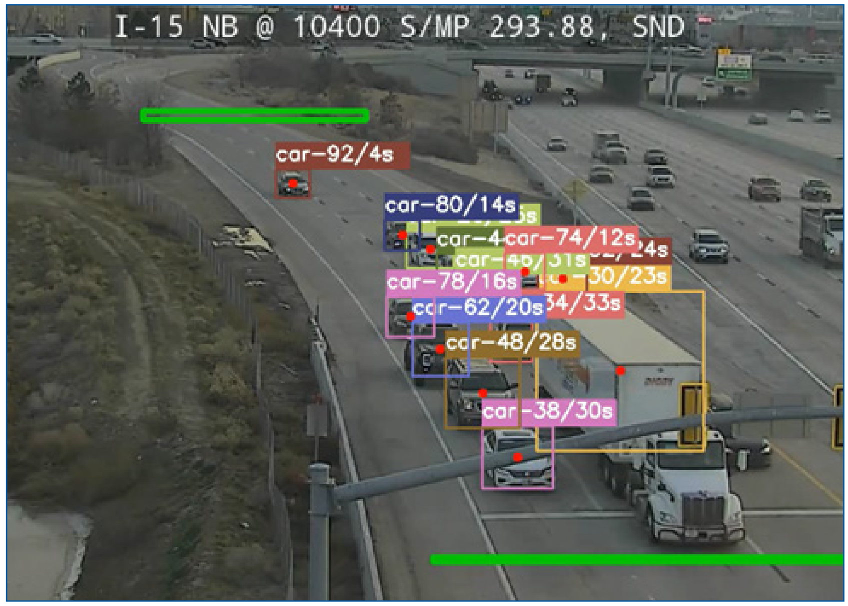
MOUNTAIN-PLAINS CONSORTIUM

MPC 23-507 | S. Tiwari, A. Rashidi, and N. Markovic

EVALUATING DIFFERENT METHODS FOR ESTIMATING QUEUE LENGTH ON ACCESS RAMPS



A University Transportation Center sponsored by the U.S. Department of Transportation serving the Mountain-Plains Region. Consortium members:

Colorado State University
North Dakota State University
South Dakota State University
University of Colorado Denver
University of Denver
University of Utah
Utah State University
University of Wyoming

MOUNTAIN-PLAINS CONSORTIUM REGION 8

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
MPC-699		

4. Title and Subtitle Evaluating Different Methods for Estimating Queue Length on Access Ramps	5. Report Date November 2023 6. Performing Organization Code
7. Author(s) Sushant Tiwari Dr. Abbas Rashidi Dr. Nikola Markovic	8. Performing Organization Report No. MPC 23-507
9. Performing Organization Name and Address University of Utah 110 Central Campus Dr. #2000b Salt Lake City, U 84112	10. Work Unit No. (TRAIS) 11. Contract or Grant No.
12. Sponsoring Agency Name and Address Mountain-Plains Consortium North Dakota State University PO Box 6050, Fargo, ND 58108	13. Type of Report and Period Covered Final Report 14. Sponsoring Agency Code

15. Supplementary Notes

Supported by a grant from the US DOT, University Transportation Centers Program

16. Abstract

Understanding the queue length and queuing time of ramp is important for transportation agencies to manage and operate the ramps with optimum performance. Since these data are collected with conventional sensors system such as coil, which are prone to error specially during the congestion. The increased deployment of cameras and recent advancements in artificial intelligence such as deep learning and computer vision gives an opportunity to employ traffic surveillance camera videos for ramp management. In this study, we employed the four location surveillance cameras videos to develop and evaluate the framework developed using the object detection and tracking algorithms. This framework uses the existing camera videos as input to the framework and determine the queueing parameters of highway on ramps such as queue length and queuing time which provides an important information to freeway management team to optimize signal timing. Additionally, this study provides a detailed implementation plan for computer vision and optimum location of the camera installation and hardware requirements.

17. Key Word	18. Distribution Statement
computer vision, data collection, ramp metering, ramps, traffic queuing	Public distribution

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	47	n/a

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

Evaluating Different Methods for Estimating Queue Length on Access Ramps

Sushant Tiwari

Dr. Abbas Rashidi

Dr. Nikola Markovic'

Department of Civil & Environmental Engineering
The University of Utah
Salt Lake City, Utah

November 2023

Acknowledgments

This project is funded by the Utah Department of Transportation (UDOT) and Mountain-Plains Consortium (MPC). The opinions and findings of the authors do not necessarily reflect the view and opinions of UDOT and MPC.

Disclaimer

ABSTRACT

Understanding ramp queue length and queuing time is important for transportation agencies to manage and operate the ramps with optimum performance. Since these data are collected with conventional sensor systems such as coils, they are prone to error, especially during traffic congestion. The increased deployment of cameras and recent advancements in artificial intelligence, such as deep learning and computer vision, provides an opportunity to employ traffic surveillance camera videos for ramp management. This study employed four location surveillance video cameras to develop and evaluate the framework developed using the object detection and tracking algorithms. This framework uses existing video cameras as input to the framework and determines the queueing parameters of highways on ramps, such as queue length and queuing time, which provide important information to freeway management teams to optimize signal timing. Additionally, this study provides a detailed implementation plan for computer vision and optimum location of the camera installation and hardware requirements.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

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 Operation 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

Extensive utilization of cameras as video recording devices offers a significant opportunity to employ artificial intelligence, encompassing both machine learning and deep learning, for the analysis of transportation issues. This research leverages real-time traffic camera video feeds through deep learning and a computer vision framework to assess on-ramp performance. Furthermore, the calculated metrics serve as valuable inputs for the traffic management team's decision-making processes, particularly with regard to ramp signal metering. Utah employs ramp metering strategies during peak hours to ensure the smooth flow of mainline traffic. This study focuses on the development and evaluation of the model using data from four specific on-ramps located on I-15: 10400s NB, 500S NB, 11400S NB, and Beck Street NB.

Data for video surveillance were acquired in collaboration with UDOT's Traffic Operations Center (TOC), wherein four specific locations were carefully selected to meet criteria related to camera orientation, height, video quality, and field of view. Video footage was captured at these chosen locations to facilitate subsequent visual analysis. The field of view was meticulously configured to encompass the maximum possible range of both the ramps and the vehicles using them. In order to enhance the system's resilience against variations in external lighting and seasonal weather changes, video recordings were conducted under diverse lighting and weather conditions. The "You Only Look Once" (YOLO) base model was subsequently retrained to enhance its accuracy across all these conditions, thereby rendering this technique more versatile and precise for real-world applications.

To enhance the capabilities of the computer vision system, the research team developed an advanced framework rooted in deep learning and computer vision technologies. Upon acquiring the requisite data, the team subjected the collected videos to a meticulous preprocessing stage before seamlessly integrating them with an object detection and tracking framework. This framework leveraged a fusion of cutting-edge algorithms, namely YOLOv4 for state-of-the-art object detection and DeepSORT for state-of-the-art object tracking, enabling the system to not only detect the class of objects but also accurately quantify the number of vehicles and calculate the total time each object spent while traversing through the ramps. The comprehensive report explains in detail the working mechanisms of these algorithms. The obtained results, which demonstrate promising levels of accuracy, underscore the efficacy of this approach, given that the captured videos' quality is maintained.

1. INTRODUCTION

1.1 Introduction

Ramp metering, initially introduced on the Eisenhower Expressway in Chicago during the 1960s, has subsequently been adopted in cities like Detroit, Los Angeles, and Minneapolis/St. Paul. Its implementation has resulted in several notable benefits, including improved traffic flow on the mainline, reduced collisions, and decreased emissions. Due to its multifaceted advantages, the practice of implementing ramp metering during peak hours has gained widespread acceptance both in the United States and around the world. Minneapolis, for instance, has documented the prevention of 1,160 tons of emissions into the environment through the successful implementation of ramp metering. Additionally, it has proven efficient in enhancing safety by facilitating smoother transitions for vehicles merging from the ramp onto the mainline traffic. Furthermore, ramp metering contributes to reducing overall freeway travel times by preventing the formation of traffic 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 are 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 video from already existing traffic cameras – provides an economical and dynamic data collection framework. Image processing can extract different types of data, like counts and speeds in various zones, without the need for additional hardware or specialized devices 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 – have 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 Current Gaps in Study

Efficient traffic management of ramps using ramp metering is very crucial for maintaining smooth highway operation that helps minimize traffic congestion and enhance safety by eliminating shockwaves. Moreover, ramp metering aids in reducing environmental pollution by lowering fuel consumption and emissions while also preventing bottlenecks at ramp and highway intersections (Wilbur, 2006). To gain insights into traffic conditions, various sensors are employed; these are categorized as in-roadway and over-roadway sensors. For real-time and efficient signal coordination, induction loops and similar sensors are commonly used. However, these installations can be costly, time-consuming, and disruptive to traffic. An alternative approach involves the use of surveillance cameras, particularly those utilized for traffic incident management, which can help reduce the expenses associated with sensors and signaling tools needed for intersections or ramps. Deep learning and computer vision algorithms, such as convolutional neural networks (CNN), can be leveraged to extract real-time traffic data from video footage recorded by these cameras (Umair et al., 2021).

Analyzing queue length and queue delay is of utmost importance when it comes to formulating effective ramp signal metering strategies. A well-crafted ramp metering signal is crucial for ensuring the smooth flow of traffic on the mainline and preventing any unwanted spillover from the ramps. Estimating queue length and queue delay can be efficiently achieved through the application of image processing techniques, particularly by leveraging object detection and tracking algorithms. Object detection plays a pivotal role in identifying not only the presence of vehicles but also their specific classification and spatial positioning within each frame. By integrating object detection with tracking algorithms, one can continually monitor and record the delays experienced by individual vehicles captured within the frames. This combined approach yields valuable information such as average delay experienced by vehicles, typical queue length, and even the classification of vehicles in the queue. An additional advantage of employing image processing in this context is its minimal disruption to road and mainline services during both installation and maintenance. Recognizing the immense potential of this innovative technology, the Utah Department of Transportation (UDOT) Traffic Management Division is actively exploring the utilization of an existing camera system to facilitate ramp metering, offering a cost-effective solution with a high degree of accuracy.

1.3 Framework objective

The principal goals of this research endeavor are centered on assisting UDOT in ascertaining the feasibility of leveraging existing camera footage to precisely assess ramp queues and delays. A secondary objective of this study is to furnish comprehensive design guidelines pertaining to the optimal mounting height and orientation of cameras, as well as the technical specifications of cameras to achieve peak performance. The initial task involves conducting an extensive review of the literature on machine learning and computer vision techniques related to object detection and tracking. Additionally, we explored existing literature on queue length estimation employing statistical methodologies and queue estimation techniques. Subsequently, we undertook the task of data collection by identifying potential ramp locations with the aid of preexisting cameras, considering factors such as camera height, orientation, and technical specifications. The collaborative efforts of our research team and the UDOT traffic operation team led us to identify a minimum of four locations for this research initiative. The recorded camera video data undergo a comprehensive analysis using the object detection and tracking framework that has been developed as part of this research.

Following the formulation of the framework and the development of algorithms, the recorded videos undergo preprocessing and subsequent data extraction. In addition to video-based tracking, we will employ queuing methods to further validate the results obtained through video processing. Furthermore, the framework will systematically evaluate the accuracy and effectiveness of the system in measuring delay and queue length. To validate our findings, manual ground truth data will be extracted from randomly selected small samples of video data. The outcomes derived from the video processing techniques will be compared to these manual ground truth data. Ultimately, our team will provide recommendations regarding the optimal frame size, tools for measuring ramp performance, object detection, and tracking algorithms based on an evaluation of their ability to accurately determine queue length and queue delay. Additionally, we will delineate the camera specifications and computing devices necessary for the deployment of this system.

2. BACKGROUND

This segment provides an overview of the merits and drawbacks associated with current sensor technologies while also elucidating the application of contemporary methods like deep learning within the realm of transportation. Furthermore, it delineates the process of extracting performance parameters for ramps through computer vision and details upon the analysis of data using both vision-based techniques and queuing methods in the context of ramp metering strategies. Vehicle detection technologies can be broadly classified into the following categories.

2.1 In-roadway Sensors

In-roadway sensors, which are integrated within the road surface, are designed to gather traffic-related data. The induction loop system serves as a traffic sensor employed for monitoring the presence, passage, count, and occupancy of vehicles on roadways. It comprises a lead-in cable wire loop with a signal frequency range spanning from 10KHz to 50KHz, transmitting signals to the controller cabinet's electronics unit via a pull box as a result of the oscillation frequency increase brought about by a decrease in inductance when vehicles come to a halt or traverse the loop system. This methodology is well-established in practical applications, making it suitable for the measurement of fundamental traffic parameters like headway, volume, vehicle presence, and spacing between vehicles. The specifics of the induction loop installation system can be observed in Figure 2.1. Nonetheless, the installation and maintenance of these loops often result in traffic disruptions, necessitating reinstallation during road and utility repairs, thereby augmenting the loop's life-cycle cost. Additionally, the stress exerted by vehicles and temperature fluctuations can also contribute to potential loop failures (Wilbur, 2006).

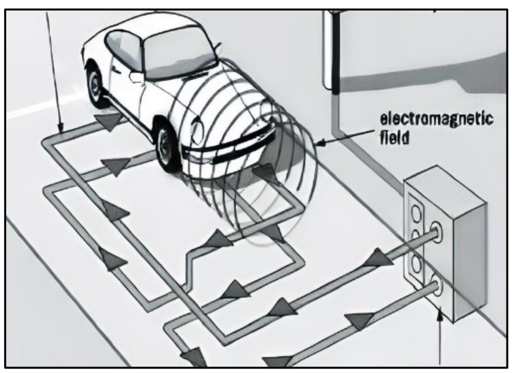


Figure 2.1 Inductive Loop Systems

Magnetometers function by sensing disturbances in the Earth's magnetic field caused by metallic objects, forming the fundamental principle behind their operational mechanism. These devices use magnetic anomalies to identify the presence of vehicles within their detection zone, as outlined in Wilbur's research (Wilbur, 2006). Magnetometers are capable of detecting fluctuations in the magnetic field, enabling the measurement of various parameters such as vehicle counts, speed, length, lane occupancy, daily and annual average traffic, road surface temperature, and wet-dry conditions (Elena Mimbela et al., 2007). It is worth noting that magnetic detectors exhibit a higher resilience to pavement stress compared with induction loops, albeit they necessitate pavement cuts and may cause some traffic disruptions. Figure 2.2 below illustrates the operational principles of magnetic sensors.

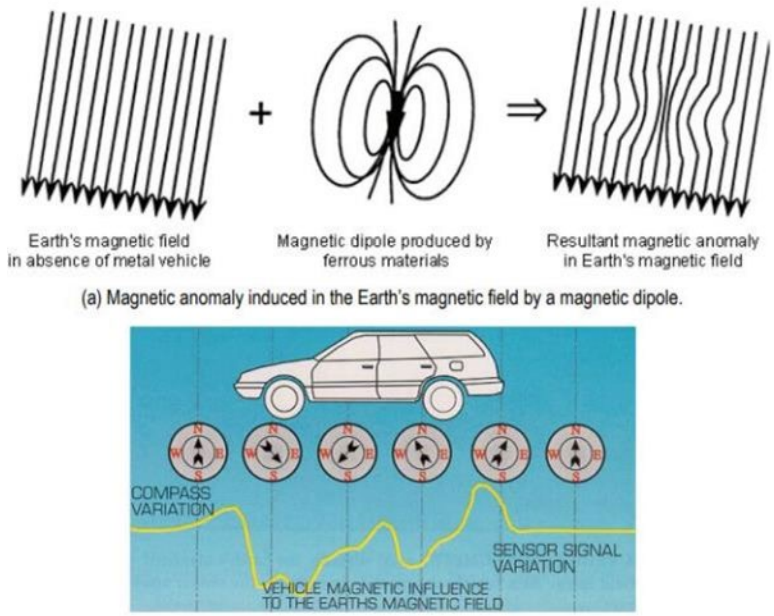


Figure 2.2 Magnetic Sensors
Source: Traffic Detector Handbook, Third Edition – Vol I

The operation of this system closely resembles that of an induction loop system. It has largely replaced induction loops in areas where it is not feasible to cut into the pavement, such as on bridge decks. Nevertheless, it does require drilling beneath the pavement. It exhibits resilience to harsh environmental conditions like rain, snow, and fog and is less susceptible to damage from the rigors of traffic. However, without the use of specific sensors and software, it cannot identify stationary vehicles.

Piezoelectric materials function as sensors that convert mechanical energy, such as vibrations, into electrical energy. They are embedded into each lane for traffic counting and axle spacing determination and covered with epoxy resin to be flush with the road's surface. When a vehicle activates two of these sensors, the system calculates the vehicle's speed and axle spacing based on the measured voltage, which is proportional to the vehicle's weight. These sensors are commonly used in weigh-in-motion (WIM) systems. Signals from these sensors are collected by a roadside junction box, providing information about vehicle count and type in digital format. Despite their small and convenient size, piezoelectric sensors are sensitive to temperature and vulnerable to water infiltration, limiting their use to measuring dynamic motions exclusively (ECSTUFF4U for Electronics Engineer: Advantages and Disadvantages of Piezoelectric Transducer).

Strain gauges categorize vehicles by analyzing the dynamic strain response patterns produced by a gauge installed in the road surface. They offer precise measurements, minimal maintenance requirements, and extended operational durability. Nevertheless, their accuracy can be influenced by temperature variations. Securing them firmly to the pavement throughout their life span can, however, be a challenging task (W. Zhang et al., 2008), (Advantages and Disadvantages of Piezoresistive Strain Gauge Sensors – TM Automation Instruments Co.), (Strain Gauge: Principle, Types, Features and Applications).

Seismic sensors can detect vibrations from moving vehicles, and when multiple sensors collaborate they can collect data to pinpoint and classify various vehicle types. Zhou employs the short-time power spectrum density technique to analyze seismic signals generated by vehicles. While seismic sensors offer a broad detection range, they need thorough calibration before deployment (Q. Zhou et al., 2013).

Pneumatic tubes transmit compressed air through the pipe when a vehicle passes over them. The compressed air actuates an air switch, generating an electrical signal for subsequent software processing. These devices are frequently employed for short-term traffic monitoring and classifying vehicles based on axle count and spacing. They are known for their quick installation, minimal energy consumption, and ease of maintenance. However, under high traffic volumes, axle counting may become less precise. These sensors are temperature-sensitive and susceptible to damage from vehicle-induced wear and tear (Elena Mimbela et al., 2007).

2.2 Over-roadway Sensors

Sensors located above the road pavement or at a certain distance from the pavement are classified as over-roadway sensors. Microwave radio detection and ranging (radar) transmits electromagnetic energy of almost 10GHz for traffic management, as shown in Figure 2.3. 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 2.3 Microwave Radar
Source: (Elena Mimbela 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 SONIC, 1997).

Active and passive infrared technologies offer versatile applications for monitoring approaching and departing vehicles in both top-mounted and side-mounted configurations. These methods serve purposes such as signal control, speed detection, volume measurement of vehicles and pedestrians, and vehicle classification. They operate on the principle of converting emitted energy into electrical signals, which can be further processed to yield results. Active radar, for example, transmits its own energy and receives reflected energy, enabling measurements of distance, speed, queuing, volume, and vehicle classification. It can even distinguish up to 11 vehicle types on toll roads and can be mounted on road surfaces (Elena Mimbela et al., 2007).

On the other hand, passive radar does not emit energy itself but rather receives energy transmitted by objects like roads and vehicles. It excels at detecting volume, lane occupancy, and passage, relying on the graybody emission from these objects. When a vehicle enters the road, it generates a signal due to emissivity differences between the road and vehicles. This approach is also weather resistant when carefully calibrated.

Infrared sensors emit multiple beams to ensure accurate traffic parameter measurements, as shown in Figure 2.4. Side-mounted models can even support multi-lane detection systems. However, it is essential to note that sunlight glint can introduce errors, and inclement weather might disperse energy, potentially affecting signal reception (Field Test of Monitoring of Urban Vehicle Operations Using Non-Intrusive Technologies Final Report SONIC, 1997).

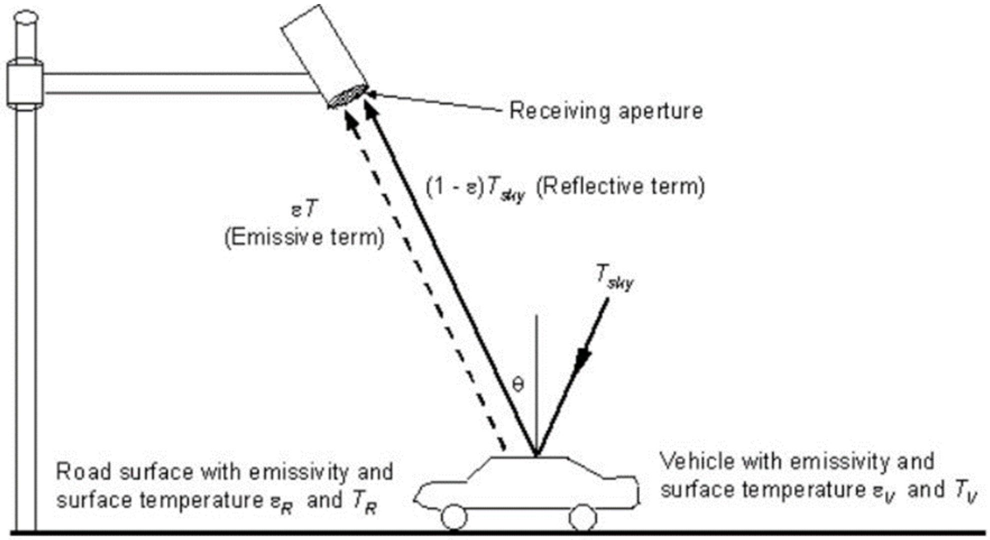


Figure 2.4 Infrared Sensors

Ultrasonic waves are utilized to generate sound waves within the 25-50 kHz frequency range, which falls outside the range of human hearing. These waves primarily employ pulse waveforms to collect data regarding vehicle count, presence, and occupancy status. Figure 2.5 shows the common configurations for these ultrasonic systems, which are overhead mounts and horizontal mounts.

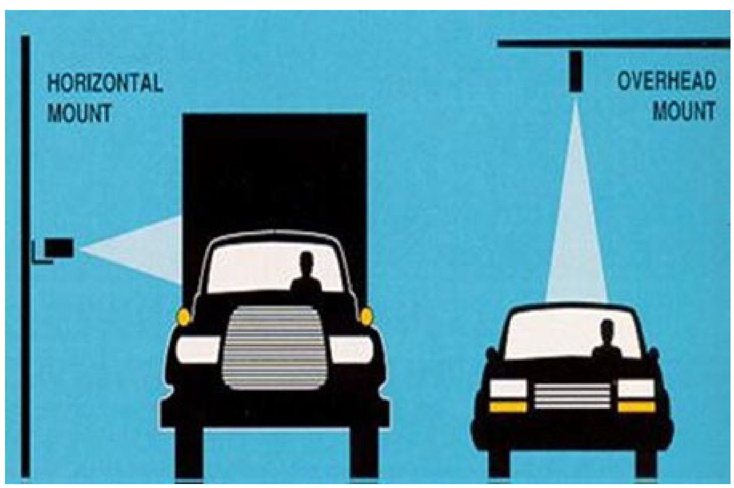


Figure 2.5 Ultrasonic Waves Sensors

Acoustic (another kind of sensor) waves relies on the capture of vehicular movement by recording the sound generated when a vehicle's tires contact the road, in addition to various other ambient sounds. Figure 2.6 shows the mounting configuration for acoustic sensors. It operates based on the Doppler effect, where an oncoming vehicle enhances sound energy, while a departing vehicle diminishes sound energy. Acoustic waves are notably advantageous due to their non-invasive nature and their ability to facilitate multi-lane vehicle detection. However, it is important to note that this technology can be affected by cold weather conditions and vehicles moving slowly in stop-and-go traffic situations.

Figure 2.6 Acoustic Sensors

LiDAR, which is short for light detection and ranging, employs various types of light, including ultrasonic, visible, and infrared rays, to determine the distance to objects. It operates by emitting and then capturing light waves in the form of pulses, and these emitted and received pulses are utilized to calculate the precise distances to objects. LiDAR technology is versatile in its data collection capabilities as it can gather information from the ground, from aerial platforms like aircraft or drones, and even from satellites in space. This technology has found applications in diverse fields, such as vehicle tracking within the development of connected-vehicle systems, as demonstrated by Yuepeng and colleagues (Cui et al., 2019). Additionally, LiDAR has been utilized for pedestrian detection and tracking (Peng & Shan, 2021) and for estimating queue lengths (Wu et al., 2020). One of the key advantages of LiDAR is its ability to function reliably both day and night, and it remains unaffected by extreme weather conditions. However, it is essential to note that LiDAR does come with certain drawbacks, including a relatively high operating cost and limitations in performance during heavy rain or overcast days (Advantages and Disadvantages of LiDAR – LiDAR and RADAR Information).

Image processing algorithms play a pivotal role in analyzing surveillance camera video feeds. They operate by meticulously scrutinizing each frame, employing background subtraction techniques to discern moving objects, with a specific focus on identifying vehicles by detecting unique connected pixel clusters. This approach has demonstrated its superiority over alternative sensor systems that necessitate multiple components. The versatility of video-based traffic flow analysis is further highlighted by its capacity to extract various types of traffic data from the same video sources, all adjustable through algorithmic modifications without demanding costly hardware upgrades. This method extends beyond vehicular data, allowing for the extraction of information related to pedestrians, traffic incident management, and facilitating multi-lane observations without the need for sensor system and hardware alterations. The real potential of these systems emerges when multiple camera feeds are interconnected, offering multiple types of data. Nevertheless, it is worth noting that these camera-based systems face challenges related to lighting conditions, weather fluctuations, obstructions, and camera movements. Encouragingly, recent advancements in deep learning and computer vision approaches hold promise in mitigating these challenges to a considerable extent (Elena Mimbela et al., 2007). Table 2.1 summarizes the comparison of different sensors to extract the different kinds of data needed to analyze traffic, such as speed, count, acceleration, and weight.

Table 2.1 Comparison of Different Sensors for Traffic Data Extraction Capacity

	Sensors	Count	Speed	Occupancy	Classification	MLMD	Acceleration	Direction	Weigt	Axle Configuration	Type & Model	Automatic
In-road Sensors	Inductive Loop	Y	Y	Y	Y	N	Y	Y	N	N	N	Y
	Magnetometer	Y	Y	Y	N	N	Y	Y	N	N	N	Y
	Induction Coil	Y	Y	Y	N	N	Y	Y	N	N	N	Y
	Piezoelectric	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
	Strain Gauge	Y	Y	N	Y	N	Y	Y	N	Y	N	Y
	Seismic	Y	N	N	Y	N	N	N	N	N	N	Y
	Pneumatic Tubes	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
Over-Roadways Sensors	Microwave radar	Y	Y	Y	Y	Y	N	N	N	N	N	Y
	Active Infrared	Y	Y	Y	Y	Y	N	N	N	N	N	Y
	Passive Infrared	Y	Y	Y	N	N	N	N	N	N	N	Y
	Ultrasonic Waves	Y	N	Y	N	N	N	N	N	N	N	Y
	Acoustic Waves	Y	Y	Y	N	Y	Y	Y	N	N	N	Y
	LiDAR	Y	Y	Y	Y	Y	Y	Y	N	N	N	Y
	Image Processing	Y	Y	Y	Y	Y	Y	Y	N	N	Y	Y

2.3 Artificial Intelligence in Transportation Engineering

Deep learning has revolutionized decision-making across various facets of human life, offering a multitude of applications. These deep learning algorithms are characterized by their intricate architecture, comprising multiple layers of non-linear modules that progressively refine and abstract data representations, facilitating the comprehension of complex functions (Lecun et al., 2015).

Within the realm of transportation engineering, deep learning algorithms play a pivotal role in enhancing safety, cost-effectiveness, and overall efficiency of transportation systems. Their application extends to the efficient management of assets and inventories. For instance, Mohammadi et al. (2023) harnessed a combination of machine learning and deep learning techniques to automate the management of culverts. Deep learning was employed to create representations of transportation networks, capturing both spatial and temporal dependencies, thus aiding in the proactive mitigation of congestion by identifying vulnerable links in advance. To forecast traffic parameters and predict the traffic situation at time T+1, their model leveraged historical data from time intervals T (Ma et al., 2015). Furthermore, deep learning finds utility in predicting traffic flow, specifically estimating the number of vehicles in road segments during different future time intervals (Lippi et al., 2013). It also contributes to vehicle classification based on GPS trajectory data (Dabiri et al., 2020). Notably, deep learning has proven to be highly effective in optimizing traffic signal control systems, surpassing traditional methods in terms of efficiency, and simultaneously reducing emissions and congestion (Nguyen et al., 2018). Genders and Razavi (Genders & Razavi, 2016) introduced a deep and cross-network (DCN) deep learning framework, incorporating deep reinforcement learning (DRL) techniques, to develop an adaptive traffic signal control system. This innovative approach has led to significant reductions in cumulative delay, queue length, and travel time in traffic control systems, achieving impressive improvements of 82%, 66%, and 20%, respectively.

Deep learning models are also integral in travel demand modeling for various modes of transportation, including cars (Zhu et al., 2017), mass rapid transit (L. Liu & Chen, 2017), and public transportation (Baek & Sohn, 2016). They facilitate the prediction of passenger demand and the corresponding supply-side infrastructure requirements, ultimately enhancing traffic management by leveraging GPS data, traffic sensors, and CCTV. Cheng et al. (2017) utilized a combination of deep neural networks (DNNs), long short-term memory (LSTM), and feature-level data fusion models to forecast day-to-day travel demand within Florida's transportation network. Moreover, deep learning has proven instrumental in monitoring complex driver behavior using GPS data, a capability with applications in insurance companies and autonomous driving, thereby bolstering road safety (Dong et al., 2016).

In addition to these applications, deep learning also plays a vital role in traffic incident management systems. For instance, Cheng et al. (2017) implemented a deep stack denoise autoencoder to model human mobility and generate a traffic incident risk map, contributing to more effective incident response strategies.

Furthermore, the extension of deep learning employing convolutional neural networks (CNNs) plays a pivotal role in image processing, further strengthening the decision-making process by enhancing the analysis of visual data. In sum, deep learning's multifaceted applications in transportation engineering make it an indispensable tool for shaping safer, more cost-effective, and efficient traffic management strategies.

The integration of deep learning technology with various image processing techniques has significantly improved decision-making processes within the transportation system. This fusion has played a pivotal role in the development of various applications such as driver assistance systems, autonomous driving technology, traffic inventory management, and traffic signal design. Notable examples of these applications include Farhadmanesh et al. (2021) and their utilization of mobile photogrammetry for highway asset and pavement monitoring, as well as Rashidi et al. (2018) and their use of videos and point clouds to automatically create bridge information models from built infrastructure data. Furthermore, the transportation infrastructure has seen a remarkable transformation with the implementation of three-dimensional reconstruction techniques through image and video processing (Brilakis et al., 2011), (Dai et al., 2013) and depth mapping facilitated by stereo vision systems (Rashidi et al., 2011). Additionally, Hassandokht Mashhadi et al., (2023) employed computer vision for providing safety ratings for rural highways.

In the context of determining queue lengths, a two-step approach is essential for comprehensive data extraction. The first step involves object detection, which identifies the spatial location of objects within a frame. Subsequently, object recognition allows us to classify these detected objects into specific categories. Alzubaidi et al. (2021) employed CNNs for vehicle detection in UAV images, while Y. Zhou et al. (2016) utilized DNNs to identify cars, sedans, and vans. Farhadmanesh et al. (2022) successfully identified light aircraft at non-towered airports using image processing tools. Huval et al. (2015) applied CNN to process images in the context of highway driving, enabling the detection of lanes and vehicles at a speed suitable for real-time systems. Jehad & Rahmat (2017) achieved up to 95% accuracy in vehicle detection by employing rapid vehicle detection techniques involving background subtraction, shadow removal, and pixel analysis. Yu et al. (2011) introduced a length-based vehicle classification method with background subtraction and threshold-based segmentation to accurately identify small and large vehicles with high precision. Once these objects are detected, they need to be categorized and tracked using bounding boxes and tracking IDs to determine the total time each vehicle spends in the queue. Table 2.2 provides an overview of prevalent object detection algorithms, with the YOLO approach standing out as the most efficient in terms of both accuracy and speed, as documented by Srivastava et al. (2021). Table 2 enlists the prevailing object detection algorithms.

Table 2.2 Prevailing 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)

Object tracking involves the continuous localization of an object within a video, analyzing each frame to assign a tracking ID to the object and following it through subsequent frames as long as it remains within the frame of interest. Pellegrini et al. (2009) employed object tracking to investigate social behavior among individuals. Koller et al. (1994) used the active contour model to simultaneously track multiple vehicles in road traffic scenarios. Betke et al. (2000) employed a combination of features such as color, edge, and motion to track road boundaries, lane markings, and other vehicles. Object tracking can be categorized into single object tracking (SOT) and multiple object tracking (MOT). MOT is more complex due to challenges like similar appearances, ID-switching problems, and occlusions. The process of object tracking typically encompasses detection, estimation modeling, data association, as well as track creation and deletion.

The detection phase relies on convolutional neural network (CNN) techniques to extract object features and determine the spatial location of the object within the frame, providing data for subsequent classification. The estimation model aims to predict the future position of the tracked object in the next frame, often utilizing the linear constant velocity model. Many estimation models make use of the Kalman filter to forecast the object's future state accurately. Data association is a critical step where the algorithm seeks to establish a correlation between the predicted target location from the estimation model and newly detected objects, typically accomplished through intersection over union (IOU) matching. When a sufficiently strong correlation, surpassing a predefined threshold, is found, the tracking ID is assigned to the new detection.

Track identities are created when an object first enters the frame under analysis and deleted when the object exits the region of interest. In essence, the process of object tracking is a sequence of stages, including detection, estimation modeling, data association, track identity creation, and track identity deletion. Table 2.3 summarizes some of the most prominent tracking algorithms in the field. Notably, Tarushree (Kumar, 2020) validates the superiority of the DeepSORT tracking algorithm when compared with other alternatives.

Table 2.3 Prevailing Object Tracking Algorithms

S. N	Tracking algorithms	References
1	Simple online and realtime 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., 2017a)

3. METHODOLOGY

This section describes the complete methodology of data collection along with the object detection and tracking process utilized in this study. This section will describe the data collection process, object detection, and recognition process, which detects the object in a frame and recognizes the class of detected objects. Tracking will track every recognized object continuously and gather information such as vehicle queuing time. Every process is described in detail below:

3.1 Data Collection

3.1.1 Data Collection for Model Evaluation

Data collection was a collaborative effort between the professional staff at the Utah Department of Transportation's Traffic Operations Center and our research team, in consultation with Technical Advisory Committee (TAC) members. To ensure the optimal capture of video footage depicting ramp operations, several key factors were considered, including the visual system, camera orientation, visibility, and camera height. Four distinct locations were carefully chosen for data collection, as depicted in Figure 3.1 through Figure 3.5. To enhance the model's generalization capabilities, data were gathered under various lighting conditions, spanning morning, daylight, evening, sunset, and nighttime periods. To bolster the computer vision model's robustness, it was subjected to testing under different seasonal conditions, which included the collection of video footage during snowfall, rainy weather, and rainy nighttime conditions. These efforts were undertaken to comprehensively evaluate the model's performance and adaptability.

Figure 3.1 Sample Frame at 10400 South Bound Ramp at I-15 Highway

Figure 3.2 Sample Frame at 500 South Bound Ramp at I-15 Highway

Figure 3.3 Sample Frame at 11400 South Bound Ramp at I-15 Highway

Figure 3.4 Sample Frame at Beck Street South Bound at I-15 Highway

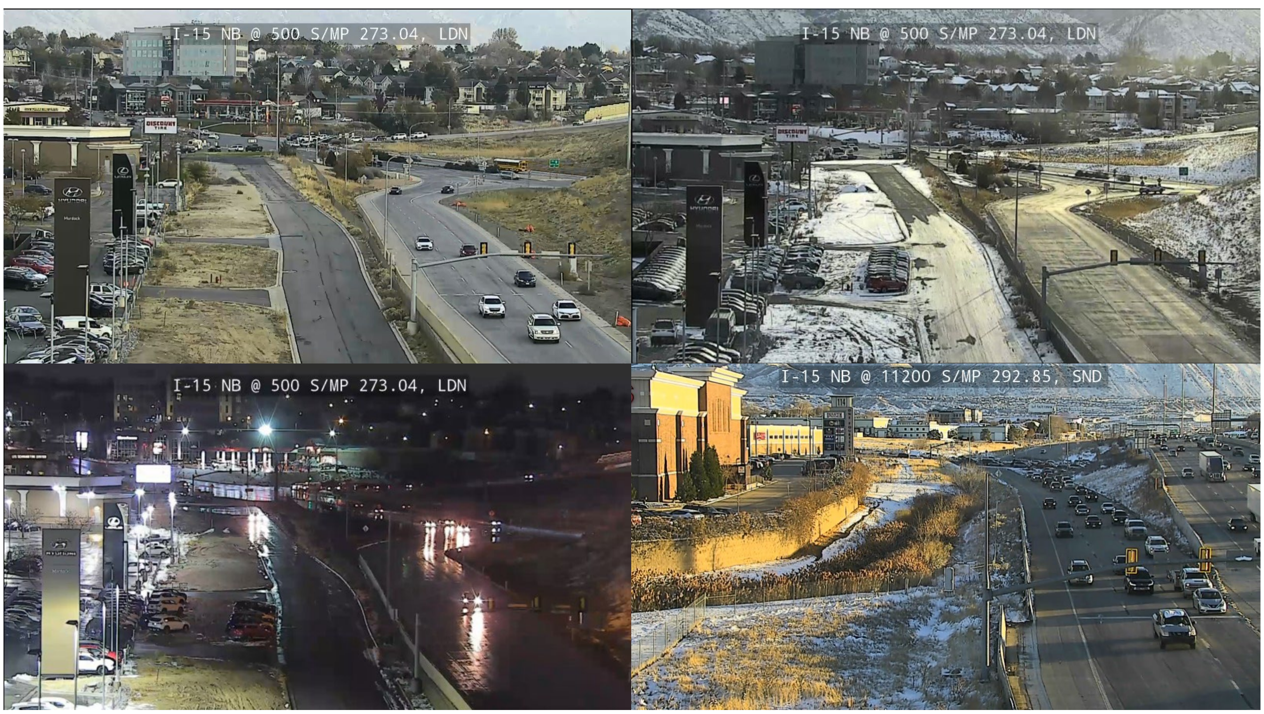


Figure 3.5 Seasonal and Lighting Variation Capture in Framework Evaluation Testing Dataset

3.1.2 Data Collection for Model Retraining

The research team employed the state-of-the-art object detection model known as You Only Look Once (YOLO) for their object detection algorithm. YOLO was initially trained using the COCO dataset, which consists of 80 object classes. However, real-world scenarios often present challenges such as varying lighting conditions and seasonal changes. Additionally, when developing the model, they encountered situations where cars were partially occluded, making object detection more complex. To address these challenges, the team decided to retrain the model using a more relevant dataset.

To enhance the model's robustness, the research team collected over 8,200 images of cars and trucks from open-source platforms, such as Google, under Creative Commons licenses (for academic purposes, not for commercial use). These images were manually labeled to ensure accurate training data. The collected dataset included images captured in diverse lighting and seasonal conditions, such as snow at night and standard daylight.

Special care was taken to include various types of cars, including SUVs and sedans, in the training dataset. Figure 3.6 provides examples of manually labeled frames that represent challenging scenarios encountered during model retraining.

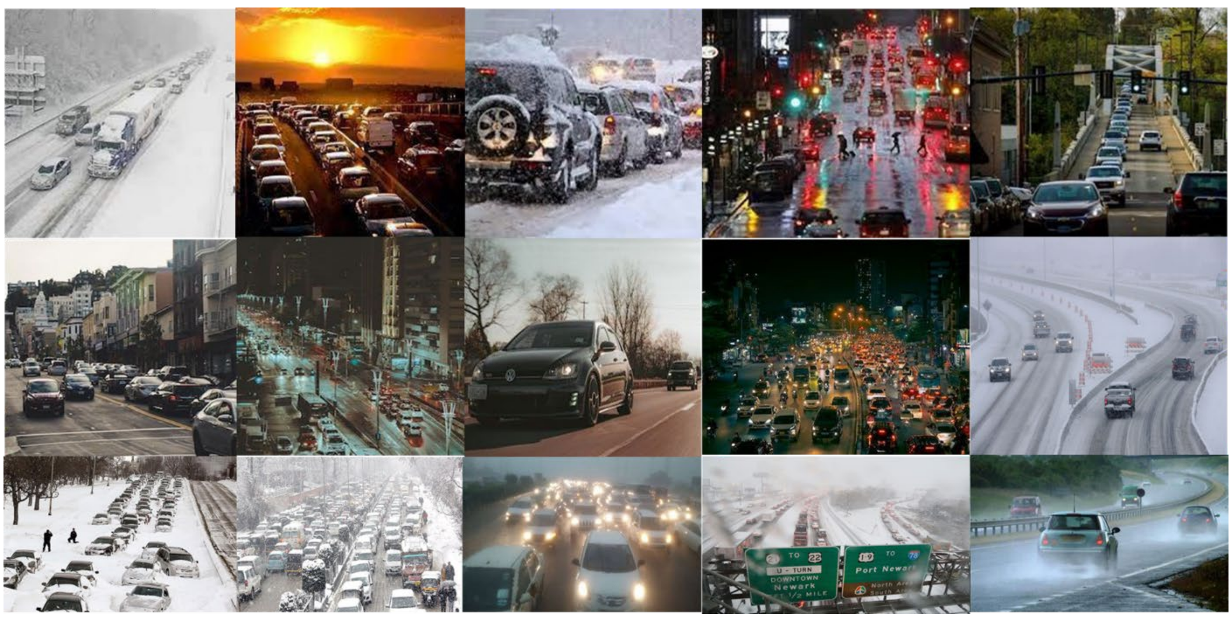


Figure 3.6 Lighting and Seasonal Variation in Training Dataset

3.2 Methodology

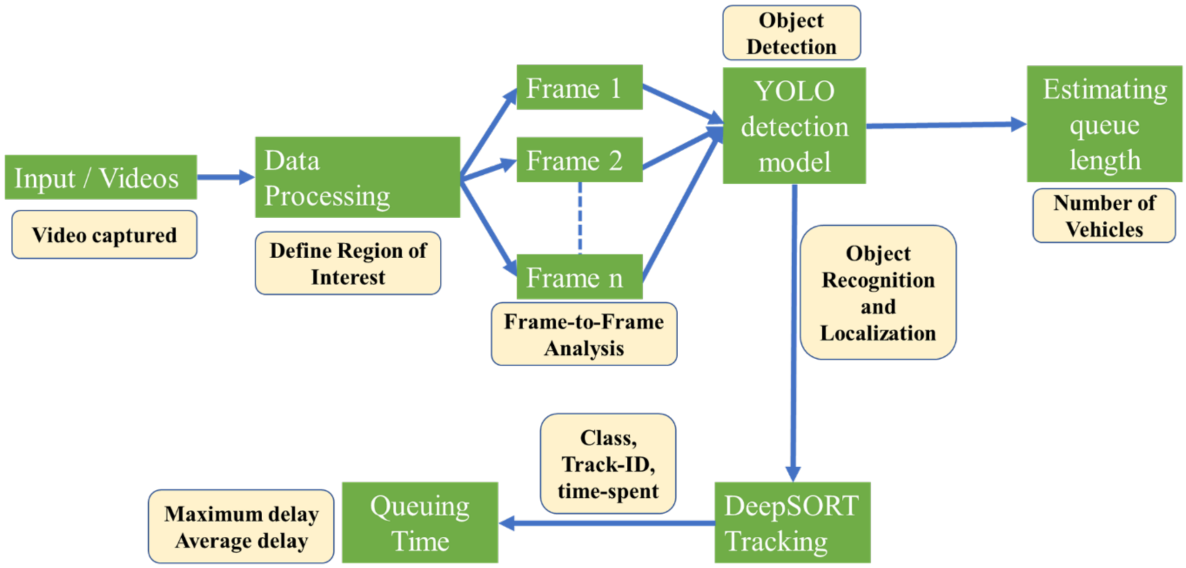


Figure 3.7 Methodology of the Given Framework

This section outlines the application of object detection and tracking algorithms to assess queue length and queue delay. Figure 3.7 illustrates the framework employed for object detection and queue length measurement. Initial data gathering was conducted in collaboration with UDOT-TOC. Prior to inputting each frame into the object detection model, data underwent pre-processing. Object detection was succeeded by a tracking procedure to obtain information regarding traffic parameters.

3.2.1 Data Collection

Figure 3.8 UDOT Traffic Operation Center

Data collection was carried out in collaboration with the UDOT-TOC team from an office in Salt Lake City, as shown in Figure 3.8. To accurately capture traffic conditions on the ramp highway section, adjustments were made to the traffic cameras. These adjustments included fixing the camera orientation, determining the appropriate zoom ratio, and enhancing the video quality. The research team and TAC members closely coordinated their efforts to ensure that the captured video faithfully represented the on-ramp traffic conditions.

3.2.2 Pre-Processing of Data

Data preprocessing was conducted to filter out vehicles that were not situated on ramps, aimed at reducing the occurrence of false positive errors. This involved the exclusion of vehicles from the main highway, parking areas, and other secondary roads by manually coding a mask that covered all regions of the image except the designated region of interest. In Figure 3.9, we can observe the notable impact of this preprocessing step on data accuracy. Without this preprocessing, the system could generate false positives for vehicles located close to, but not actually on the ramps. The input frames provided to the object detection and tracking framework will be subject to this masking process. However, the resulting information, such as the bounding box, vehicle class, and tracking time, will be displayed in the original frame. In Figure 3.9, we can see a comparison between the input frame with masking and without masking, along with the corresponding results.

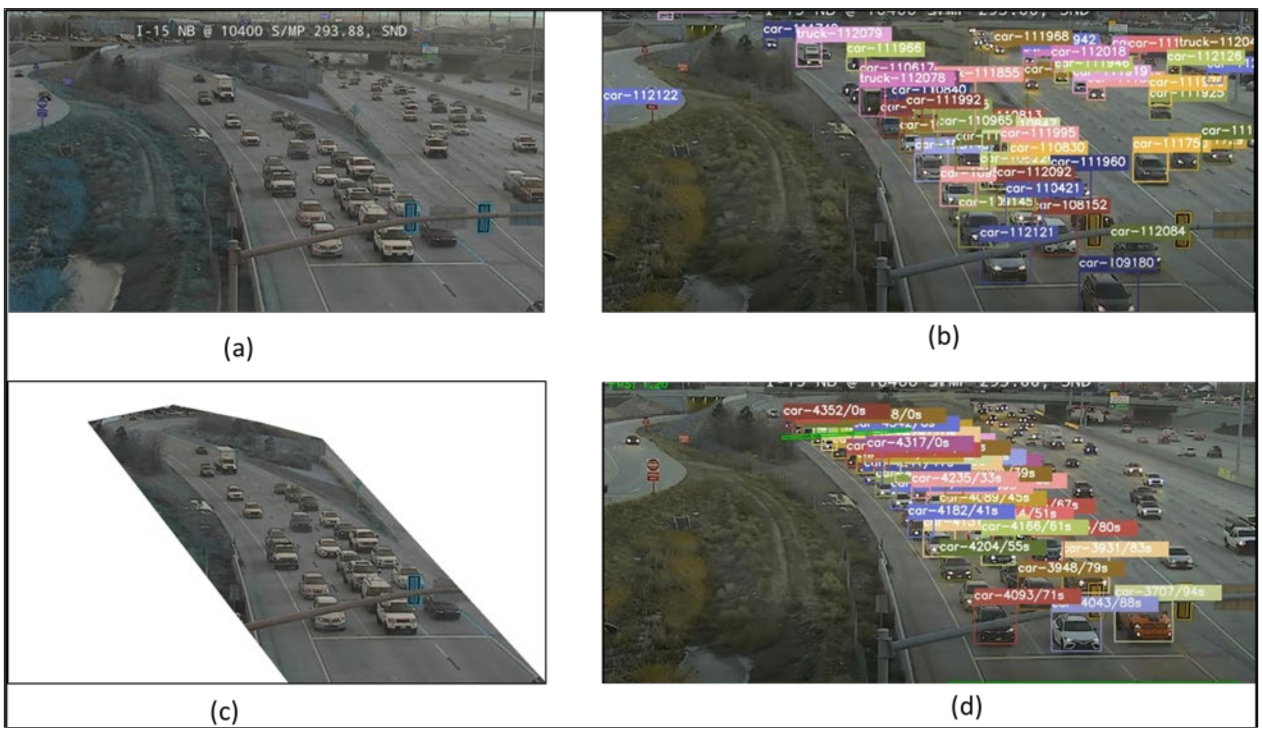


Figure 3.9 (a) Raw Input Frame, (b) Result for Raw Input Frame, (c) Processed Input Frame, (d) Result for Processed Input Frame.

3.2.3 Object Detection and Recognition

Object detection is the task of identifying the presence of objects in individual frames of a video and assigning them to specific categories from a predefined set of classes. To accomplish this, videos are divided into a series of frames, each of which is processed by object detection models like YOLO. These models provide information about the object's class and its spatial location by enclosing it in a bounding box. While numerous object detection methods are in use today, YOLO stands out for its real-time performance and high accuracy. YOLO can detect objects belonging to 80 different classes. After object detection, the subsequent step is object recognition, which assigns a class from the pool of 80 possible classes to the detected object. Figure 3.10 illustrates the results of detection and recognition achieved using the YOLO, with the information inside the bounding box indicating the class of the object, such as "Car," along with a confidence score, e.g., "Car: 0.47."

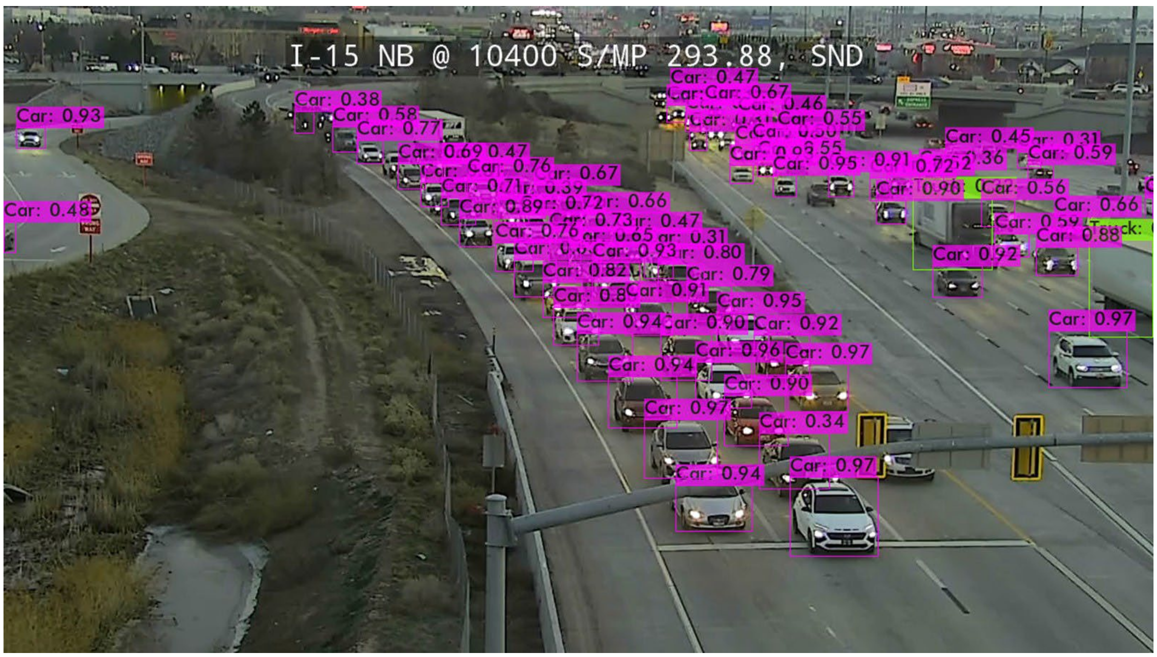


Figure 3.10 Sample Frame after Object Detection and Recognition

YOLOv4 is a notable iteration of the YOLO model, featuring input, backbone, neck, and head blocks for object detection. It was trained on the COCO dataset and can classify objects into 80 different classes. One of YOLOv4's key advantages over its predecessors is the use of CSPDarknet53 as its backbone. CSPDarknet53 employs a cross-stage partial network that combines previous and current inputs before passing them to the dense layer, significantly reducing computational bottlenecks and making the model 20% faster. The head block, derived from YOLOv3, handles the final detection process, providing bounding boxes for detected objects. The neck gathers features from various stages of the backbone and passes them to the head for the detection process, while the backbone is responsible for feature extraction. In addition to these architectural improvements, YOLOv4 introduces a "bag of freebies" and a "bag of specials." The bag of freebies enhances model performance by augmenting the training dataset with methods such as photometric distortion, geometric distortion, Mixup, and Cutmix. The bag of specials, while significantly boosting performance, also increases inference time. Specials implemented in the backbone include mish activation, cross-stage partial connection, and multi-input weighted residual connections (MiWRC). The detector benefits from specials like mish activation, SPP-blocks, and Pan path aggregation blocks.

Despite its strengths, the original YOLOv4 model exhibits lower accuracy in challenging conditions like low lighting and seasonal variations. Consequently, the YOLOv4 model underwent a retraining process using image frames that encompassed such challenging situations. This retraining procedure involved collecting images, labeling them, and then retraining the deep learning model to better handle these adverse conditions.

3.2.4 Object Tracking

Object tracking is a crucial task in computer vision that involves continuously monitoring and maintaining the location information of objects across multiple frames in a video sequence. Various tracking algorithms have been developed, with DeepSORT emerging as a state-of-the-art solution that effectively addresses challenges, such as identity switching and object occlusion, and outperforms other tracking methods. To facilitate tracking, information regarding the bounding boxes of objects detected by an object detection model is supplied to tracking algorithms. These algorithms then monitor the object's position, compute similarity scores between bounding boxes in consecutive frames, and carry forward the tracking ID from the previous frame to the current one if the similarity score exceeds a predefined threshold. Each object entering the frame is assigned a unique ID, which is retained until the object exits the frame. The result of object detection and tracking is displayed, typically showing the vehicle class, a unique tracking ID, and the elapsed time since the object's entry into the frame. This information is formatted as "vehicle class – tracking ID/ time spent," as demonstrated in Figure 3.11 above the bounding box.

DeepSORT is an enhancement of the SORT (simple online and realtime tracking) algorithm, designed to address issues arising from occlusions among detected objects. The SORT algorithm incorporates elements such as bounding box prediction, Kalman filtering, and intersection-over-union (IOU) matching to perform object tracking. DeepSORT, on the other hand, builds on the foundation of SORT by introducing cascade matching and deep appearance descriptors, coupled with Hungarian algorithms for improved performance. The Kalman filter relies on linear velocity methods to predict an object's future position, subsequently matching this prediction with the actual detection in the next frame to compute the IOU matching score. IOU provides a quantitative measure of the similarity between two bounding boxes based on their spatial location and size. If the IOU matching score surpasses the predefined threshold, the tracking ID is transferred, and the Kalman filter updates the tracking data based on the current frame. Furthermore, deep appearance descriptors furnish a similarity score based on cosine distance. DeepSORT combines the scores from both the IOU matching and cosine distance to determine the final similarity of a tracked object between the current and previous frames. Cascade matching is an extension of IOU matching that considers the temporal dimension, aiming to match the most recent detection with newer IDs and older detections with older IDs. This approach effectively mitigates the issue of identity switching between objects detected earlier and those detected later in the video sequence (Wojke et al., 2017b).

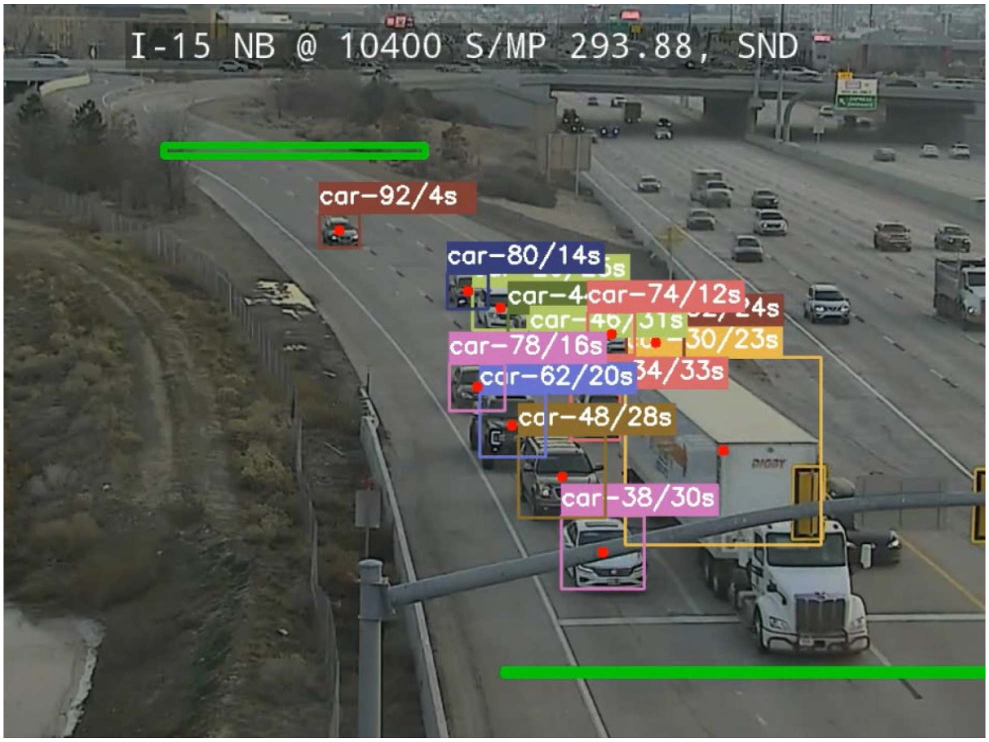


Figure 3.11 Sample Frame after Implementation of Object Tracking Algorithms

3.2.5 Queue Length Determination Using Computer Vision

The term "traffic queue length" pertains to the count of vehicles queued up and waiting to progress along a road or through a signal. The determination of queue length assumes paramount significance when it comes to optimizing signal timing, especially in the context of mitigating the risk of vehicular overflow onto ramps. This optimization involves fine-tuning the signal timing to ensure that the queue length does not surpass the ramp's capacity, thereby averting any spillover. To ascertain the number of vehicles in each lane at any given moment, the system employs a counter mechanism that increments or decrements as vehicles pass through designated entry or exit zones. The count is then presented in the current frame in a formatted manner, illustrating the count per lane as "lane number: total number of vehicles per lane," as depicted in Figure 3.12.

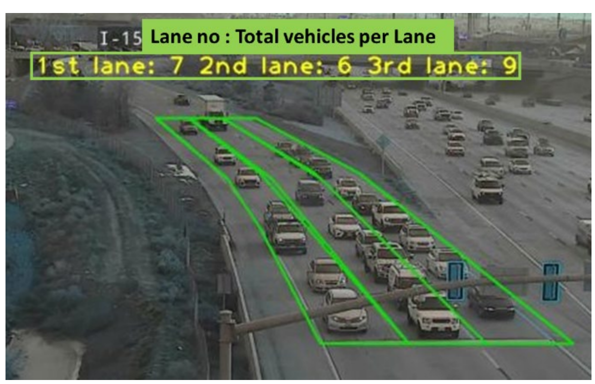


Figure 3.12 Queue Length Detection Using Image Processing

3.2.6 Queuing Time Calculation Using Computer Vision

Queuing time refers to the duration, measured in minutes, that every vehicle undergoes from the moment it enters a ramp until it leaves the traffic signal and merges into the mainline. As vehicles enter the frame, specialized algorithms assign a unique identification number to each object and commence the process of tracking the time it takes for them to traverse the entire exit zone. Within this analytical framework, we meticulously record and store data related to the delay experienced by these vehicles as they navigate through the ramps. These data include the vehicle's classification, its unique tracking ID, and the time it spends within the exit zone. This critical information is then overlaid on the bounding box encompassing each vehicle, displaying it in the format of "vehicle class - Tracking id/total time spent," as visually represented in Figure 3.13. These data serve as vital inputs for computing the maximum delay within the system as well as the average delay. The delay calculation is performed for each individual vehicle with the assistance of its unique tracking ID and the corresponding frame number in the video, which are stored in a Python dictionary. By utilizing these keys and their associated dictionary values, the total time that each vehicle spends within the frame is meticulously computed, ensuring that this duration is continuously tracked until the vehicle successfully exits, as exemplified in Figure 3.13.

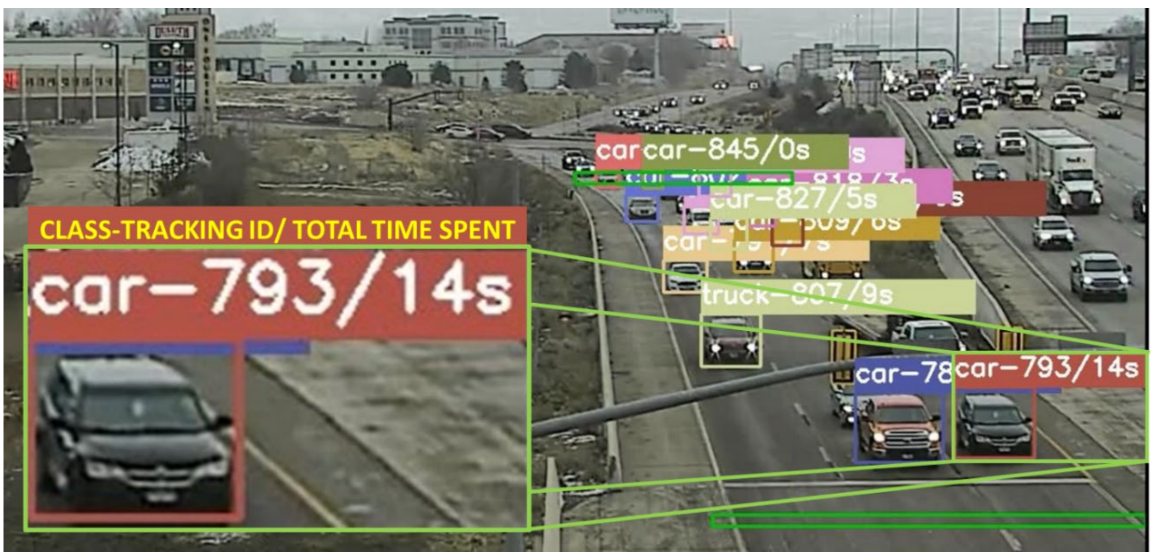


Figure 3.13 Queuing Time Calculation Using Image Processing

3.2.7 Queuing Time Calculation Using Queuing Graphs

A queuing graph is a valuable tool for calculating various traffic-related metrics, including maximum delay, maximum queue length, average delay, and average queue length. To construct this graph, essential data are extracted from video footage, specifically the number of vehicles entering from an entrance zone, their entry times, and the number of vehicles exiting from an exit zone, along with their exit times. The graph typically displays two curves: a blue solid line representing vehicle exits and a red dotted line representing vehicle entries. Time is plotted in seconds on the x-axis, while the y-axis shows the cumulative count of vehicles entering and exiting. The vertical separation between these two curves reflects the queue length at any given time, and the horizontal separation indicates the delay experienced by each vehicle. Consequently, the maximum horizontal distance between these curves represents the maximum delay, which, in the case of Figure 3.14, is 145.73 seconds. The maximum vertical separation provides information about the maximum queue length, which is 52 vehicles in the example of Figure 3.14. In this context, the maximum delay is denoted by a black dotted line, and the maximum queue length is represented by a dashed orange line. The cumulative delay of the entire system can be determined by calculating the total area under the curves, as shown in green on the graph.

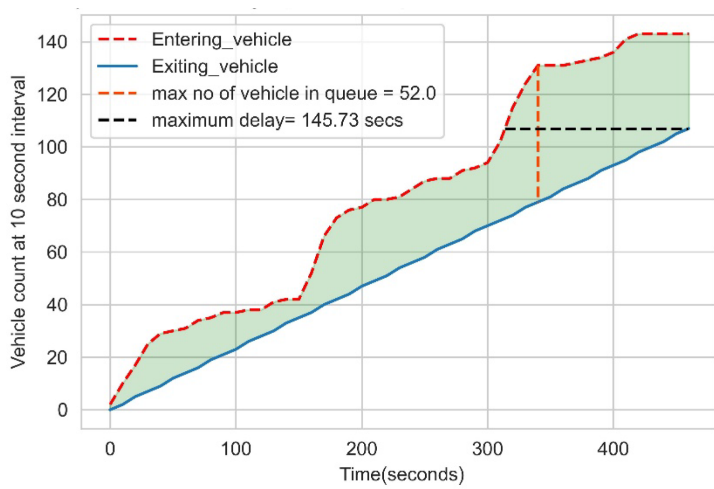


Figure 3.14 Queuing Graph for Ramp Parameters

4. RESULTS AND DISCUSSION

Measuring traffic queue length and queue delay on ramps is crucial for assessing ramp performance. Various sensor technologies can be employed to measure these parameters at on-ramps. However, this study introduces an innovative approach that combines machine learning and computer vision principles to extract traffic measurements, specifically queues and delays, from ramp data. This method utilizes object detection and tracking techniques to determine the required parameters accurately. Given that the effectiveness of this framework relies heavily on the object detection model used, it is imperative to develop a robust object detection model that can provide higher accuracy in these measurements.

4.1 Object Detection and Recognition Performance

Various object detection models are utilized for different applications, and when it comes to traffic detection and analysis in real-time, the YOLO (You Only Look Once) object detection model stands out as a high-performing option. YOLO, known for its exceptional speed and accuracy, excels in identifying objects in video streams. However, it does face certain challenges, particularly in adverse environmental conditions like poor lighting and seasonal variations. Conditions like snowfall and rainfall can introduce additional noise into the video frames, making it more challenging to accurately detect and track objects.

Another critical aspect to consider in optimizing object detection models is the input frame size. The choice of frame size significantly impacts the trade-off between accuracy and speed. Larger frame sizes tend to yield better accuracy, but this improvement comes at the cost of reduced processing speed. Therefore, it is essential to find the optimal frame size that strikes the right balance between these factors.

To evaluate the performance of the YOLO model under various conditions, video samples of different durations, ranging from 1 minute to 15 minutes, were carefully analyzed. In total, 205 minutes of video footage, comprising 6,150 frames, were manually assessed to establish ground truth data. The results, as summarized in Table 4.1, provide insights into the model's accuracy under different conditions and frame sizes.

Notably, the accuracy of object detection and tracking tends to increase as the input frame size is enlarged. However, it is essential to recognize that accuracy is not solely determined by the frame size; external factors, such as lighting and weather conditions, also play a significant role. For instance, the highest accuracy, reaching 72.56%, was achieved when using a frame size of 1024 x 1024 under normal conditions. In contrast, during nighttime, the accuracy dropped to 45.01%, indicating the model's sensitivity to lighting variations.

Furthermore, the developed framework excels in accurately determining queue lengths than calculating queue delays. It is crucial to acknowledge that the speed of processing video frames decreases as the frame size increases. After evaluating various combinations of frame sizes and processing speeds, it was determined that a frame size of either 960 x 960 or 1024 x 1024 strikes the optimal balance, offering competitive accuracy while maintaining justifiable processing speed.

Table 4.1 Object Detection and Recognition Accuracy of Base YOLOv4 Model

Input Frame size		**416 * 416**	**512 * 512**	**736 * 736**	**960 * 960**	**1024 * 1024**
Normal day light	*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 4.1 presents accuracy percentiles for the base YOLO model across varying frame sizes, considering different lighting and seasonal conditions. While the base model exhibits higher accuracy under typical daylight conditions with larger frame sizes, it falls short in delivering competitive accuracy under challenging circumstances, such as low lighting or the presence of external disturbances like rain or snow.

4.2 Retrained Model for Object Detection and Recognition

The basic YOLO model, while known for its fast and accurate results, faces challenges in adverse environmental conditions. To mitigate these drawbacks, a solution lies in retraining the model with manually labeled images that encompass a diverse range of challenging scenarios. Section 3.1.2, Data Collection for Model Retraining, provides insights into the dataset used to depict these challenging conditions. The retraining process involved the utilization of the YOLOV4 base network, and Figure 4.1 visualizes the training progression of the YOLO model. The training loss, an average measure of loss per training epoch, and mAP (mean Average Precision), representing the average precision across all classes, were key metrics in this process. Precision, the ratio of true positives to the sum of true positives and false positives, contributes to mAP. A higher mAP indicates superior model generalization. The culmination of this extensive training effort yielded a final model with exceptional performance, achieved through 120 hours of training across 6,000 batches.

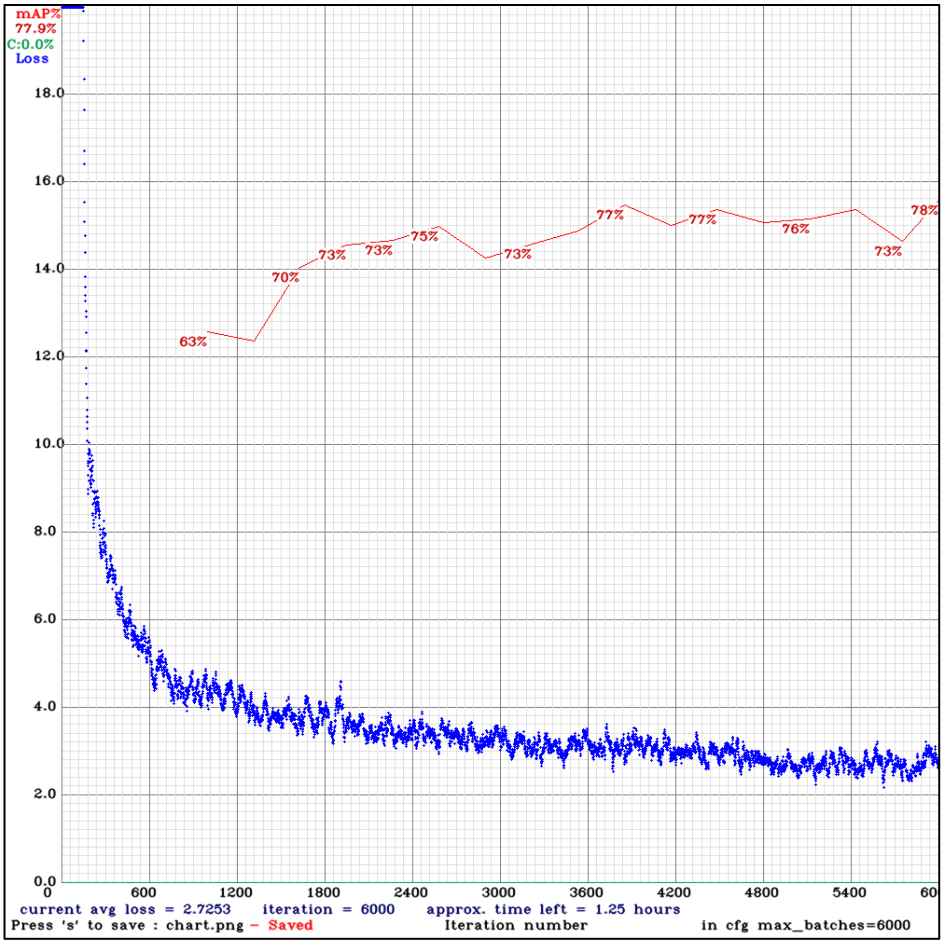


Figure 4.1 YOLOv4 Training Curves

The visual comparison results depicted in Figure 4.2 indicate that the retrained model outperforms the base model in various scenarios, including challenging conditions such as poor lighting and seasonal variations, even when subjected to different types of noise. Consequently, the decision has been made to employ the retrained model for more in-depth analysis and further investigations.

Figure 4.2 Comparison of Base YOLO Model and Retrained YOLO Model

Table 4.2 Accuracy of the Trained Model in Object Detection and Recognition

Input Frame size		416 * 416	512 * 512	736 * 736	960 * 960	1024 * 1024
Normal day light	*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 4.2 presents the performance metrics of a framework utilizing a retrained YOLOv4 model for object detection tasks. Under typical environmental conditions, the framework demonstrates a commendable accuracy rate of 83% when estimating queue length and 82% when predicting queue delay. When we assess the framework's overall performance, it becomes evident that it excels in detecting normal conditions, surpassing its accuracy in inclement weather, particularly during rainy and snowy

conditions. This disparity in performance can be attributed to the adverse impact of rain and snow, which introduces external noise during the image processing stage.

4.3 Comparison of Vision with Queuing Method

As outlined in the preceding section, the utilization of computer vision and queue graphs offers distinct means to independently ascertain both queue length and queue delay. In a study by Sharma et al. (2007), the effectiveness of queuing methods was confirmed by comparing their real-time traffic parameter extraction capabilities with manual ground truth data. Table 4.3 presents a comparison of delay estimation discrepancies between queue graphs and computer vision techniques. The "error" column quantifies the percentage difference between the two methodologies, revealing a minimal disparity of less than 0.02%. This underscores that the vision-based approach yields competitive accuracy in detecting both queue length and queue delay.

Table 4.3 Vision in Comparison to Queuing Metrics

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

4.4 Computer Vision Method Verification with Ground Truth

Ground truth data serve as a crucial benchmark for validating image processing techniques. To conduct this verification, we randomly select video clips spanning approximately 5 to 10 minutes in duration. The resulting differences between these video samples are visually represented in Figure 4.3. In the upper-left quadrant, we observe a queuing graph representing the ground truth data, which were meticulously collected through manual observation.

The top-right segment of the figure exhibits another queuing graph, but this time it is generated using data extracted by image processing models that have undergone retraining. Meanwhile, the figure presents a comparison of image differences between the ground truth data and the results obtained from the retrained models. In this comparison, the green solid line depicts the count of exiting vehicles according to the ground truth data, while the black solid line represents the count of exiting vehicles as determined by the image processing algorithms.

Remarkably, these two lines closely overlap, indicating a high degree of similarity between the ground truth and the image processing results. Any observed minor discrepancies, when scrutinized visually, is due to the presence of a mast-arm in the video near the exiting zone. This mast-arm introduces occasional issues with ID switching, accounting for the small variations between the two sets of data.

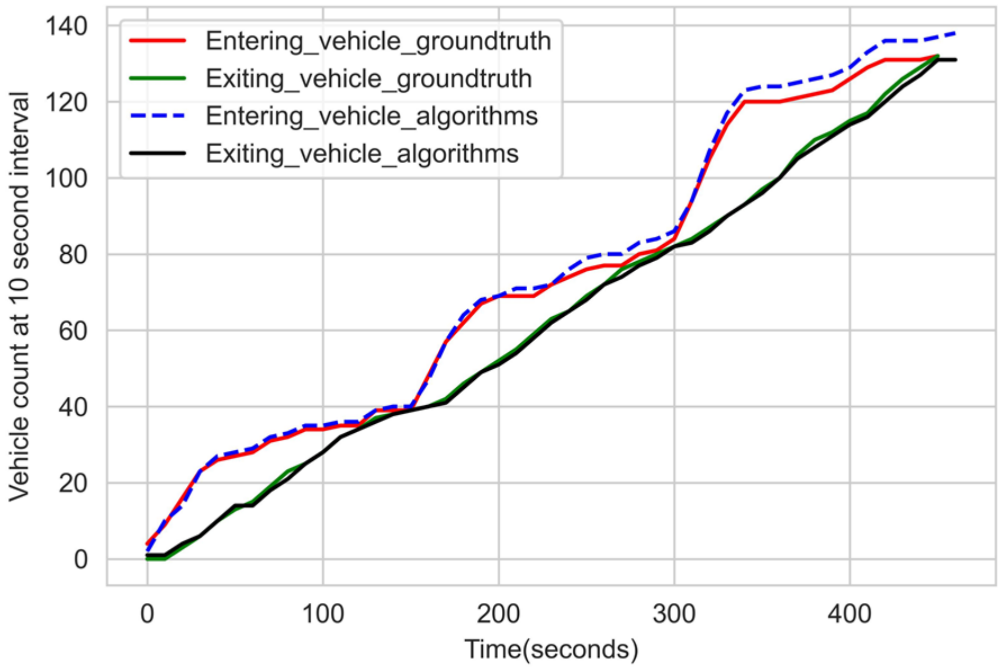


Figure 4.3 Comparison of Vision Method with Ground Truth

4.5 Signaling Strategies with Queue Metrices

Image processing techniques enable the generation of multi-dimensional data arrays from the same video with minor algorithmic adjustments. This data extraction encompasses vital parameters such as vehicle arrival rate and vehicle exit rate, as well as delay and queue information. In Figure 4.4 we observe the distribution of these parameters at 15-minute intervals over a 1.5-hour video analysis. It is important to note that the queue characteristics exhibit variability when either the exit rate of ramp metering or the signal phase is modified. Figure 4.4 visually illustrates how cumulative delay, maximum delay, and maximum queue lengths are affected by changes in the exit rate, while arrival rate remains constant.

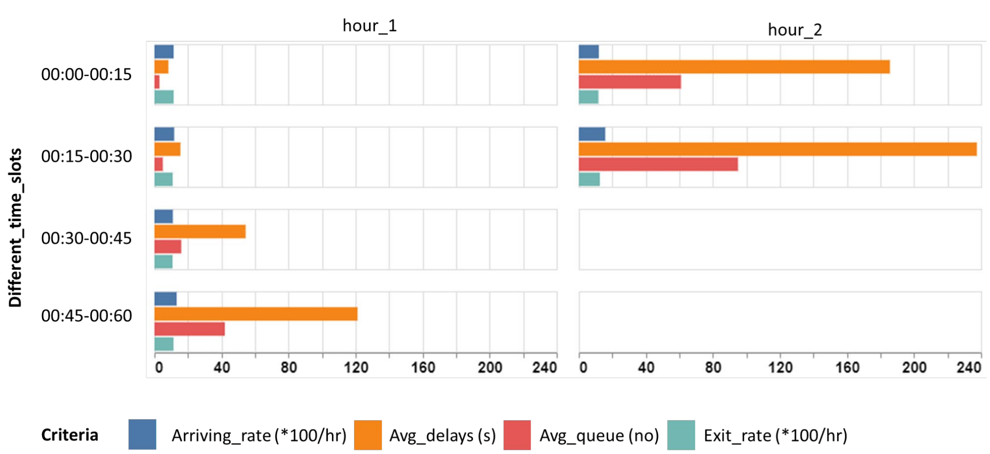


Figure 4.4 Ramp Performance Analysis

Figure 4.4 provides a clear graphical representation of the relationship between a decrease in the exit rate of a signal and the increase in cumulative delay. As expected, when the exit rate is reduced while maintaining a constant entry rate, queues tend to accumulate and delays become prolonged, as evident in the aforementioned figure. Conversely, when the exit rate is increased, the opposite effect is observed. Notably, when the exit rate is increased by 20%, all vehicles experience delays of less than 1 minute. Under the current exit rate, which reflects real-world scenarios at Utah's ramps during the video recording, no vehicle faces a delay exceeding 2 minutes, as shown in Table 4.4.

Table 4.4 Queue Metrices with Respect to 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%
1-2 min	38%	49%	0%
2-3 min	0%	20%	0%
3-4 min	0%	0%	0%
>4 min	0%	0%	0%

Specifically, when the exit rate is decreased by 20%, a significant proportion of vehicles, specifically 20%, encounter delays exceeding 2 minutes. It is crucial to underline that adjusting the exit rate through signal timing is an effective means of ensuring that no vehicle at the ramp experiences a delay exceeding a certain predefined threshold, aligning with established standards, guidelines, or policies while maintaining a smooth flow of mainline traffic.

5. CONCLUSION

In this project, the research team harnessed a combination of image processing techniques and deep learning algorithms to effectively extract crucial information pertaining to ramp queues and delays. They accomplished this by leveraging the video footage obtained from UDOT's preexisting traffic camera network. The team meticulously collected and preprocessed the video data, culminating in the development of a comprehensive framework dedicated to data analysis and extraction. The primary goal was to facilitate the retrieval of pertinent traffic-related insights, which were subsequently utilized to assess and refine traffic signal control strategies. This comprehensive approach incorporated both queuing analysis methods and advanced computer vision models to achieve its objectives.

5.1 Findings

The research team conducted extensive experiments involving a variety of frameworks coupled with multiple combinations of object detection and tracking algorithms in order to identify the most precise and effective framework. In addition, we explored different frame sizes for video processing to determine the ideal balance between speed and accuracy, ultimately settling on a frame size of 960 x 960 as the optimal choice. The team also used statistical analysis tools to quantify the performance of the traffic monitoring system, utilizing data extracted through visual methods. The key findings of this research endeavor can be summarized as follows:

a. Image processing has emerged as a dependable and highly promising sensor technology, facilitating the extraction of real-time traffic information.

b. Leveraging computer vision technology, the team successfully utilized existing traffic camera video footage to accurately measure parameters such as queue length and queue delay.

c. After careful consideration of the trade-off between processing speed and precision, the researchers arrived at the conclusion that a frame size of 960 x 960 stands as the most optimal choice for their video analysis.

d. To validate the effectiveness of their approach, the results obtained through computer vision, using a retrained object detection model, were meticulously compared to queuing methods and ground truth data. Remarkably, the results closely aligned with ground truth values, exhibiting an acceptable level of accuracy.

5.2 Implementation

UDOT places a strong emphasis on the practical application of research findings. In this section, we outline a comprehensive plan for the deployment of a computer vision camera system within the I-15 corridor. This implementation plan comprises three key components:

1. Camera Functionality: Details regarding the operational features and capabilities of the camera system.
2. Image Processing: An overview of the image processing procedures and techniques to be employed.
3. Mounting and Orientation Guidelines: Guidelines for the proper installation and positioning of the camera system to ensure optimal performance and coverage.

5.2.1 Camera Specifications

UDOT currently has three types of camera devices in use: AXIS CCTV, Gridsmart Vehicle Detection, and Flir Vehicle Detection. During our research, only the AXIS CCTV camera was tested for computer vision and was found to be generally reliable.

UDOT is transitioning to a network camera model, the AXIS Q6215-LE PTZ Network Camera. This camera can be assigned to a network with secured login credentials, like other devices managed by UDOT's Traffic Management Division.

Our research has tested computer vision in adverse conditions, including reduced daylight and obstructive weather events such as rain, snow, and wind. Training the computer vision using UDOT's current camera technology has shown that it can still be effective in these challenging conditions. However, upgrading camera technology is expected to enhance visioning capabilities in less favorable conditions.

The AXIS Q6215-LE offers several improvements over the current Axis camera, including better nighttime imagery up to 400m/1300ft, reduced motion blur in low light conditions, and strong backlight handling with WDR features. It also features a wiper to remove rain, dust, and snow. Superior data storage and compression features, such as H.264 or MPEG, make it suitable for computer vision applications. Other features like a 58.6° horizontal field of view, variable focus, and environmental hardening for temperature further enhance computer visioning capabilities.

Our research team believes that deploying the AXIS Q6215-LE camera will significantly improve computer vision accuracy, and future camera technology improvements will yield even better results. PTZ capabilities are important, but for reliable feed into a ramp metering system, cameras must be stationary or returned to a preset calibration position. There are two options to manage this. First, UDOT can install stationary cameras dedicated to ramp metering, freeing up PTZ cameras for custom uses like incident inspection. Second, UDOT can keep some PTZ cameras designated as "dual use" stationary during congested periods when accurate information on ramp delay and queue length is critical. If a camera needs to be moved for special events or incidents during these periods, its detection capabilities may be temporarily compromised. Further research can be done as an extension of this project on the process of recalibrating a camera after a PTZ movement using a deep learning-based segmentation framework.

5.2.2 Image Processing Requirements

The research team recommends utilizing serverless GPUs for real-time processing of ramp video imagery in a cloud computing environment. This approach is favored over GPU cloud servers for several reasons:

1. Automatic data provisioning, eliminating the need for manual setup as required by GPU cloud servers.
2. Faster startup and automatic scaling of video processing with serverless GPUs.
3. Cost-effective "pay-per-use" pricing, ideal for sparse image processing needs during peak hours only (e.g., AM and PM).
4. Maintenance of serverless GPUs is the responsibility of the service provider.

Table 5.1 provides annual cost estimates for subscribing to a cloud computer with serverless GPUs. The assumption is that computer visioning will be utilized for two hours during each AM and PM peak, amounting to 1,040 subscription hours annually. The table displays the 2023 hourly price for a commercial subscription ($1.80/hour), with the potential for rate reduction with recent technology advancements.

Table 5.1 Cloud Computing Subscription Cost

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

5.2.3 Camera Height and Orientation

The choice of camera mounting, and orientation is highly dependent on the specific site and can vary based on the vertical and horizontal alignment of the ramp. Ideally, the camera should be mounted on a stable object, such as a post, to minimize the impact of wind and vibrations. It is not advisable to use mast arm mounting for computer vision systems due to the frequent vibrations. Other key design parameters for obtaining optimal camera imagery include:

1. Mounting height: The recommended range is between 25 and 40 feet, in accordance with UDOT Standard Drawing AT 11A, which often involves 45', 60', or 75' poles.
2. Mounting position relative to the ramp stop bar: Ideally, the camera should be positioned 75 to 125 feet behind the stop bar, toward the mainline.
3. Mounting location relative to the roadway: The specific placement depends on the curvature and alignment of the ramp. However, a camera mounted on a typical mast arm pole with the required clear zone setback is generally suitable.

To assist in determining the optimal camera location, there are commercially available video system design tools. Tools like JVSG, available at https://www.jvsg.com/, were employed to finalize the camera's location, orientation, and pole height. The software demonstration was carried out at the I-15 @ SR-77 SPV SB on-ramp, as shown in Figure 5.1. The following steps should be followed to identify the best camera placement:

1. Import a map from Google by specifying latitude and longitude coordinates.
2. Lock the image in place after zooming and rotating it to the correct orientation for the camera.
3. Select the camera company and model, if available, within the platform. Otherwise, manually enter details about the camera system.
4. Adjust the camera's location to eliminate any blind spots.
5. Model the objects of interest, such as approaching cars, considering their speed, or any potential obstructions like trees in the vicinity.
6. Utilize measurement tools to determine the setback distance and identify the camera pole locations.

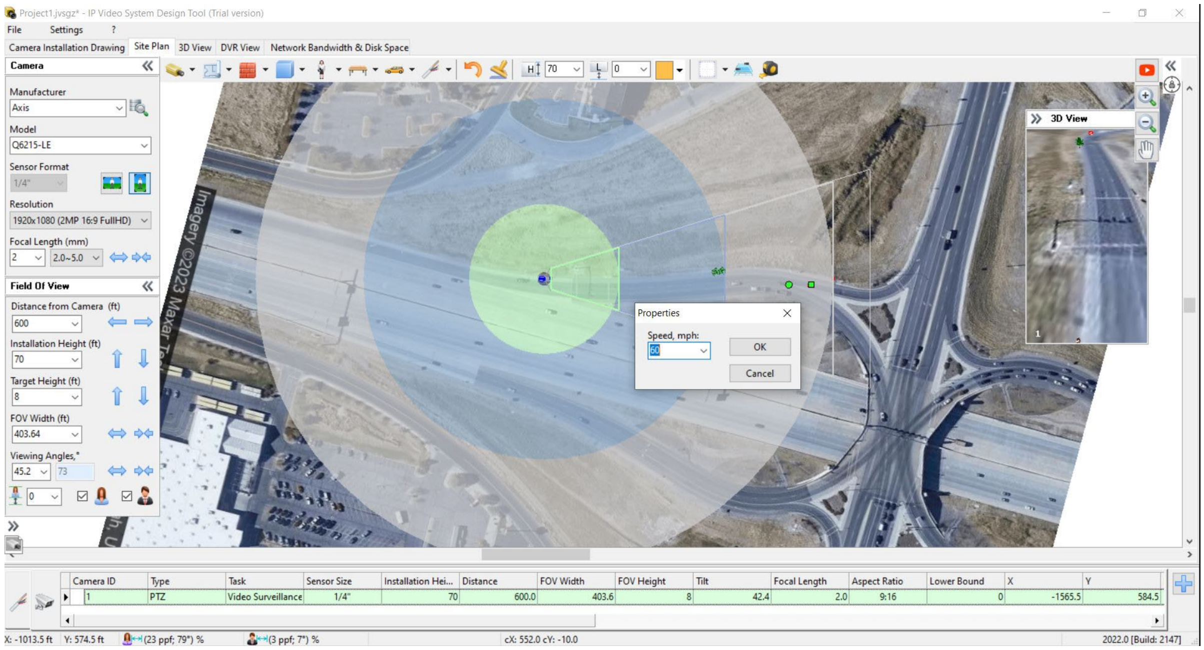


Figure 5.1 Camera Location Modeling

6. REFERENCES

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

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 Sonic. (1997).

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." *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*. Vol. 7, Issue 7, pp. 1655-1661

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

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

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

Rashidi, A., & Karan, E. (2018). *Video to BrIM: Automated 3D As-Built Documentation of Bridges*. 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 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. 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. (2017a). "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

Wojke, N., Bewley, A., & Paulus, D. (2017b). "Simple Online and Realtime Tracking with a Deep Association Metric." Proceedings - International Conference on Image Processing, ICIP, 2017-September, 3645–3649. https://doi.org/10.1109/ICIP.2017.8296962

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. (n.d.). *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." International Conference on Digital Signal Processing, DSP, 0, 276–280. https://doi.org/10.1109/ICDSP.2016.7868561

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