Maldives Journal of Engineering and Technology
ISSN: 3007-6536 (Online)
https://mjet.mnu.edu.mv/

A Camera-Based Intelligent Traffic Signal Control System for Linear Cities: A Case Study of Hulhumalé, Maldives

Amin Rasheed[1*] and Yoosuf Nizam[2,]

[1] *Department of Computer Science, Faculty of Engineering, Science and Technology, Maldives National University, Sosun Magu, Male' City, 20068, Maldives;*
[2]*Department of Engineering, Faculty of Engineering, Science and Technology, Maldives National University, Sosun Magu, Male' City, 20068, Maldives;*

*Corresponding: s081916@student.mnu.edu.mv;

Abstract: Traffic congestion is a growing challenge in Hulhumalé as it develops into a smart city. A major factor contributing to delays is the time vehicles spend idling at traffic lights, particularly during peak hours on the Sinamalé Bridge, where waiting times can reach up to two minutes per signal. Existing smart traffic solutions in other cities often rely on embedded sensors, which are expensive to install and disruptive to road infrastructure. To address these limitations, this project proposes a vision-based traffic light system capable of detecting vehicles in real time and dynamically adjusting green-light durations. By adapting to actual traffic conditions instead of relying on fixed schedules, the system aims to reduce waiting times and improve overall traffic flow. The concept will be tested using a prototype of a three-way junction simulated in Python with a custom trained YOLO-v8 for vehicle detection. Development follows a prototype-based experimental engineering methodology. The expected outcome is a scalable and cost-effective solution capable of reducing average waiting times by at least 30% compared to the current system, offering a practical step toward smarter and more efficient urban mobility.

Keywords: Camera, Sensors, Raspberry Pi, Simulation, YOLO-v8, Hulhumal, Traffic congestion

1. INTRODUCTION

In recent years, there has been much talk of making Hullumale a smart city. Under this, many services and systems that have existed in the Maldives have become digitalized and modernized. This includes application-based ticketing in place in Raaje Transport Link ferries and buses and the "tap to pay" option in buses in the greater Male and Hulhumale region, according to the "Payments Bulletin 2024" released by the Maldives Monetary Authority. However, there seems to be no mention of implementing one of the hallmarks of many existing smart cities, which is a dynamic traffic light management system. Having a smart traffic light system would alleviate some of the problems that come with being densely

Received: 22 August 2025
Accepted: 3 November 2025
Published: 30 November 2025

 https://doi.org/ 10.70592/mjet.2025.2.02.006

populated. Unfortunately, implementation of already existing systems would prove to be costly, as most of the systems require sensors built into the road and are not meant for linear architecture [16].

The aim of VISTA (Vision Integrated Smart Traffic Automation) is to dynamically adjust the traffic of the current intersection such that the lane with more vehicles gets more green light and ones with less or none get the minimum amount of green light. The simulation would be used to see the time taken for both the proposed system and the current system, which would be compared to figure out how effective the proposed system is.

2. LITERATURE REVIEW

By the very definition of a smart city, it should "encompass a broad use of technology to gather and process information for monitoring, optimizing, and managing a city." [28]. An aspect of a smart city as such would be a way to optimize traffic. This would prove to be beneficial for us as an ever-growing smart city. Many systems that make traffic efficient already exist in many developed cities, but as previously stated, they are built with their respective country in mind. As such, they would not be easy to implement in this country, either, due to software or manpower limitations. Therefore, when proposing a system to manage traffic specifically geared toward the Maldives, it should be important to look at the literature that proposes how the authors would ideally implement this. This can be achieved by looking at what sensors those systems use and what algorithms they have opted to utilize.

As such, in this literature review, we will be looking at systems that have used different means of input, highlighting how said input works for their system and the benefits of using said input while also going over some of the challenges associated with using said means. The analysis of algorithms is going to take the same path, describing their architecture, their advantages and disadvantages.

2.1 Sensors

2.1.1 Introduction to Sensors

[4] classifies sensors as intrusive and non-intrusive. Intrusive sensors are sensors that require a connection to be made between the sensor and object being measured, while non-intrusive sensors do not need that connection. When thinking of making a dynamic traffic system, it is important to take this into account, along with what type of sensors are being used and their advantages and limitations.

2.1.2 Magnetic Sensor

These are a type of sensor that is used to monitor changes that occur in the earth's magnetic field, which is oftentimes caused by metals that have magnetic properties [11]. [4] have said that, even though there are a multitude of ways to detect passing vehicles, they all require the sensor to be installed inside traffic lanes.

Though installing these in traffic junctions that have already been built would prove to be disadvantageous, if they were installed before the junction was built, they would be low cost and would not take that much energy [11]. These sensors have the ability to be used in areas that have highly varying weather conditions, as they are not affected by them unless they are extreme, and have a wide variety of uses, such as counting vehicles, finding the speed of vehicles that cross them, and even classifying vehicles based on the vehicles' height and width [4].

However, it is not as perfect as it may seem. In real-life tests carried out by [6], they encountered a multitude of errors, such as heavy vehicles such as trucks being detected in an unassociated lane, vehicles being detected while they pass a lane that is not related to the route said vehicle is taking, and double counting, which occurs when vehicles that have two different points where metal is being concentrated, such as a truck and a trailer, cross the sensor. These sensors would also need to be installed very close to the lanes, as tests conducted by [4] in scenario 2 revealed that they need a minimum of 2 meters in order for the sensor to register that something has crossed. They have also stated that placing them outside of traffic would falsify the data the sensor receives, giving the system a more difficult time in detecting vehicles.

2.1.3 Accelerometer

Defined by [3] as a sensor that senses the vibrations of vehicles that pass by it. These vibrations can be used to find the number of vehicles and the speed of vehicles. Categorized as non-intrusive by that same author, these sensors consume low energy and are relatively easy to install

It has been tested that, by using three-axis accelerometers, a system can detect what kind of vehicle it is in diverse traffic lanes, though it should be moderate [3]. In addition to this, the same author has stated that by using a combination of extended Kalman filters, a speed model, and a signal model that consists of wave propagation, excitation, preprocessing, and identification, a system can track and estimate the speed of vehicles within a given traffic. He has also stated the use of an algorithm, which uses microphone and magnetometer sensors to adaptively change the threshold in order to detect vehicles, can be utilized by accelerometers by detecting vibrations made by vehicles.

These sensors, just like magnetic sensors, would need to be installed within the road itself. In the system made by [4], the accelerometer he had used has a range of +/-2g, and the average accuracy of the vehicle detection across all axes in scenario 1 is 50% while in scenario 2 it rose to 60%, making it lower than magnetometers, which in scenario 1 is 79% and in scenario 2 is 93%. The author has stated that the accuracy of systems that utilize accelerometers heavily depends on the sensitivity of the sensor, what kind of road the sensor is on, and how it is mounted. These sensors are also prone to give wrong values when placed in heated areas or will have a small offset, which over time would cause calculations to be off [33]. The same author has also stated that when placed near electrical components, small fluctuations would occur, causing the reading to be slightly off.

2.1.4 Distributed Acoustic Sensor

The modern version of an acoustic sensor, the distributed acoustic sensor (DAS), is categorized by [35] as a technological advancement belonging to the category of distributed fiber-optic sensing. As [9] have put it, DAS is based on converting the unused fiber-optic cables, also known as dark fiber, into a dense array of acoustic sensors. The system works by making bursts of light that have a constant phase relationship occur when vibrations happen on the cable, which is sent from one end of the cable to the other. Due to how the physics of light works, small fractions of the light get reflected back towards the detector due to Rayleigh scattering, allowing the fiber cable to be used as a sensor.

Since this technology can utilize fiber optic cables that are not in use, we can say that this technology is low cost and is very easy/fast to install while being relatively low maintenance [35, 37]. This can also be used to cover many meters of a road, thus making sure that there are no blind spots, or areas that cannot be detected by the sensor [9]. It is non-intrusive and, according to [9], is not affected by weather conditions in the slightest. This technology also offers maximum privacy to people on the road while also being able to identify the weight of vehicles, making classifications that much easier.

While it appears highly promising, it is not without flaws. According to [37], without machine learning being implemented on the waves being sent back, long-distance monitoring will become tedious due to the massive volume of data that is sent back. In the conclusion of the paper written by [9], the author has stated that, though under controlled conditions labeling of data is possible, in a real-life scenario, another data input source would be needed. It is also good to state that, unless you use an array-based approach, there would be false detections being made by other unrelated sources, such as a tree swaying in the wind [37].

2.1.5 Inductive Loops

[27] classifies approaches that collect traffic data as two different categories: in situ and Floating Car Data (FCD). The latter is defined as methods that involve locating each vehicle through either GPS or cellular phone. In situ is defined as a method that collects data using sensors placed along the road. The author states that the inductive loop sensor is in the category of in situ. [4] have stated that these sensors work by measuring the inductance value of a wire placed within the road. The system presented by [25] has three main sections: the inductive loop section, the electronic vehicle detector section, and the counter section. All of these sections work together to count the number of vehicles present within the traffic.

When used with good-quality sensors and having a good strategy for deployment, these sensors can measure the volume of traffic at the given intersection, how occupied it is, and what speed a vehicle is travelling [20]. This can be linked to the fact that it provides higher precision than other techniques commonly used, as stated by [11], and the fact that it is not affected by weather conditions such as sunlight or rainfall, as stated by the same author.

These inductive loop sensors have some unfortunate events that could be experienced when they are not properly placed. In the attempts of [20] to create a simulation with the help of the data which they had in Munich, it was mentioned that in spite of the fact that it was possible to detect traffic at the intersection, because of the way they were located, the system would not know whether a car turned or not. Other statements in the same report indicated that the detectors lacked the capability of differentiating the type of vehicles. The author of the journal by [4] has implicitly suggested that it will be hard to install such on existing roads because they will have to install on the inner parts of the road, which would mean using more manpower and money than required.

2.1.6 Light Detection and Ranging

More commonly referred to as LIDAR, [8] and [31] describe it as a sensor that has high accuracy when used to map things out on a map, making it a key component in most of the systems used to maximize the efficiency of traffic. Enabling the measurement of speed and classification of vehicles, LIDAR utilizes pulsing laser lights that are used to detect objects and calculate how far they are from the traffic light [4]. The calculations done are based on the time interval between the sensor sending the pulse and receiving it back [2].

In the abstract of [8], the author states that compared to other sensors utilized in smart traffic systems, LIDARs offer high performance and precision. This is backed up by [2], who also adds that since these sensors are weather resistant, they are the perfect candidate for measuring traffic dynamics where data inaccuracies may lead to unfortunate situations. [8] also state that LIDARs provide anonymity among the drivers while maintaining that top-of-the-line data reliability it boasts about. Findings from [2] have also proven that due to their real-time data collection ability, LIDARs have reduced the chance for a gridlock to happen at intersections due to signal failures or congestions. The author also adds that due to its non-intrusive nature, there is no need for road closure to take place in order to place the sensor.

In the observations made by [31], the authors have stated that, when the number of lanes increases, capturing cars in the lanes they are supposed to be becomes more challenging. These sensors are also more expensive compared to others and oftentimes consume a lot of energy [4]. In the research made by [4], it was observed that some beams of light emitted by the LIDAR have the chance of being avoided by vehicles given that the lane is wide enough. The same author has also stated that without a shield or an object present within its sensing range, LIDARs lose their stability and get rendered ineffective.

2.1.7 Ultrasonic Sensors

Unlike LIDARs that utilize light as their means of measuring distance between themselves and objects, [13] have stated that these sensors use ultrasonic waves as their means of measuring distance. These sensors work through the emission of ultrasonic waves, time that the waves are sent to reflect back and finally, the distance is calculated. When combined with the algorithms ALPR, YOLO, GDPR, and CNN, data from the ultrasonic sensors can be used to automatically count the number of vehicles [34].

[13], [30], and [34] have all stated ultrasonic sensors are more cost-effective compared to other means. In addition to that, [30] have also mentioned that these sensors offer a balance between effectiveness, accuracy, and simplicity when using them for real-time applications. In the discussion of [34], the authors have stated since these sensors do not require light in order to work, they can be used any time of the day. The system proposed by [13] utilized ultrasonic sensors, as they can be installed and maintained on a larger scale compared to other forms of detecting traffic. The analysis of the system made by [30] also includes the fact that these sensors proved to have a high accuracy along with being very responsive.

In [34], the authors have stated that when vehicles are too close together, which oftentimes is the case in a place like Maldives, errors within the data will arise due to signals getting mixed. The same author has also stated that these sensors are unable to identify vehicles that remain too close to the sensor as well. In [21], the authors have mentioned that ultrasonic sensors are noise sensitive, and the data can be distorted due to environmental factors such as temperature, air turbulence, and humidity. [4] have documented that these sensors should not be used in lanes that have many lanes or have high traffic volume, as the detection accuracy plummets when doing so.

2.1.8 Camera

Often hailed as the most common method of data collection in smart traffic systems, cameras are often used as a way to capture images at different lanes in order to monitor the current traffic density as described by [10]. Different authors have used different cameras in their systems/prototypes. [32] suggests the usage of Closed Circuit Television (CCTV); [10] used a high-resolution 1080px camera; and [36] used an ESP32 camera module kit in the authors' experimental approach, while for the prototype they used an iPhone XR smartphone along with a selfie stick for balancing purposes. In all the systems, the camera acts as a means to get the live data, which is then fed into different algorithms. [18] and [32] have opted to use versions of You Only Look Once (YOLO); [36] have used a combination of the Python library OpenCV for image processing and Background Subtractor MOG2 to differentiate moving objects from the stationary background; and [10] utilized edge detection as its core process to extract the outlines of vehicles.

When combined with software solutions, cameras offer high accuracy compared to other means of vehicle detection [36]. The same author has also stated that cameras, when combined with computer vision, offer a lower-cost solution to the more commercially available sensors that require constant maintenance. The cost can be further reduced by utilizing open-source technologies as well. Systems that utilize cameras as opposed to sensors, like the one proposed by [10], can operate without the need of external hardware such as sensors and wireless routers. The author's system also detects vehicles in real time without failure; as such, the green light given to the lanes that require it the most is more compared to other systems even at unexpected times. Due to this, the chance for traffic congestion to occur is less. The system proposed by [18] has the ability to detect vehicles no matter how far they are from the camera. In the abstract of [18], the authors have stated that their system was able to detect motorcycles at a 100% rate, which is exceptionally good for a country mostly dominated by motorcycles like the Maldives.

According to [32], the accuracy of their proposed system drops when the vehicles cast a huge shadow or during the night when headbeams would create areas that would meet their detection criteria. When utilizing simple motion detection software like how [36] has done, the accuracy drops when faced with heavy traffic. In the results and analysis of [18], the authors have stated that unless you use the pretrained dataset with a lot of images, the system cannot identify small vehicles that are far away. If you are utilizing edge detectors as your means of identifying vehicles, the detector you use will have its own drawbacks as well. According to [10], Roberts is prone to noise, Sobel takes more time to compute, Active Contour is both prone to noise and takes more time to compute, and Laplacian of Gaussian does not detect all the edges efficiently. [36] have also stated that in order for systems that use cameras as the means of gathering data to work effectively, a stable power connection is necessary.

2.1.9 Discussion on Sensors

When making a smart traffic system, it is paramount that we know what kind of sensor would best fit the system we are going to make. What type of sensor, intrusive or non-intrusive, also matters as much as that [4].

Among the discussed intrusive sensors, magnetic sensors offer reliable performance regardless of the weather condition and have the capability of being used to count the number of vehicles, find the speed of said vehicles, and classify them according to vehicle type [4, 11]. Accelerometers, another intrusive sensor, require less power to operate and are relatively easy to install compared to other discussed sensors [3]. However, according to [6], magnetic sensors, when used in real life, suffer from double counting and misclassification, and according to [3, 33], accelerometers are sensitive to temperature and to electrical noise. The last intrusive sensor, the inductive loop sensor, is precise and, just like magnetic sensors, operates well regardless of weather conditions [11]. They can be used to find the total number of vehicles at an intersection, determine how occupied said intersection is, and determine the speed the vehicles are moving at [20]. Unfortunately, according to [4], their installation and maintenance on existing roads would be labor intensive and would prove to be costly. In addition to that, according to the results of [20], inductive loop sensors are unable to distinguish between vehicle types or what direction they are turning when exiting the area the sensors are installed over. This category of sensors is not fit for our system due to the fact that they would require the roads to be closed in order for the sensors to be installed.

Since intrusive sensors are not what our system is looking for, let us revisit the discussed non-intrusive sensors. DAS enables the system it is being used by to detect vehicles regardless of weather while also allowing high spatial coverage [9, 35]. However, [37] have stated that the system generated massive volumes of data, requiring them to use machine learning to interpret it. The authors have also stated that without the proper signal filtering, the chance that the system produces false detections would be higher. LIDAR, another non-intrusive option, provides exceptional precision, anonymity, and allows for real-time data collection [2, 8]. Though these sensors are ideal for mapping and classification of vehicles, they are extremely expensive and require a lot of energy to work effectively, and according to tests carried out by [31], they also become less effective when being used across multiple lanes or wide intersections. Ultrasonic sensors are cheaper and offer easier ways of detecting the vehicle, unlike LIDARs [13, 30]. Though these sensors perform well under various lighting conditions [34], said performance would fall off when met with high-density traffic while also being sensitive to environmental factors such as temperature and humidity [21].

Though the sensors stated above all have their own strengths and weaknesses, for this system, we would be using cameras. According to the authors [10], [18], [32], and [36], cameras are the most common method of detecting traffic around the world due to their versatility and their ability to be integrated with computer algorithms such as You Only Look Once (YOLO), OpenCV, and edge detection techniques. When combined with the author's choice of object detection, these systems would get the ability to monitor traffic conditions in real time while also being cost-effective. However, according to the tests carried out by [32], the system the author had made performs poorly when met with low-light conditions, shadows, or glares. In addition, according to [36], the authors' system relied heavily on a stable power source and high-quality datasets. However, those same authors [32, 36] have also stated that these limitations can be mitigated by having better training datasets and a better detection algorithm, which is an area we will focus on when making the system.

2.2 Traffic Detection Algorithms

Since this system will be using cameras as its means of input, it is important to know the ways some of the proposed systems process those images, that is, what traffic detection algorithms those systems have used. [15] classifies object detectors into two categories: single-stage and two-stage. According to the author, two-stage detectors split the detection into two stages: feature extraction and then regression along with classification, while single-stage detectors combine the two stages into one, allowing an object to be found and identified in a single stage. Both of these detection algorithms have been used in traffic management systems proposed by people around the world. As such, this section would be outlining the most popular options from the two types of object detectors that have been used in traffic systems.

2.2.1 Single stage: You Only Look Once (YOLO)

Out of all the single-stage detectors being used, YOLO has been used more than all of them [15]. First made in 2015, the core structure of YOLO, according to [17], allows the use of neural networks to directly output the position of the object and its category with its bounding box.

The latest iteration of YOLO, which is YOLOv12, was released in the year 2025, 10 years after the first iteration of it [1]. According to the authors, this version had signified a huge advancement in the area of real-time object detection. The making of YOLOv12 represented "a paradigm shift through the integration of attention-centric mechanisms, streamlined architectural designs, and optimized training pipelines" [1]. Though the architecture became more complicated, it still maintains the speed the YOLO series is known for. The authors have also stated that this version of YOLO can be broken down into 3 main components: **Backbone**, **Neck**, and **Head.**

- Backbone: Providing basic representations needed for tasks that require detection, the task of the backbone is to transform raw images into multi-scale feature maps [1]. Since it utilizes Residual Efficient Layer Aggregation Network, or R-ELAN, the ability of the model to capture the tiniest detail on varying sizes of objects has been boosted. A new convolution block that prioritizes lightweight operations and higher parallelization, which was not utilized in previous versions, is used here. According to [1], YOLOv12 processes faster without ruining the quality due to the fact that computation is done across multiple small convolutions rather than a few larger ones. In addition to the new convolutional blocks, this version of YOLO reduces the load on computation by utilizing techniques such as 7x7 separable convolutions, which effectively keep up the spatial awareness whilst reducing the number of parameters [1].

- Neck: Working as a channel between the backbone and the head, the neck of YOLOv12 sums up and refines the multi-scale features. According to [1], the area attention mechanism, which is a key innovation, increases the model's ability to focus on key areas in cluttered scenes. This reduces the number of memory transfers and the overhead for computation, which allows YOLOv12 to inference in real-time at even higher input resolutions.
- Head: Bounding box coordinates and scores are generated here. According to [1], this component of YOLOv12 converts the feature maps, which have been refined by the neck, into the final predictions and has streamlined multi-scale detection pathways. In addition to that, this component utilizes a specifically made loss function that balances objectives by extending a typical "YOLO-style" loss to incorporate new attention or confidence terms. The performance of YOLOv12 is enhanced due to the incorporation of these new terms [1].

The authors [1] have stated that, by making all of these components work together, new standards for object detection in terms of speed and accuracy have been set.

Though the system as a whole may be good, when used in systems such as a smart traffic system, external conditions must be considered [22]. In [22], the author has commented that due to weather conditions, such as weather, inaccuracies in traffic counts were met. This has led to inaccurate traffic estimates. The author has also mentioned that the right street lamps must be put in place in order for the system to work during nighttime. Classification of vehicles was also difficult for the author due to the fact that there were no specific lanes for each class of vehicle. In addition to that, since the author had opted to use YOLOv7, the starting cost was higher due to the installation cost. Due to the simplified architecture of more lightweight versions such as YOLOv4-tiny, the detection accuracy may become worse [23]. In [22], the author has stated that when using YOLOv7, due to its super-fast reasoning rate, there was a tiny loss of reliability, especially when tiny or unclear objects are present.

2.2.2 Two stage: Region-based Convolutional Neural Networks (R-CNN)

[5] have stated that the region-based convolutional neural network, more commonly referred to as R-CNN, has revolutionized object detection methodologies and has redefined the areas within image analysis. Characterized by the author as a two-stage object detection algorithm, R-CNN combines region proposals made with deep convolutional neural networks (CNNs) along with bounding box regression, specifically designed for object detection.

The typical R-CNN architecture consists of three main components: **Region Proposal**, **Feature Extraction**, and **classification and localization [5]**.

- Region Proposal: This is where the process starts. According to [5], the region proposals are essentially rectangular boxes on an image that the algorithm determined has the possibility of containing an object. 2000 of these region proposals are made, which are produced by a selective search algorithm. The author states that the selective search algorithm works by grouping images that look similar, such as by having the same color or texture. This process is done iteratively. These 2000 boxes are then filtered out and resized to ensure a consistent input for the next image in line [24].

- Feature Extraction: The resized regions being made in the first step are then passed through a CNN, which, in the case of most R-CNNs, uses the AlexNet Architecture [5]. This is done to extract the relevant features. Visual characteristics such as shapes, colors and textures are captured due to the generation of a feature vector made by the CNN. These vectors are then normalized to a fixed length in order to reduce the impact of variations.
- Classification and Localization: he normalized vectors are then fed into two separate branches: classification and localization [5].
    - Classification: The vectors being made from each region are fed into a Support Vector Machine (SVM) Classifier [24]. The classifier outputs a probability distribution over the classes, that of which the highest will be assigned to the region proposal [5].
    - Localization: The vectors made in the feature extraction component also pass through a linear regression layer [5]. Within this regression layer, regression parameters such as offset values are made, which are used to predict bounding box coordinates. According to Maity et al. (2021), these predictions refine the original proposal coordinates made in the region proposal component.

[5] have stated that, despite all of the advantages R-CNN brings to the table, due to the feature extraction being performed in every region, the computational cost of using R-CNN is significantly high. Though this high cost has hindered its ability to be used in real-time applications, further iterations of R-CNN have tried to alleviate some of that hindrance.

Out of all the variants discussed by [5], Faster R-CNN is the best variant for a smart traffic system. This variant has been investigated extensively by [7] and [24] as the backbone for the advanced research in smart traffic management. This variant provides the necessary vehicle localization efficiency needed for traffic monitoring [7]. In addition to that, it is being touted as the fastest variant of R-CNN by [5], even when compared to later variants like Mask R-CNN. This speed is being achieved due to the variant integrating the Region Proposal Network into the algorithm. As stated by [29], the speed became of Faster R-CNN became a critical factor in the system made by the author. The system proposed by those authors achieved an accuracy of 95.7%. In the analysis done by [5], Faster R-CNN has the lowest computational requirements out of all 4 variants. In that same analysis, it is also stated that compared to the base R-CNN, Faster R-CNN has a higher mean average precision as well, that being 0.78. [29] have also added that this system handles overlapping objects better than one-stage detectors, due to Faster R-CNN's architecture.

Though it may seem as if Faster R-CNN is the better option overall, [5] have stated that, if we were to use Faster R-CNN in an application needing real-time detection, like a smart traffic management system, a significantly powerful GPU becomes a necessity. The same author also adds that using Faster R-CNN means that parameter tuning must be done in a more careful manner compared to when using other algorithms. Though it has the lowest computational requirements according to [5], the model made by [29] ran into some issues, as, according to those authors, models using Faster R-CNN require high computational resources for training. [5] have also stated that, though faster R-CNN improves speed compared to fast R-CNN, it has lower accuracy compared to it.

2.2.3 Discussion of Detection Algorithm

When using cameras as a means of vehicle detection in any system, having an object detection algorithm that fits the situation is paramount to the success of said system. Algorithms that fit both of the categories defined by [15] have been used in many proposed traffic management systems, each having its unique advantages and disadvantages.

The YOLO family of models has been the most popular amongst the single-stage detectors, especially in traffic management systems. Its architecture allows object localization and classification to be done using a single neural network pass [17]. The latest and most advanced version, YOLOv12, represented a major step forward in real-time object detection. Through the integration of attention-centric mechanisms, optimized training pipelines, and improved architectural efficiency, the development of YOLOv12 represents a major step forward in real-time object detection. The system maintains YOLO's traditional speed advantage while improving its detection precision.

Despite these improvements, authors have noted several limitations when YOLO is implemented in real-world smart traffic systems. [22] reported that environmental factors such as rain, glare, or low light can cause inaccuracies in vehicle counting and classification. The authors also have highlighted the importance of street lighting, as when faced with a situation that has poor lighting, detection accuracy falls. This lack of detection accuracy can also be attributed to the lack of lane separation between vehicle types as well. [22] observed that YOLOv7, while extremely fast, occasionally suffers from reliability issues when detecting small or partially obscured objects. [23] found that lighter versions such as YOLOv4-tiny compromise accuracy for lower cost because of their simplified architecture. Overall, YOLO's high speed and reasonable accuracy make it favorable for real-time traffic applications, though it remains somewhat sensitive to environmental conditions.

Among two-stage detectors that are used in smart traffic management systems, R-CNN prioritizes precision over speed [5]. However, although the base version of R-CNN attains high accuracy, the author has stated that it is associated with a substantial computational cost. This cost is elevated the most by the Faster R-CNN variant of the base version, which is proven by the computational table by [5]. Studies by [7], [24], and [29] demonstrate its capability in dynamic traffic management systems, achieving accuracies as high as 95.7% and a mean average precision (mAP) of 0.78. In addition to that, [29] have added that, the Faster R-CNN variation has better handling of overlapping compared to other models.

Though Faster R-CNN was meant for real-time applications, [5] note that it would need a powerful GPU for it to function well. [29] reported that the training process remained computationally expensive. Furthermore, although Faster R-CNN improves upon earlier variants in speed, its accuracy is slightly lower than that of Fast R-CNN [5].

As we take all of the advantages and disadvantages of both of the algorithms, along with the challenges the authors faced when using said algorithms, this proposed system would use YOLO as its detection algorithm. As [5] stated, though R-CNN and Faster R-CNN boast strong accuracy, they come with hardware demands. In addition to that, it is less capable in real-time applications compared to any algorithm in the YOLO family. YOLO also offers a balance between providing high-speed processing, adequate accuracy, and computational efficiency, which are all ideal for continuous, real-time vehicle detection in diverse traffic conditions [1, 15, 22].

3. METHODS AND METHODOLOGY

Having a good, clear outline on how a project will move forward directly correlates to the success of said project. The first step in having a clear outline is choosing a methodology the project will follow. As such, this chapter outlines the methodology adopted for this study, along with the tools and technologies used throughout the development of the proposed system.

For this project, a prototype-based experimental engineering methodology was selected. This approach is suitable because the primary objective of the project is to construct and demonstrate a proof-of-concept model of a smart traffic management system. This project aims to verify the feasibility of deploying a smart traffic system such as VISTA in the Maldives by making a physical model of the three-way intersection on the bridge connecting Male', Hulhumale', and Hulhule'.

Construction, testing, and refinement of system components are all enabled in an iterative manner when using prototype-oriented methodologies. The most appropriate methodology amongst the available prototype-based methodologies was determined to be experimental engineering

Other methodologies, such as evolutionary prototyping, depend heavily on continuous user involvement and progressive refinement, while throwaway prototyping emphasizes temporary models that are discarded. Incremental prototyping structures development into independent modules, and agile prototyping prioritizes continuous user collaboration and frequent updates.

Since this project relies on producing a functional model in sequential steps, most of the methodologies do not fit this project. However, the experimental engineering method accommodates this project well. This can be attributed to the fact that this methodology allows the components of this project to be made and tested individually. When combined into the intersection model, it can be used for the simulation and analysis of this system.

3.1 Phases

This project can be divided into five phases: dataset creation, d**ata labeling**, m**odel training**, **real-time detection implementation**, and **simulation**.

In order to train a YOLO model, a dataset must be made. This dataset will consist of images of miniature models of cars, buses, and motorcycles. Different angles and different lighting will be used when capturing the images. As recommended by Kam et al. (2024), by doing this, the model will be more robust.

Objects of interest, such as the vehicles, will be annotated on all the images. Appropriate class assigning, such as car, bus, or motorcycle, and bounding box drawing will be done here. An open-source graphical annotation tool known as LabelImg will be used in order to do the labeling. This tool supports formats compatible with YOLO-based systems.

An 80-15-5 split will be done onto the labeled dataset. This is done in order to divide the dataset such that 80% will be used for training, 15% will be used for validation, and the rest will be used for testing. The model that will be trained using this split will be YOLOv8.

In order to ensure compatibility and optimal performance on the Raspberry Pi, the trained YOLOv8 model will be converted to NCNN format. The Raspberry Pi will serve as the microcontroller for real-time vehicle detection within the physical model.

A simulation made in Python will be developed in order to assess the system's performance. The number of vehicles, with each having its own timer, the simulation simulates with will be provided by the Raspberry Pi. There will be two modes of simulation: Static mode, which will serve as our baseline, as it will simulate normal traffic, and adaptive mode, which will serve as our proposed system simulation, as this mode will take the vehicle count into account when assigning green lights.

3.2 Technologies used

LabelImg is used by researchers for various image annotation needs, this is a powerful and open-source image annotation tool [19]. It enables users to draw bounding boxes around target objects and assign class labels, generating annotation files. This supports multiple formats needed on many machine learning frameworks, which includes YOLO.

The model which this system will use is the YOLOv8. Though YOLOv12 has a higher accuracy and a better architecture [1, 14], YOLOv8 is more hardware efficient and has a more lightweight architecture. Due to this, YOLOv8 performs well on resource-constrained devices [15]. In addition to that, simplified training and enhanced performance on small objects are achieved due to its anchor-free design.

Frequently used in research and prototyping, Raspberry Pi is a compact and versatile single-board computer [12]. The implementation of real-time vehicle detection for this project can be done on the Raspberry Pi due to its processing capability, support for Python, and suitability for computer vision systems. In addition, significantly greater computational power is offered in Raspberry Pi compared to other single-board computers such as Arduino.

Due to the readability, available libraries, and strong compatibility with the simulation workflow and Raspberry Pi, Python is chosen as the project's primary programming language.

4. EXPECTED RESULTS AND DISCUSSION

Although the proposed system has not yet been deployed in a real-world traffic environment at the time of writing, its expected performance can be evaluated through analytical reasoning, system design analysis, and comparison with conventional fixed-time traffic control systems.

Traditional traffic signals employ a fixed-duration approach, assigning equal green time to all directions regardless of actual vehicle density. In highly dynamic environments such as the Sinamalé Bridge corridor, this often results in inefficient use of green time, unnecessary idling, and increased congestion in high-volume lanes.

The proposed system introduces a demand-responsive mechanism by using real-time image data and a trained YOLO-v8 model to detect and count vehicles approaching the intersection. Based on this information, green-signal time is dynamically reallocated to prioritize lanes with higher congestion while maintaining a minimum green threshold for low-density or empty lanes.

From a theoretical and system design perspective, the following improvements are expected:

- Reduced average vehicle waiting time
- Improved traffic throughput
- More efficient use of green signal time
- Reduced queue formation in peak directions
- Better adaptation to sudden traffic surges

In a linear city layout such as Hulhumalé, where traffic often builds in one dominant direction, this adaptive prioritisation becomes particularly beneficial. The system is therefore expected to perform most effectively under peak and semi-peak conditions when traditional methods fail to respond to real-time demand.

However, the performance of the system may be influenced by factors such as weather conditions, lighting, camera angle, and dataset quality. These limitations will be addressed in future system iterations through improved dataset diversity, enhanced camera placement, and possible integration of infrared imaging for low-light conditions.

5. CONCLUSION

This project explored an alternative and more practical approach to managing traffic in linear urban environments such as Hulhumalé, where conventional smart traffic systems are often costly, disruptive to install, and poorly suited to the existing road structure. A camera-based intelligent traffic signal control system was therefore proposed, using computer vision and a YOLO-based model to detect vehicles and adjust signal timings according to real-time traffic conditions.

Through a detailed review of existing sensor technologies, cameras were identified as the most suitable option for the Maldivian context. Unlike embedded sensors, cameras do not require road excavation and can be installed with minimal impact on infrastructure. Likewise, YOLO-based object detection was selected due to its strong balance between speed and accuracy, making it well-suited for real-time applications on resource-constrained devices such as the Raspberry Pi.

Although the system has not yet been tested in a real-world environment, the design, methodology, and simulation framework suggest that it has strong potential to reduce unnecessary waiting time at traffic lights and improve the overall flow of vehicles—particularly during peak hours on the Sinamalé Bridge. This represents a meaningful step toward smarter and more responsive traffic management in Hulhumalé.

Future work will involve deploying the system at an actual intersection, collecting and analysing real traffic data, and comparing its performance with the current fixed-time signal system. Additional improvements may include enhancing low-light detection, strengthening the training dataset, and expanding the system to coordinate multiple intersections. With further development, this approach could contribute to wider smart city initiatives in the Maldives and provide a useful model for other small island cities facing similar traffic challenges.

6. REFERENCE

[1] l Rabbani Alif, M. A. R. A., & Hussain, M. H. (2025). *YOLOV12: A BREAKDOWN OF THE KEY ARCHITECTURAL FEATURES.* Computer Vision and Pattern Recognition. https://doi.org/10.48550/arXiv.2502.14740

[2] Ansariyar, A. (2023). *Real-Time traffic control and safety measures analysis using LiDAR sensor during traffic signal failures*. International Conference on Transportation and Development. https://doi.org/10.20944/preprints202310.1431.v1

[3] Ayaz, S., Khattak, K., Khan, Z., Minallah, N., Khan, M., \& Khan, A. (2022). *Sensing technologies for traffic flow characterization:* From heterogeneous traffic perspective. Istrazivanja I Projektovanja Za Privredu, 20(1), 29--40. https://doi.org/10.5937/jaes0-32627

[4] Bernas, M., Płaczek, B., Korski, W., Loska, P., Smyła, J., & Szymała, P. (2018). A survey and comparison of Low-Cost sensing technologies for road traffic monitoring. *Sensors*, *18*, 3243–3243. https://doi.org/10.3390/s18103243

[5] Bisht, S. B., Joshi, S. J., & Rana, U. R. (2024). Comprehensive Review of R-CNN and its Variant Architectures. In *International Research Journal on Advanced Engineering Hub (IRJAEH)* (e ISSN 2584-2137; Vol. 02, Issue 04, pp. 959–966). https://doi.org/10.47392/IRJAEH.2024.0134

[6] Bugdol, M., Miodonska, Z. E., Krecichwost, M., & Kasperek, P. (2014). Vehicle detection system using magnetic sensors. In *Transport Problems*. Silesian University of Technology, Faculty of Biomedical Engineering. https://www.researchgate.net/publication/287944531

[7] Chaudhuri, A. (2024). Smart traffic management of vehicles using faster R-CNN based deep learning method. *Scientific Reports*, *14*(1), 10357. https://doi.org/10.1038/s41598-024-60596-4

[8] Cherif, B., Ghazzai, H., Alsharoa, A., Besbes, H., & Massoud, Y. (2023). Aerial LiDAR-based 3D object detection and tracking for traffic monitoring. *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–5. https://doi.org/10.1109/iscas46773.2023.10181371

[9] Chiang, C.-Y., Jaber, M., Chai, K. K., & Loo, J. L. (2023). Distributed acoustic sensor systems for vehicle detection and classification. In *VOLUME 11, 2023.* https://doi.org/10.1109/ACCESS.2023.3260780

[10] Chong Chiew Meng, B., Damanhuri, N. S., & Othman, N. A. (2021). Smart traffic light control system using image processing. *IOP Conference Series: Materials Science and Engineering*. https://doi.org/10.1088/1757-899X/1088/1/012021

[11] Dongarwar, T., & Pitale, N. H. (2024). Review-Comparative study on Traffic Actuated Signal System for Traffic Management System. *IOP Conference Series Earth and Environmental Science*, *1326*(1), 012107. https://doi.org/10.1088/1755-1315/1326/1/012107

[12] Ellison Mathe, S., a, Kondaveeti, H. K., b, Vappangi, S., a, Vanambathina, S. D., a, & Kumaravelu, N. K., c. (2024). *A comprehensive review on applications of Raspberry Pi*. Elsevier Inc. https://doi.org/10.1016/j.cosrev.2024.100636

[13] Goyal, R., Elawadhi, O., Sharma, A., Bhutani, M., & Jain, A. (2023). Adaptive Density-Based Traffic Management System utilizing Ultrasonic Units and RODEM Algorithm. In *2023 10th International Conference on Signal Processing and Integrated Networks (SPIN)*. https://doi.org/10.1109/SPIN57001.2023.10116660

[14] Hidayatullah, P., Syakrani, N., Sholahuddin, M. R., Gelar, T., & Tubagus, R. T. (2025). *YOLOV8 to YOLO11: A Comprehensive Architecture In-depth Comparative Review*.

[15] Hussain, M. (2023). YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection. In Sang Do Noh (Ed.), *Machines* (Vol. 11, p. 677). https://doi.org/10.3390/machines11070677

[16] Islam, M. (2025). ANALYSIS OF AI-ENABLED ADAPTIVE TRAFFIC CONTROL SYSTEMS FOR URBAN MOBILITY OPTIMIZATION THROUGH INTELLIGENT ROAD NETWORK MANAGEMENT. *College of Engineering, Lamar University, TX, USA*, *04*(02), 207–232. https://doi.org/10.63125/358pgg63

[17] Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A review of Yolo algorithm developments. *Procedia Computer Science*, *199*, 1066–1073. https://doi.org/10.1016/j.procs.2022.01.135

[18] Kam, Y. Z., Uttraphan, C., Mohd, M. N. H., Khan, F. S., & Lim, C. C. (2024). Vehicle detection and counting for traffic congestion estimation using YOLOV5 and DeepSORT in Smart Traffic Light application. *Online Journal for TVET Practitioners*, *9*(2). https://doi.org/10.30880/ojtp.2024.09.02.021

[19] Ke, H. K., Li, H. L., Wang, B. W., Tang, Q. T., Lee, Y. L., & Yang, C. Y. (2024). Integrations of LabelIMG, You Only Look Once (YOLO), and Open Source Computer Vision Library (OpenCV) for chicken open mouth detection. In *Sensors and Materials*. MYU Tokyo. https://doi.org/10.18494/SAM5108

[20] Keler, A., Kunz, A., Amini, S., & Bogenberger, K. (2023). Calibration of a microscopic traffic simulation in an urban scenario using loop detector data. *SUMO Conference Proceedings*, *4*, 153. https://doi.org/10.52825/scp.v4i.223

[21] Kheder, M. Q., & Mohammed, A. A. (2024). Real-time traffic monitoring system using IoT-aided robotics and deep learning techniques. In *Kuwait Journal of Science*. https://doi.org/10.1016/j.kjs.2023.10.017

[22] Kunekar, P., Narule, Y., Mahajan, R., Mandlapure, S., Mehendale, E., & Meshram, Y. (2024). Traffic Management System Using YOLO Algorithm. *International Conference on Recent Advances in Science and Engineering*, *11*, 210. https://doi.org/10.3390/engproc2023059210

[23] Lin, C.-J., & Jhang, J.-Y. (2022). *Intelligent Traffic-Monitoring System based on YOLO and convolutional fuzzy neural networks*. https://doi.org/10.1109/ACCESS.2022.3147866

[24] Maity, M., Banerjee, S., & Chaudhuri, S. S. (2021). Faster R-CNN and YOLO based Vehicle detection: A Survey. In *Proceedings of the Fifth International Conference on Computing Methodologies and Communication (ICCMC 2021)*. IEEE. https://doi.org/10.1109/ICCMC51019.2021.9418274

[25] Oluwatobi, A. N., Tayo, A. O., Oladele, A. T., & Adesina, G. R. (2021). The design of a vehicle detector and counter system using inductive loop technology. *Procedia Computer Science*, *183*, 493–503. https://doi.org/10.1016/j.procs.2021.02.089

[26] Oweisana, G. C., & Ordua, V. N. (2022). Influence of Traffic Congestion on Psychological Stress and Pro-Social Behaviour among Commuters in Port-Harcourt Metropolis. In Federal College of Education (Tech.), *Educational Psychology, Guidance and Counseling*.

[27] Padrón, J. D., Hernández-Orallo, E., Calafate, C. T., Soler, D., Cano, J., & Manzoni, P. (2023). Realistic traffic model for urban environments based on induction loop data. *Simulation Modelling Practice and Theory*, *125*, 102742. https://doi.org/10.1016/j.simpat.2023.102742

[28] Peza Ramirez, E. P. R. (2024). *Is a smart city a safe city?*

[29] Raheem Sarwar, Naif R. Aljohani, Ibrahim A. Hameed, Muhammad Umair Hassan, Konain Abbas, S. K. A., Usman Ghani Khan, M. U. G. K., & Zhu, J. Z. (2024). Vision based intelligent traffic light management system using Faster R-CNN. In *CAAI Transactions on Intelligence Technology*. https://doi.org/10.1049/cit2.12309

[30] Reddiboina, L. K., Katta, S., Kasireddy, V. S., & Mood, T. (2025). Smart Traffic Light Control System using Ultrasonic Sensors and FPGA. *International Journal of Engineering Research & Technology (IJERT)*. http://www.ijert.org

[31] Rose, A. R., Nisce, I. N., Gonzalez, A. G., Davis, M. D., Uribe, B. U., Carranza, J. V., Flores, J. F., Jia, X. J., Li, B. L., & Jiang, X. J. (2023). *A cloud-based Real-Time traffic monitoring system with LIDAR-based vehicle detection*. 2023 IEEE Green Energy and Smart Systems Conference (IGESSC). https://doi.org/10.1109/IGESSC59090.2023.10321763

[32] Sharma, M., Bansal, A., Kashyap, V., Goyal, P., & Sheikh, Dr. T. H. (2021). Intelligent Traffic Light Control System based on traffic environment using deep learning. *IOP Conference Series: Materials Science and Engineering*, *1022*, 012122. https://doi.org/10.1088/1757-899X/1022/1/012122

[33] Silva, F. O., Paiva, L. P. S., & Carvalho, G. S. (2021). Error analysis of Accelerometer- and Magnetometer-Based stationary alignment. *Sensors*, *21*, 2040. https://doi.org/10.3390/s21062040

[34] Tippannavar, S. S., & D, Y. S. (2023). Real-Time Vehicle Identification for Improving the Traffic Management System-A review. *Journal of Trends in Computer Science and Smart Technology*, *5*(3), 323–342. https://doi.org/10.36548/jtcsst.2023.3.007

[35] Van Den Ende, M., Ferrari, A., Sladen, A., & Richard, C. (2022). Deep Deconvolution for Traffic Analysis with Distributed Acoustic Sensing Data. In *IEEE Transactions on Intelligent Transportation Systems*. https://doi.org/10.1109/TITS.2022.3223084

[36] Ximenes, T. S. D. S., De Oliveira Silva, A. C., De Martino, G. P., Emiliano, W. M., Menzori, M., Meyer, Y. A., & Júnior, V. E. M. (2024). Vehicular traffic flow detection and monitoring for implementation of smart traffic light: A case study for road intersection in Limeira, Brazil. *Future Transportation*, *4*(4), 1388–1401. https://doi.org/10.3390/futuretransp4040067

[37] Ye, Z., Wang, W., Wang, X., Yang, F., Peng, F., Yan, K., Kou, H., & Yuan, A. (2023). Traffic flow and vehicle speed monitoring with the object detection method from the roadside distributed acoustic sensing array. *Frontiers in Earth Science*, *10*. https://doi.org/10.3389/feart.2022.992571