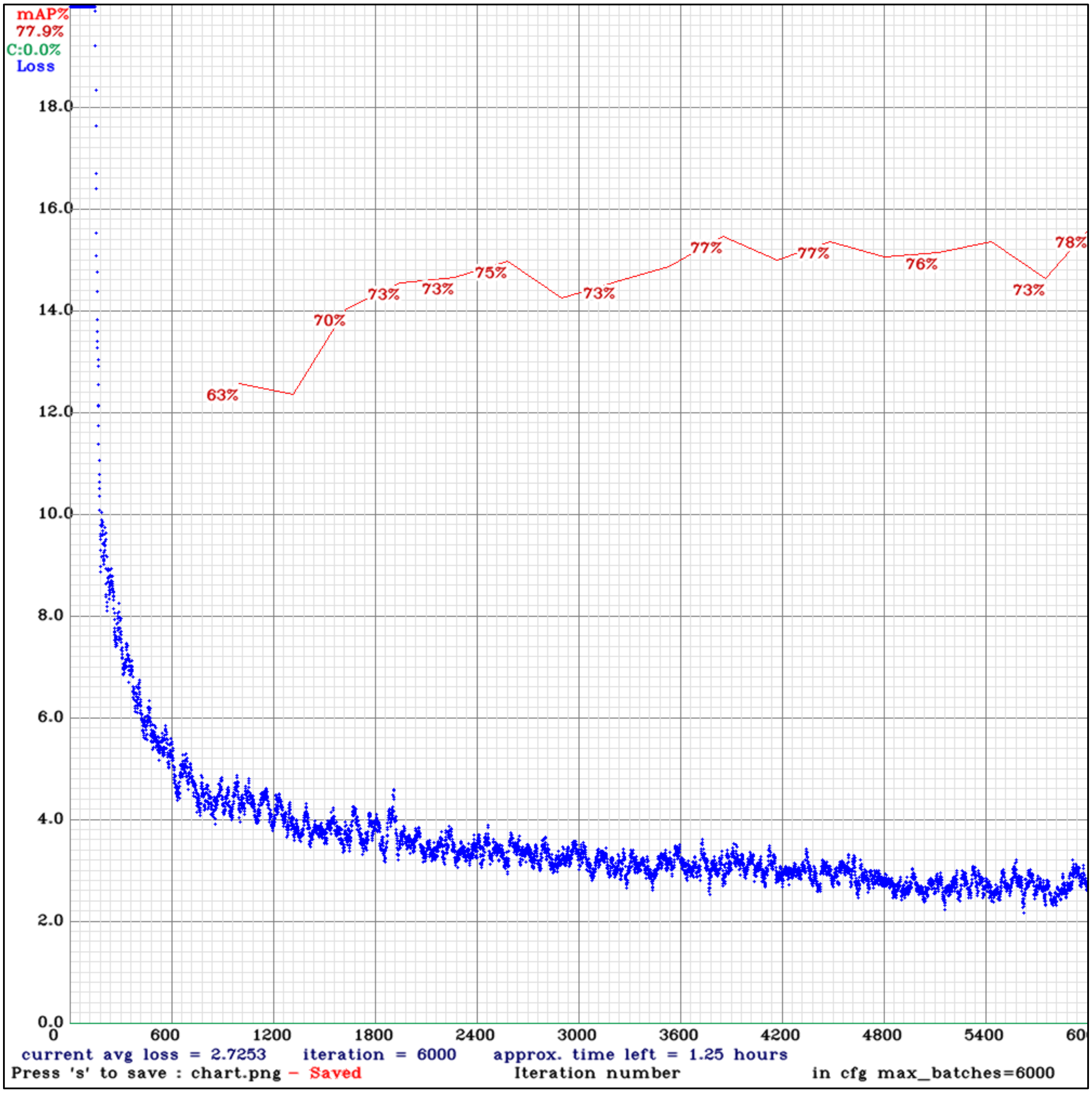


Figure 25: Model Retraining Chart Showing Loss and Average Precision

Figure 26 below shows the comparison results between the base model and retrained model visually. The retrained model is better even when there are poor lighting conditions and seasonal variations with different noise. So, the retrained model will be used for further analysis.

Figure 26: Comparison of Base YOLO Model and Retrained YOLO Model

Table 5: Accuracy of Retrained YOLOv4 Model

Input Frame Size		416 * 416	512 * 512	736 * 736	960 * 960	1024 * 1024
Normal daylight	*queue length*	78%	78%	80%	81%	83%
	queue delay	75%	76%	78%	79%	82%
Sunrise/sunset	*queue length*	77%	77%	78%	80%	80%
	queue delay	75%	76%	76%	76%	79%
Nighttime	*queue length*	74%	76%	75%	78%	79%
	queue delay	74%	75%	75%	77%	79%
Rain	*queue length*	68%	69%	69%	72%	75%
	queue delay	66%	67%	68%	72%	74%
Snow	*queue length*	68%	68%	70%	71%	74%
	queue delay	69%	69%	71%	73%	74%

Table 5 shows the accuracy of framework with the retrained YOLOv4 model as object detection algorithms. The framework can estimate the queue length with up to 83% accuracy in normal time conditions, while it estimates queue delay with 82% accuracy in normal conditions. Comparing the average performance, the framework detects the normal time with higher accuracy, while it underperforms during the rain and snow. This is because snow and rain appear as external noise during image processing.

5.3 Computer Vision vs Queue Graph

As discussed in Section 4 above, computer vision and queue graphs can individually determine the queue length and queue delay. Sharma et al. (2007) verified that queuing method is effective to extract the real-time traffic parameters by comparing them with manual ground truth. Table 6 shows the discrepancy in delay estimation using the queue graphs and computer vision methods. The error column describes the difference in percentage between the two methods and the discrepancy between them is less than 0.02%. This confirms the vision method provides a competitive accuracy for queue length and queue delay detections.

Table 6: Queue Delay Comparison Between Algorithms and Queuing Method

Cameras	Video Length	Avg Delays(algorithms)	Avg Delays (queuing)	Error%
10400 S	1:31:06	44.4032	44.4	-0.007207207
500 S	1:09:36	17.81	17.81	0.022459293
Beck Street	1:15:00	21.48	21.48	0.023227747
11400 S	0:54:05	19.671	19.67	-0.005083884

5.4 Image Processing Technique Verification with Ground Truth

Image processing techniques need to be verified using ground truth data. Randomly extracted videos of a duration of around 5-10 minutes are sampled and the graph showing the difference between them is plotted in Figure 27. The top left shows the queuing graph for the ground truth data. Ground truth data were collected through manual observation. The top right graph shows the queuing graph plotted using the data extracted by the retrained image processing models. The bottom portion of the figure shows the image difference in the queue between the ground truth and retrained model. The green solid line shows exiting vehicles using ground truth while the black solid line is the exiting vehicles using the image processing algorithms. They are nearly identical, indicating that the image processing results are very promising. The minor discrepancies when observed visually are due to the presence of a mast arm in the video near the exiting zone which produces the ID switching problems.

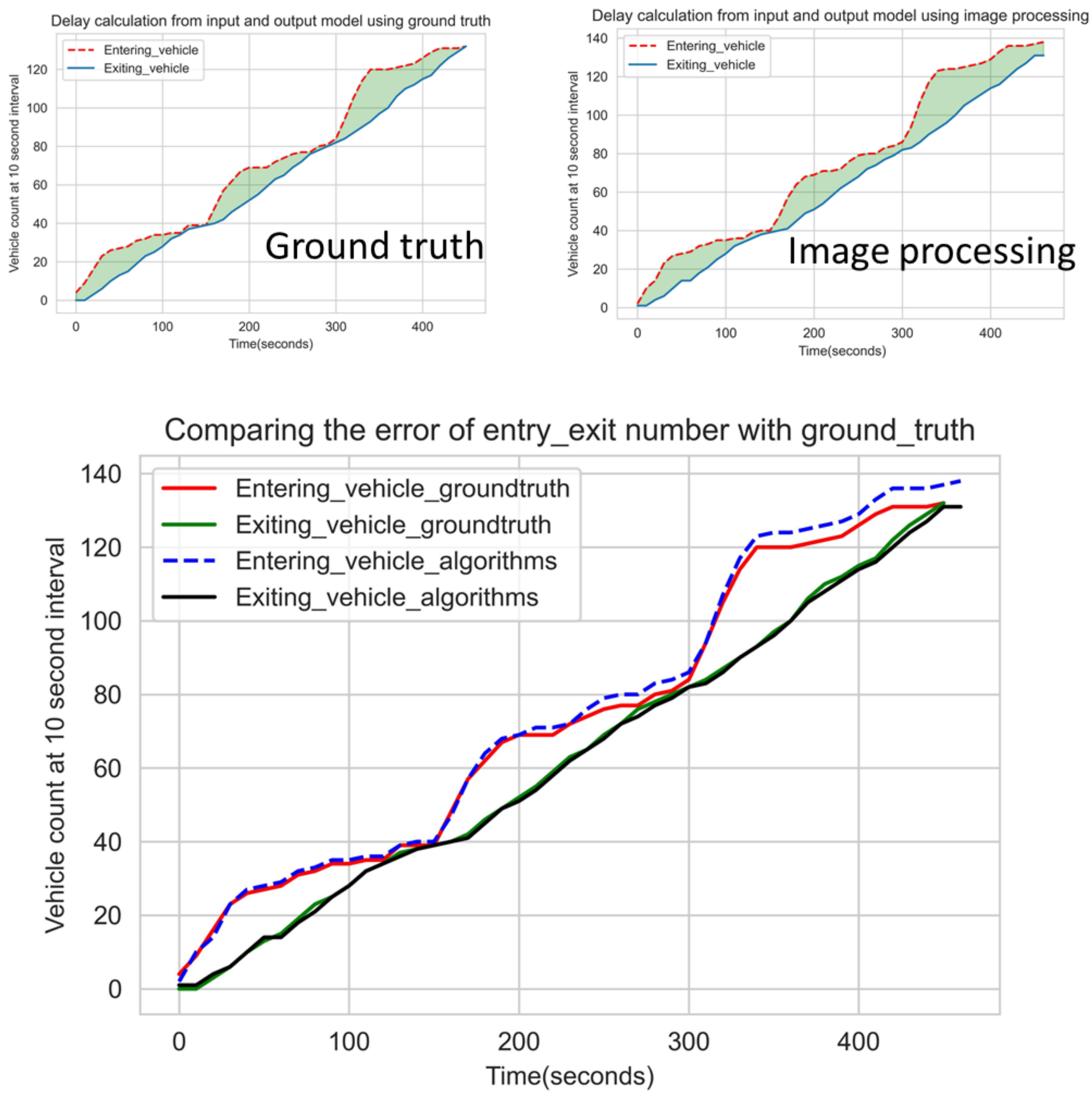


Figure 27: Comparison of Ground Truth with Image Processing

5.5 Signaling Strategies with Queue Analysis

Image processing can generate a multi-array of data from the same video using minor modifications in algorithms. We can extract data such as vehicle arrival rate, vehicle exit rate, and delay and queue information. Figure 28 shows the distribution of these data for every 15-minute interval for a 1.5-hour video analysis. The queue parameters change when the ramp metering's exit rate or signal phase is changed. Figure 29 depicts how cumulative delay, max delay, and max queues change when the exiting rate changes while the arrival rate remains constant.

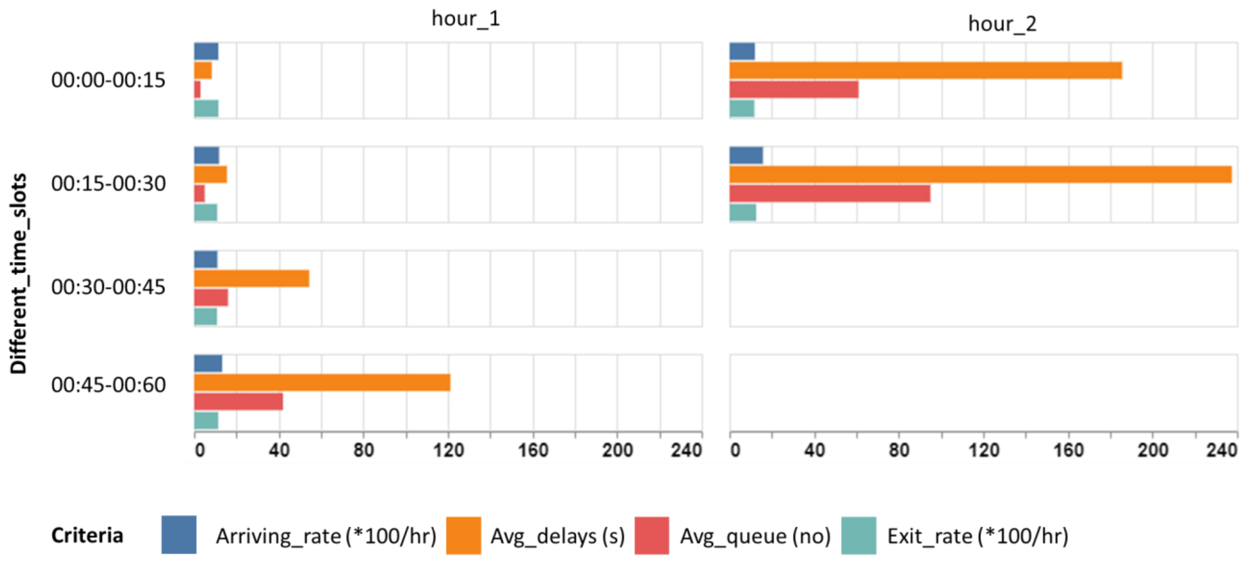
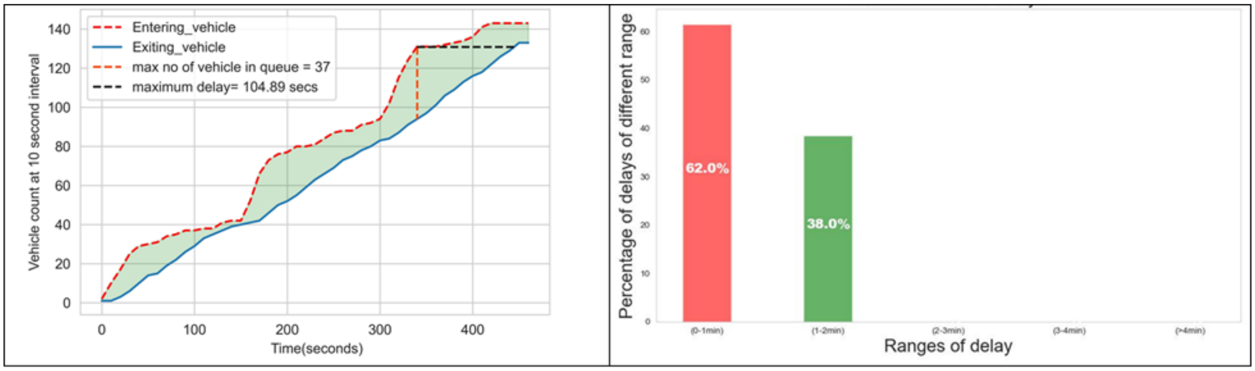


Figure 28: Ramp Performance Analysis at 15-Minute Intervals

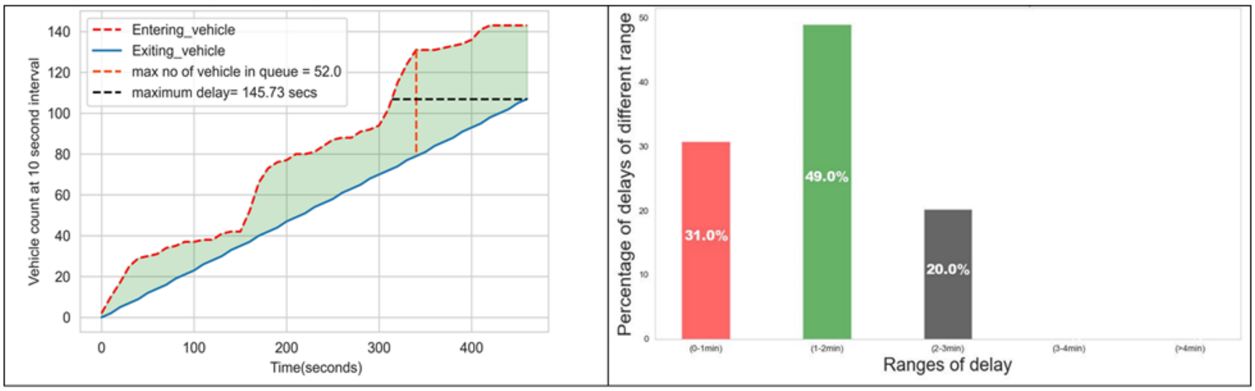
In Figure 29, we can see graphically that cumulative delay increases when the exit rate of a signal is decreased. As expected, when there is a lower exit rate – keeping the entry rate constant – queues build up and delays lengthen, as shown in the figure above. The opposite phenomenon is seen when the exit rate is increased. Table 7 shows the numerical value of the parameters and the delay distribution. When the exit rate was increased by 20%, all the vehicles faced a delay of less than 1 minute. In the current exit rate, no vehicles face a delay of more than 3 minutes. The current exit rate refers to the normal exit rate which means the exit rate in real-field scenarios at the ramps of Utah when video was recorded.

Table 7: Ramp Parameters Variation with Signal Adjustment

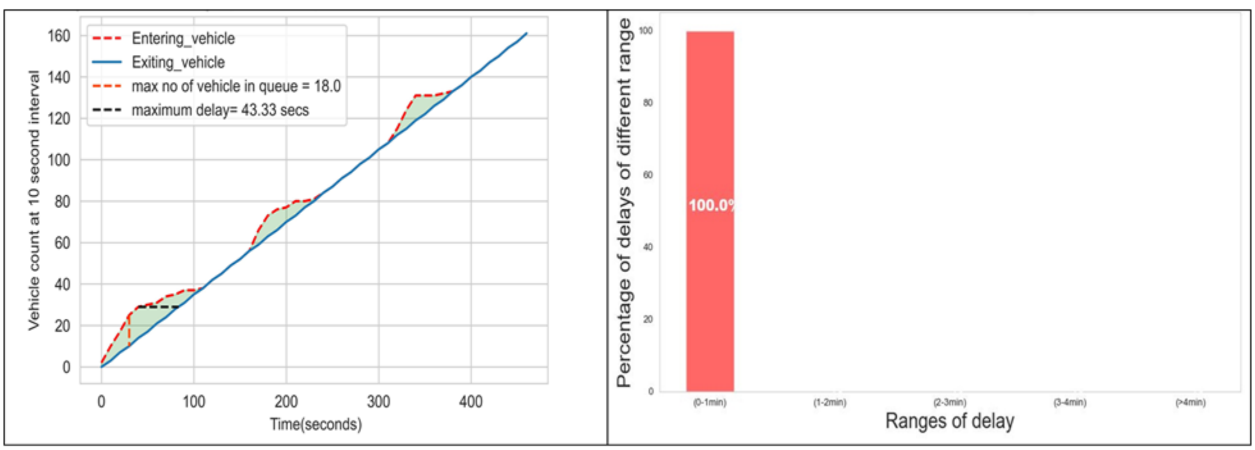
	Normal exit rate	**0.8* Normal exit rate**	**1.2* Normal exit rate**
Max_queue	37 veh	52 veh	18 veh
Max_delay	105 sec	146 sec	43 sec
Delay distribution by percentage factor			
0-1 min	62%	31%	100%
1-2min	38%	49%	0%
2-3min	0%	20%	0%
3-4min	0%	0%	0%
>4min	0%	0%	0%



(a) Normal exit rate; Queue Delay (left), Delay distribution(right)



(b) 0.8 times Normal exit rate; Queue Delay (left), Delay distribution(right)



(c) 1.2 times Normal exit rate; Queue Delay (left), Delay distribution(right)

Figure 29: Queue Length and Delay Variation in Response to Signal Timing

However, when the exit rate is decreased by 20%, 9% of the vehicles experience delays of more than 4 minutes. The exit rate can be adjusted using signal timing such that no vehicle in the ramp should face more delay than a certain threshold as per a standard, guideline, or policy, maintaining continuous flow in mainline traffic.

6.0 CONCLUSION

In this project, the research team used image processing techniques with deep learning processing algorithms to extract data about ramp queues and delays using the video from UDOT's existing traffic camera videos. The team obtained the data, preprocessed and developed the framework to process the data, extracted the information, and tested the traffic signal control strategies using analysis from queuing methods and computer vision models.

6.1 Findings

The research team has experimented with multiple frameworks with multiple combinations of object detection and tracking algorithms to determine the most accurate framework. The team has also experimented with different frame sizes of videos to find the optimum frame size. The research team has explored multiple statistical tools to quantify ramp performance using the data extracted using visual methods. The major findings of this research are as follows:

a. Image processing provides reliable and promising sensor technology to extract real-time traffic information.
b. Computer vision technology can use existing traffic camera video footage to measure information like queue length and queue delay.
c. The optimal frame size considering the trade-off between speed and accuracy is 1024 * 1024.
d. Finally, the results from computer vision using the retrained object detection model are compared with the standard queuing approach and ground truth data. The result was found very close to ground truth with acceptable accuracy.

6.2 Limitations

The object detection and framework have competitive accuracy, however, there are a few limitations to these studies.

a. Being a visual system, the system cannot measure the full extent of a queue and, hence, delay, in cases where there is a curved on-ramp.

b. Location, orientation, and height of the camera play a major role in data extraction accuracy. The mounting and orientation of the camera during installation need to be calibrated to the specific conditions of any ramp. Stable video footage is extremely important, which suggests that post-mounting rather than mast arm mounting is the preferred approach. Additionally, the resolution quality of the camera affects the results. Hence, all the existing ramp cameras might not qualify for this framework because of these requirements.
c. The computing power of the machine required is high and edge computing might not be feasible for this method. However, cloud computing can eliminate this limitation. With the development of advanced and lighter deep learning frameworks for object detection and tracking algorithms, in the near future, it is expected to be compatible with edge computing devices.

6.3 Implementation Plan

UDOT's Research program emphasizes the implementation of research results to achieve the maximum benefit from research efforts and expenditures. This section describes a plan for implementing a computer vision camera system for UDOT on the I-15 corridor in the Wasatch Front urban area. The implementation plan consists of four parts:

- Camera functionality
- Image Processing
- Mounting and orientation guidelines
- Suitability of existing metered ramp alignments (37 metered ramps on I-15)

6.3.1 Camera Functionality

Currently, UDOT has three camera devices deployed: AXIS CCTV, Gridsmart Vehicle Detection, and Flir Vehicle Detection. For this research, only the AXIS CCTV camera was tested for computer visioning, and it proved to be generally reliable. Video images from the other cameras were determined to be of lower resolution with grainier images, leading to greater uncertainty in visioning results.

UDOT is evolving its ramp cameras toward a network camera model, the AXIS Q6215-LE PTZ Network Camera (also known as the AXIS H.264). As implied, this camera model can be assigned to a network with typical login credentials, as would be typical of network devices managed by UDOT's Traffic Management Division.

The present research has tested computer vision under adverse "vision" conditions, including conditions of reduced daylight and occluding weather events (i.e., rain, snow, wind). The image training using UDOT's existing camera technology has established that the camera visioning technology can still be effective under adverse visioning conditions. That said, improving camera technologies is certain to improve visioning capabilities under these less-than-favorable conditions.

Regarding the AXIS Q6215-LE, this camera improves upon the currently deployed AXIS version in the following ways:

- provides better nighttime imagery (up to 400m/1300ft).
- reduces motion blur in low light conditions which helps to maintain the quality of video during low light conditions.
- handles scenes with strong backlight with WDR features.
- It has a wiper that can remove rain, dust, and snow.

The AXIS Q6215-LE also has superior data storage and compression features (H.264 or MPEG) which facilitate computer vision methods. Other features of this camera which will improve computer visioning capabilities include:

- Horizontal field of view – 58.6 degrees
- Variable focus – 6.7-201mm, F1.6-5.3
- Environmentally hardened for temperature (NEMA TS 2 compliant)

The research team believes the deployment of the AXIS Q6215-LE camera will increase overall computer vision accuracy. Further improvements in camera technology will only lead to improved results in the future. The PTZ capabilities are important to address, however. UDOT requires PTZ to view and focus on specific locations on each ramp. For a computer vision system to reliably feed a ramp metering system, the cameras need to be stationary or be returned to a pre-set status associated with the camera calibration position. There are two options for managing this. First, UDOT can install a set of stationary cameras dedicated to ramp metering detection, thus freeing up the existing PTZ cameras for custom purposes such as incident inspection. Second, UDOT could keep PTZ cameras that are designated as “dual use” stationary for the critical congested periods when accurate information on ramp delay and queue length is most critical. If, during these periods, a camera needs to be moved to focus on a special event or incident, the detection capabilities of the camera will be temporarily compromised. Further research could focus on the process of recalibrating a camera after a PTZ movement. Preliminary research suggests this can be done within a one-minute time frame.

6.3.2 Image Processing

The research team recommends Cloud Computing using Serverless GPUs (Graphical Processing Unit) for real-time processing of ramp video imagery. Another option is a GPU Cloud Server. The research team believes Serverless GPUs are superior for the following reasons:

- Provisioning of data is automatic with Serverless GPUs as opposed to manual for the GPU Cloud Server
- The Serverless GPU has a faster startup time and automatically scales video imagery.
- The Serverless GPU has a “pay-per-use” cost model, which is advantageous since the computer visioning capability will be necessary for a short period each day (e.g., two peak hour periods, AM and PM).
- Maintenance of the Serverless GPU is the responsibility of the provider (not UDOT).

Table 7 shows proforma annual cost calculations for subscribing to a cloud computer through a serverless GPU. It is assumed that computer visioning would be engaged for two hours for each AM and PM peak over a 260-day year. This is the typical operating parameter for ramp metering, resulting in 1,040 subscription hours per year. Table 7 shows a typical current (2023) hourly price for a commercial subscription ($1.80/hour). Costing can be reduced by reducing image processing by 50%, which has been shown in the current research to reliably maintain vision accuracy.

Table 8: Calculation of Annual Costs for Cloud Computing

Operating Cost per Hour	Hours per Day	Days per Year	Operating Hours per Year	Annual Operating Cost per Camera
$1.80	4	260	1040	$1,872.00
$1.07	4	260	1040	$1,112.80

6.4 Mounting and Orientation Guidelines

Regarding camera mounting and orientation, it is highly site-specific and will vary given the ramp's vertical and horizontal alignment. Generally, it is optimal for the camera to be mounted on a fixed object, such as a post, which would not be affected by wind and vibration. Mast arm mounting is not recommended for computer vision systems given the high occurrence of vibrations. Mounting on a mast arm pole is preferred. Other design parameters for optimal camera imagery are:

- Mounting height: 25-40 feet, which would conform to UDOT Standard Drawing AT 11A, which tends to specific 45', 60', or 75' poles.
- Mounting position relative to ramp stop bar: 75-125 feet behind stop bar (i.e., toward the mainline)
- Mounting location relative to the roadway: dependent on curvature/alignment of the ramp. However, a camera mounted on a typical mast arm pole, with its required clear zone setback, is suitable.

There are commercially available video system design tools that can be used to optimize camera location. External tools like JVSG (publicly available at https://www.jvsg.com/) were utilized to finalize the location, orientation, and height of the pole. The demo of software use was done at the I-15 @ SR-77 SPV SB on-ramp as shown in Figure 30. The following steps need to be taken to find the optimal place for the camera:

a. Import map from Google using latitude and longitude.
b. Lock the image in place after zooming or rotating it in the correct orientation with respect to the camera.
c. Select the camera company and model, if available on the platform. Otherwise, create your own list by entering details about the given camera system.

d. Move the location of the camera to eliminate dead zones.

e. Models the object of interest, for example, a car with their approaching speed or any disturbances, like a tree in our case.

f. Measure the setback distance using the measurement tools to find the location of the camera poles. Download the camera installation drawing from the software as below.

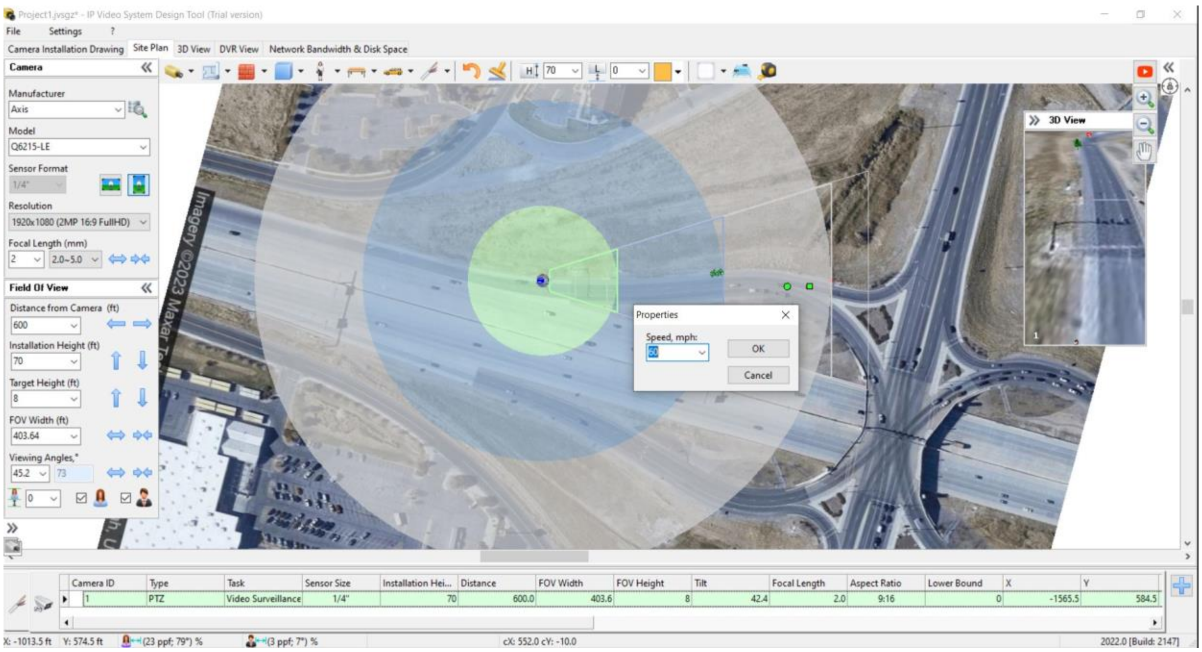


Figure 30: User Interface for Camera System Modeling Software

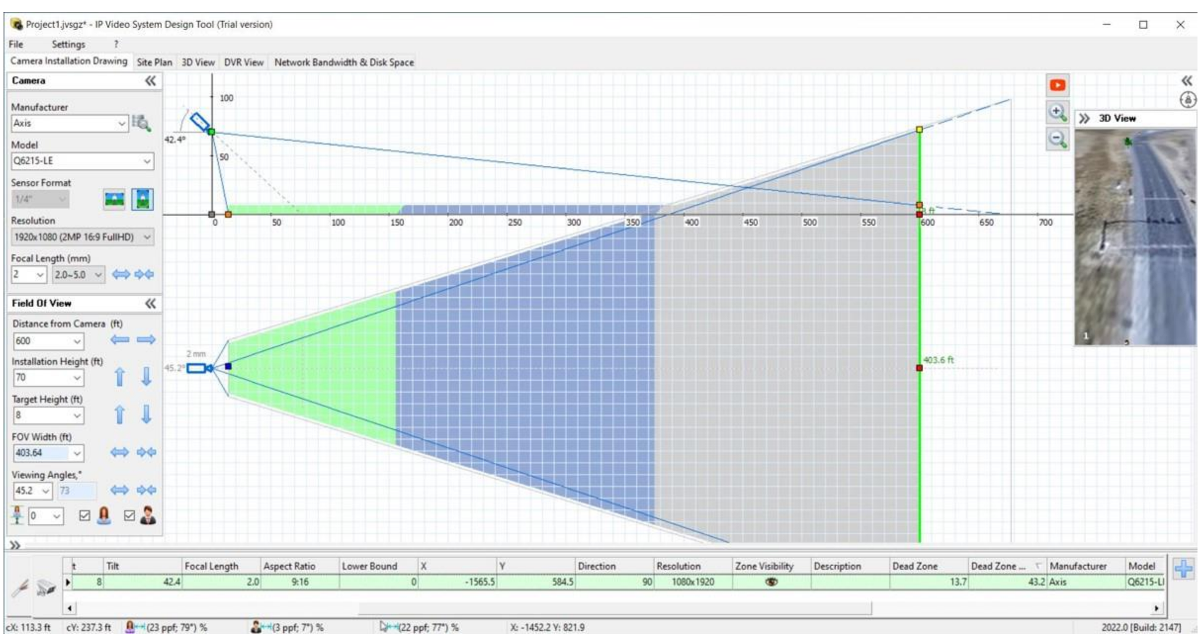


Figure 31: Installation Drawing for Camera System

6.5 Suitability of I-15 Metered Ramps to Computer Vision Deployment

This section rates the existing 37 I-15 metered ramps for their suitability for camera/computer vision deployment. A qualitative rating scale is used to describe the suitability as "Excellent," "Good," "Fair," or "Poor." Ratings have been determined based on the research team's inspection of aerial and street-view imagery of each of the 37 ramps. The imagery provided the basis for assessing ramp alignment relative to a projected ideal camera mounting location, height, and orientation. The Suitability and Rationale for each ramp are shown in Table 8.

Table 9: Qualitative Suitability Assessment of UDOT's I-15 Metered Ramps for CV

Ramp	I-15 MP	Description	RGN	Suitability	Rationale
2122	259.9	I-15 @ SR-77 SPV SB Onramp	3	Good	Good sightlines to a large gore area.
2123	260.2	I-15 @ SR-77 SPV NB Onramp	3	Excellent	Excellent sightlines to an average or smaller gore area.
2124	261.6	I-15 @ 1400 N SPV SB Onramp	3	Fair	Sightlines obstructed by trees.
2125	262	I-15 @ 1400 N SPV NB Onramp	3	Fair	Sightlines obstructed by trees, curved alignment.
2126	263.1	I-15 @ University Ave Provo SB Onramp	3	Poor	Curved alignment.
2127	263.7	I-15 @ University Ave Provo NB Onramp	3	Poor	Extremely curved alignment causes obstructed sightlines.
2128	265.3	I-15 @ Center Provo SB Onramp	3	Good	Good sightlines for > 80% of the ramp area.
2129	265.8	I-15 @ Center Provo NB Onramp	3	Good	Straight alignment, and horizontal alignment issues.
2132	268.9	University Pkwy & I-15 SB Onramp	3	Good	Good sightlines to a large gore area.
2133	269.3	University Pkwy & I-15 NB Onramp	3	Good	Good sightlines to a large gore area.
2136	270.6	Center St & I-15 SB Onramp	3	Excellent	Excellent sightlines to the ramp entrance.

2137	270.9	Center St & I-15 NB Onramp	3	Good	Good sightlines to a large gore area.
2138	271.6	800 N & I-15 SB Onramp	3	Excellent	Excellent sightlines to the gore area.
2139	271.8	800 N & I-15 NB Onramp	3	Good	Good sightlines to a large gore area.
2140	272.7	1600 N & I-15 SB Onramp	3	Fair	Curved alignment.
2141	273	1600 N & I-15 NB Onramp	3	Good	Good sightlines to an average gore area.
2142	275.2	Pleasant Grove & I-15 SB Onramp	3	Excellent	Excellent sightlines, and straight alignment.
2143	275.5	Pleasant Grove & I-15 NB Onramp	3	Excellent	Excellent sightlines, and straight alignment.
2144	276.4	A. Fork 500 E & I-15 SB Onramp	3	Good	Good sightlines to a large gore area.
2145	276.7	A. Fork 500 E & I-15 NB Onramp	3	Excellent	Excellent sightlines to the gore area.
2146	278.4	I-15 SB Pioneer Crossing Onramp	3	Good	Good sightlines to a large gore area.
2147	278.7	I-15 NB Pioneer Crossing Onramp	3	Excellent	Excellent sightlines to the gore area.
2148	279.7	Lehi Main St & I-15 SB Onramp	3	Good	Good sightlines to a large gore area.
2149	279.9	Lehi Main St & I-15 NB Onramp	3	Excellent	Excellent sightlines to the gore area.
2150	282.2	2100 N & EB MVC I-15 SB Onramp	3	Poor	Extremely curved alignment causes obstructed sightlines.
2151	282.5	2100 N to I-15 NB Onramp	3	Excellent	Excellent sightlines to the ramp entrance.
2155	283.7	Triumph Blvd & I-15 NB Onramp	3	Good	Good sightlines to a curved alignment.
2156	282.9	SR-92 SB Onramp	3	Poor	Curved alignment; bridge occlusion.
2157	284.2	SR-92 NB Onramp	3	Good	Good sightlines to a curved alignment.
2302	288	14600 S SB Onramp	2	Good	Good sightlines to a curved alignment.

2303	288.5	14600 S NB Onramp	2	Good	Good sightlines to a large gore area.
2306	289.7	Bangerter Hwy SB Onramp	2	Good	Good sightlines to a large gore area.
2307	290	Bangerter Hwy NB Onramp	2	Excellent	Excellent sightlines to the ramp entrance.
2308	291.2	12300 S SB Onramp	2	Good	Good sightlines for > 80% of the ramp area.
2309	291.5	12300 S NB Onramp	2	Good	Good sightlines to a large gore area.
2310	292.4	11400 S SB Onramp	2	Excellent	Excellent sightlines to the ramp entrance.
2311	292.8	11400 S NB Onramp	2	Excellent	Excellent sightlines to the ramp entrance.

RGN: Region

Of the 37 metered ramps, 30 were rated as being “Good” or “Excellent” for a computer vision application. In the cases of a “Good” rating, many of these ramps had larger gore areas at the ramp entrance. Gore areas occur when two sub-ramps enter from each direction of the surface street feeding the ramp. These gore areas typically define the limit of a sightline and, hence, of a computer visioning system. The larger the entrance gore, the larger the section of a ramp that is undetectable.

7.0 <u>REFERENCES</u>

Advantages and Disadvantages of LiDAR – LiDAR and RADAR Information. Retrieved October 11, 2022, from https://lidarradar.com/info/advantages-and-disadvantages-of-lidar

Advantages and disadvantages of Piezoresistive strain gauge sensors - TM Automation Instruments Co., Ltd. Retrieved October 11, 2022, from https://www.tmvenus.com/Advantages-and-disadvantages-of-Piezoresistive-strain-gauge-sensors-id3876792.html

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00444-8

Baek and, J., & Sohn, K. (2016). Deep-Learning Architectures to Forecast Bus Ridership at the Stop and Stop-To-Stop Levels for Dense and Crowded Bus Networks. *Applied Artificial Intelligence*, *30*(9), 861–885. https://doi.org/10.1080/08839514.2016.1277291

Betke, M., Haritaoglu, E., & Davis, L. S. (2000). Machine Vision and Applications Real-time multiple vehicle detection and tracking from a moving vehicle. *Machine Vision and Applications*.

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple Online and Realtime Tracking. *Proceedings - International Conference on Image Processing, ICIP*, *2016-August*, 3464–3468. https://doi.org/10.1109/ICIP.2016.7533003

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. http://arxiv.org/abs/2004.10934

Brilakis, I., Fathi, H., & Rashidi, A. (2011). Progressive 3D reconstruction of infrastructure with videogrammetry. *Automation in Construction*, *20*(7), 884–895. https://doi.org/10.1016/j.autcon.2011.03.005

Cai, Z., & Vasconcelos, N. (2017). *Cascade R-CNN: Delving into High Quality Object Detection*. http://arxiv.org/abs/1712.00726

Chen, C., Xiang, H., Qiu, T., Wang, C., Zhou, Y., & Chang, V. (2018). A rear-end collision prediction scheme based on deep learning in the Internet of Vehicles. *Journal of Parallel and Distributed Computing*, *117*, 192–204. https://doi.org/10.1016/j.jpdc.2017.08.014

Chen, Q., Song, X., Yamada, H., & Shibasaki, R. (2022). *Learning Deep Representation from Big and Heterogeneous Data for Traffic Accident Inference*. www.aaai.org

Cheng, Q., Yang, L., Wei, W., & Zhiyuan, L. (2017). Analysis and Forecasting of the Day-to-day Travel Demand Variations for Large- scale Transportation Networks: A Deep Learning Approach. *TRB 2017 Transportation Analytics Contest*, *March*, 1–18. https://doi.org/10.13140/RG.2.2.12753.53604

Chu, P., Wang, J., You, Q., Ling, H., & Liu, Z. (2021). *TransMOT: Spatial-Temporal Graph Transformer for Multiple Object Tracking*. https://doi.org/10.48550/arxiv.2104.00194

Cui, Y., Xu, H., Wu, J., Sun, Y., & Zhao, J. (2019). Automatic Vehicle Tracking with Roadside LiDAR Data for the Connected-Vehicles System. *IEEE Intelligent Systems*. https://doi.org/10.1109/MIS.2019.2918115

Dabiri, S., Marković, N., Heaslip, K., & Reddy, C. K. (2020). A deep convolutional neural network based approach for vehicle classification using large-scale GPS trajectory data. *Transportation Research Part C: Emerging Technologies*, *116*(November 2018), 102644. https://doi.org/10.1016/j.trc.2020.102644

Dai, F., Rashidi, A., Brilakis, I., & Vela, P. (2013). Comparison of Image-Based and Time-of-Flight-Based Technologies for Three-Dimensional Reconstruction of Infrastructure. *Journal of Construction Engineering and Management*, *139*(1), 69–79. https://doi.org/10.1061/(asce)co.1943-7862.0000565

Dong, W., Li, J., Yao, R., Li, C., Yuan, T., & Wang, L. (2016). *Characterizing Driving Styles with Deep Learning*. http://arxiv.org/abs/1607.03611

ECSTUFF4U for Electronics Engineer: Advantages and disadvantages of piezoelectric transducer. Retrieved October 11, 2022, from https://www.ecstuff4u.com/2019/07/advantages-and-disadvantages-of-piezoelectric-transducer.html

Elena Mimbela Project Manager, L. Y., Klein, L. A., Klein, P., & Herrera, S. (2007). *Summary Of Vehicle Detection And Surveillance Technologies Used In Intelligent Transportation Systems Submitted To: Federal Highway Administration's (FHWA) Intelligent Transportation Systems Program Office PREPARED BY*. http://www.nmsu.edu/~traffic/

Farhadmanesh, M., Cross, C., Mashhadi, A. H., Rashidi, A., & Wempen, J. (2021). Highway asset and pavement condition management using mobile photogrammetry. In

Transportation Research Record (Vol. 2675, Issue 9, pp. 296–307). SAGE Publications Ltd. https://doi.org/10.1177/03611981211001855

Farhadmanesh, M., Rashidi, A., & Markovic, N. (2022). General Aviation Aircraft Identification at Non-Towered Airports Using a Two-Step Computer Vision-Based Approach. *IEEE Access*, *10*, 48778–48791. https://doi.org/10.1109/ACCESS.2022.3172963

FIELD TEST OF MONITORING OF URBAN VEHICLE OPERATIONS USING NON-INTRUSIVE TECHNOLOGIES Final Report S O N I C. (1997). www.adobe.com

Genders, W., & Razavi, S. (2016). *Using a Deep Reinforcement Learning Agent for Traffic Signal Control*. 1–9. http://arxiv.org/abs/1611.01142

Girshick, R. (2015). *Fast R-CNN*. http://arxiv.org/abs/1504.08083

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2013). *Rich feature hierarchies for accurate object detection and semantic segmentation*. http://arxiv.org/abs/1311.2524

Hassandokht Mashhadi, A., Rashidi, A., & Markovic, N. (2023). *AUTOMATED SAFETY ASSESSMENT OF RURAL ROADWAYS USING COMPUTER VISION*. www.udot.utah.gov/go/research

Huval, B., Wang, T., Tandon, S., Kiske, J., Song, W., Pazhayampallil, J., Andriluka, M., Rajpurkar, P., Migimatsu, T., Cheng-Yue, R., Mujica, F., Coates, A., & Ng, A. Y. (2015). *An Empirical Evaluation of Deep Learning on Highway Driving*. http://arxiv.org/abs/1504.01716

Jehad, A. E., & Rahmat, R. A. O. K. (2017). DEVELOPING AND VALIDATING A REAL TIME VIDEO BASED TRAFFIC COUNTING AND CLASSIFICATION. In *Journal of Engineering Science and Technology* (Vol. 12, Issue 12).

Koller, D., Weber, J., & Malik, J. (1994). Robust multiple car tracking with occlusion reasoning. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *800 LNCS*, 189–196. https://doi.org/10.1007/3-540-57956-7_22/COVER

Kumar, T. (2020). *INTERNATIONAL RESEARCH JOURNAL OF ENGINEERING AND TECHNOLOGY (IRJET) Comparative Study of Existing Tracking Algorithms*.

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). *Focal Loss for Dense Object Detection*. http://arxiv.org/abs/1708.02002

Lippi, M., Bertini, M., & Frasconi, P. (2013). Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning. *IEEE Transactions on Intelligent Transportation Systems*, *14*(2), 871–882. https://doi.org/10.1109/TITS.2013.2247040

Liu, L., & Chen, R. C. (2017). A Mrt daily passenger flow prediction model with different combinations of influential factors. *Proceedings - 31st IEEE International Conference on Advanced Information Networking and Applications Workshops, WAINA 2017*, 601–605. https://doi.org/10.1109/WAINA.2017.19

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2015). *SSD: Single Shot MultiBox Detector*. https://doi.org/10.1007/978-3-319-46448-0_2

Ma, X., Yu, H., Wang, Y., & Wang, Y. (2015). Large-scale transportation network congestion evolution prediction using deep learning theory. *PLoS ONE*, *10*(3), 1–17. https://doi.org/10.1371/journal.pone.0119044

Mohammadi, P., Rashidi, A., Malekzadeh, M., & Tiwari, S. (2023). Evaluating various machine learning algorithms for automated inspection of culverts. *Engineering Analysis with Boundary Elements*, *148*, 366–375. https://doi.org/10.1016/j.enganabound.2023.01.007

Nguyen, H., Kieu, L. M., Wen, T., & Cai, C. (2018). Deep learning methods in transportation domain: A review. *IET Intelligent Transport Systems*, *12*(9), 998–1004. https://doi.org/10.1049/iet-its.2018.0064

Partl, M. N., Raab, C., & Arraigada, M. (2015). Innovative asphalt research using accelerated pavement testing. *Journal of Marine Science and Technology (Taiwan)*, *23*(3), 269–280. https://doi.org/10.6119/JMST-014-0326-1

Pellegrini, S., Ess, A., Schindler, K., & Van Gool, L. (2009). You'll never walk alone: Modeling social behavior for multi-target tracking. *Proceedings of the IEEE International Conference on Computer Vision*, 261–268. https://doi.org/10.1109/ICCV.2009.5459260

Peng, X., & Shan, J. (2021). Detection and tracking of pedestrians using doppler lidar. *Remote Sensing*, *13*(15). https://doi.org/10.3390/rs13152952

Ramp Metering: A Proven, Cost-Effective Operational Strategy - A Primer: 1. Overview of Ramp Metering. Retrieved March 25, 2023, from https://ops.fhwa.dot.gov/publications/fhwahop14020/sec1.htm

Rashidi, A., & Karan, E. (2018). *Video to BrIM*: Automated 3D As-Built Documentation of Bridges. *ASCE Journal of Performance of Constructed Facilities,* 32(3)
https://doi.org/10.1061/(ASCE)CF.1943-5509.0001163

Rashidi, A., Fathi, H., & Brilakis, I. (2011). Innovative stereo vision-based approach to generate dense depth map of transportation infrastructure. *Transportation Research Record*, *2215*, 93–99. https://doi.org/10.3141/2215-10

Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. http://arxiv.org/abs/1506.01497

Sharma, A., Bullock, D. M., Bonneson, J. A., Sharma, A.;, & Bullock, D. M. ; (2007). *Input-Output and Hybrid Techniques for Real-Time Prediction of Input-Output and Hybrid Techniques for Real-Time Prediction of Delay and Maximum Queue Length at Signalized Intersections Delay and Maximum Queue Length at Signalized Intersections Part of the Civil Engineering Commons "Input-Output and Hybrid Techniques for Real-Time Prediction of Delay and Maximum Queue Length at Signalized Intersections" (2007). Civil Engineering Faculty Publications*. https://doi.org/10.3141/2035-08

Srivastava, S., Divekar, A. V., Anilkumar, C., Naik, I., Kulkarni, V., & Pattabiraman, V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00434-w

Umair, M., Farooq, M. U., Raza, R. H., Chen, Q., & Abdulhai, B. (2021). Efficient video-based vehicle queue length estimation using computer vision and deep learning for an urban traffic scenario. *Processes*, *9*(10). https://doi.org/10.3390/pr9101786

Strain Gauge: Principle, Types, Features and Applications | Encardio Rite Retrieved October 11, 2022, from https://www.encardio.com/blog/strain-gauge-principle-types-features-and-applications/

Wilbur, A. (2006). *NOTICE QUALITY ASSURANCE STATEMENT*. http://www.tfhrc.gov

Wojke, N., Bewley, A., & Paulus, D. (2017). Simple Online and Realtime Tracking with a Deep Association Metric. *Proceedings - International Conference on Image Processing, ICIP*, *2017-September*, 3645–3649. https://doi.org/10.48550/arxiv.1703.07402

Wu, J., Xu, H., Zhang, Y., Tian, Y., & Song, X. (2020). Real-time queue length detection with roadside lidar data. *Sensors (Switzerland)*, *20*(8). https://doi.org/10.3390/s20082342

Yu, Y., Yu, M., Yan, G., & Zhai, Y. (2011). Length-based vehicle classification in multi-lane traffic flow. *Transactions of Tianjin University*, *17*(5), 362–368. https://doi.org/10.1007/s12209-011-1598-0

Zhang, W., Suo, C., & Wang, Q. (2008). A Novel Sensor System for Measuring Wheel Loads of Vehicles on Highways. *Sensors*, *8*, 7671–7689. https://doi.org/10.3390/s8127671

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., & Wang, X. (2022). *ByteTrack: Multi-Object Tracking by Associating Every Detection Box*. Retrieved October 12, 2022, from https://github.com/ifzhang/ByteTrack.

Zhang, Y., Wang, C., Wang, X., Zeng, W., & Liu, W. (2020). FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking. *International Journal of Computer Vision*, *129*(11), 3069–3087. https://doi.org/10.1007/s11263-021-01513-4

Zhou, Q., Li, B., Kuang, Z., Xie, D., Tong, G., Hu, L., & Yuan, X. (2013). A quarter-car vehicle model based feature for wheeled and tracked vehicles classification. *Journal of Sound and Vibration*, *332*(26), 7279–7289. https://doi.org/10.1016/j.jsv.2013.08.042

Zhou, Y., Nejati, H., Do, T.-T., Cheung, N.-M., & Cheah, L. (2016). *Image-based Vehicle Analysis using Deep Neural Network: A Systematic Study*. http://arxiv.org/abs/1601.01145

Zhu, X., Li, J., Liu, Z., & Yang, F. (2017). Location deployment of depots and resource relocation for connected car-sharing systems through mobile edge computing. *International Journal of Distributed Sensor Networks*, *13*(6), 1–15. https://doi.org/10.1177/1550147717711621