Faculty of Engineering & Applied Science

ENGR 4940U
Capstone System Design I

Design of Smart Traffic Signals for Autonomous Transportation and Connected Communities in Smart Cities

R2: Concept Generation, Conceptual Design and Prototype Report

Group#: 8
Section CRN#: 44900
Advisor: Dr. Hossam Gaber
Coordinator: Dr. Vijay Sood
Due Date: November 8th, 2022

First Name	Last Name	Student Number
Tirth	Patel	100751761
Vatsal	Patel	100728903
Saro	Karimi	100755079

Abstract Executive Summary

Our report includes a table of content that helps the reader, followed by that we have our introduction of the overall project. We have generated many concept designs for our project and have written about each design in detail, how it will support our project overall. After having couples of designs concept ready, we decide to rank it against the requirements that we defined in report 1. After ranking the 4 designs, we came up with an idea to continue with 2 designs that was ranked the highest out of 4 designs concept. After choosing our design, we have specified in more detail, like the result of a function decomposition of the system into components and hence a specification of the components of a system is required. Followed by that, we have defined integration testing of our designs. We also have outlined the estimated cost for our design. We have described a project plan for achieving the deliverables outlined in the handbook that was given to us. Last but the least, we have added a contribution matrix, of how much percentage each person has given in this report.

Acknowledgement

We want to appreciate and sincerely thank our advisor Dr. Hossam Gaber for making this work possible. His guidance accompanies us through every phase of this report-writing. We also want to express our gratitude to our coordinator Dr. Vijay Sood for providing us with some inputs regarding the reports and the overall project.

Table of Contents

List of Tables

List of Figures

Abbreviations

AI	Artificial Intelligence
CPS	Countdown Pedestrian Signal
IoT	Internet of Things
M2M	Machine-to-machine
P2P	Peer-to-peer
STS	Smart Traffic System
SCATS	Sydney Coordinated Adaptive Traffic System
SCOOT	Split Cycle Offset Optimisation Technique
TCS	Traffic Control System
V2I	Vehicle-to-infrastructure
V2P	Vehicle-to-pedestrian
V2V	Vehicle-to-vehicle
V2X	Vehicle-to-everything

1. Introduction

This report presents four conceptual designs for the Smart Traffic System (STS). The first conceptual design explores the ideas about vehicle priority index for each of them. The STS will gather information about each car's destination and assign priority to each lane. The second conceptual design explores video processing to determine density and set timing for each traffic light. If a lane has a higher density, then it will have a longer green signal. The third conceptual design explores the blockchain structure for STS. Each car is viewed as a block in a block chain structure, which uses peer-to-peer protocol to communicate with each other within a network. The last conceptual design explores the idea of machine learning and image processing. Machine learning we can create data analytic software to make many different scheduling plans for traffic signals.

This report is divided into six sections:

- Part I: Concept Generation and Analysis
- Part II: Conceptual System Design
- Part III: Definition of Integration Tests
- Part IV: Estimate Cost of Project
- Part V: Updated Project Plan
- Part VI: Contribution Matrix

2. Concept Generation and Analysis

2.1 Concept Design 1

Concept 1 explores the ideas about vehicle priority index for each of them. As you can see, there are traffic jams where the lights are red, plus the things that look like pan on the road are sensors. So sensors give detections of traffic jams to the main traffic management platform. Then it gives the data to the control application, from there it gives the data to alter traffic lights. After the traffic lights get the data it informs the other car moving that the signal is going to turn red. How the vehicles are going to give signals to traffic signals is through an app that will be implemented in each car and each traffic signal, so it can receive data like information from the vehicle.

Vehicle priority index will be new in smart traffic lights, where although we have count which is there like we have a sensor to count and statistics all those are there, but what is not there is a priority index for each vehicle, so that the signal will be able to include in addition to the account and waiting time the priority index of each vehicle. It is automatically calculated, we will have an algorithm to calculate but we will not depend on the people, we will have our own algorithm to calculate the priority with some trust and some validation. So the priority index is automatically calculated by the system. Only thing the people keep is just information which is validated information like who they are, where they are going, where they are going to drop. So that the typical information fit to the system will calculate the priority endings in real time and that will be used with this system so that is one add value to the current system.

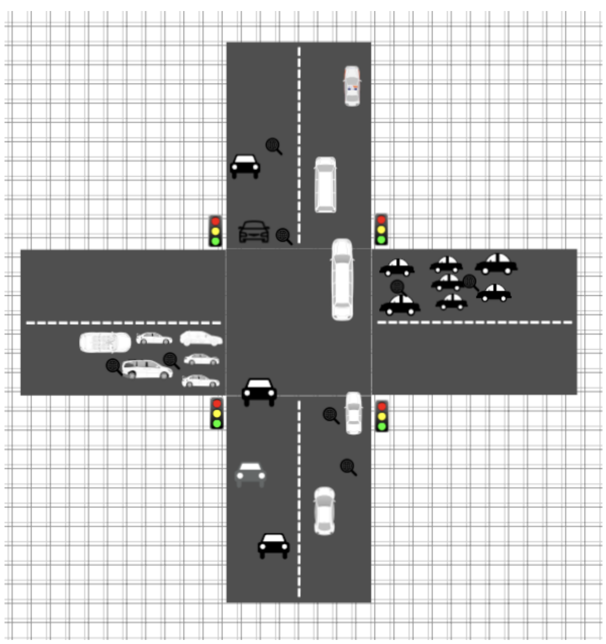


Figure 2.1: Concept Design 1

2.2 Concept Design 2

Concept Design 2 is built upon the idea of using video processing. The design is made up of a number of autonomous cameras, one for each lane being handled. Each camera includes a density checker to determine the number of cars traveling along its designated lanes and can detect vehicles within 200 meters of traffic light. The algorithm for timing the traffic lights depending on the input density for each lane is likewise executed by each camera. The timing algorithm for traffic lights determines what color signal should be shown by a traffic light for how long. An illustration of the design is shown below:

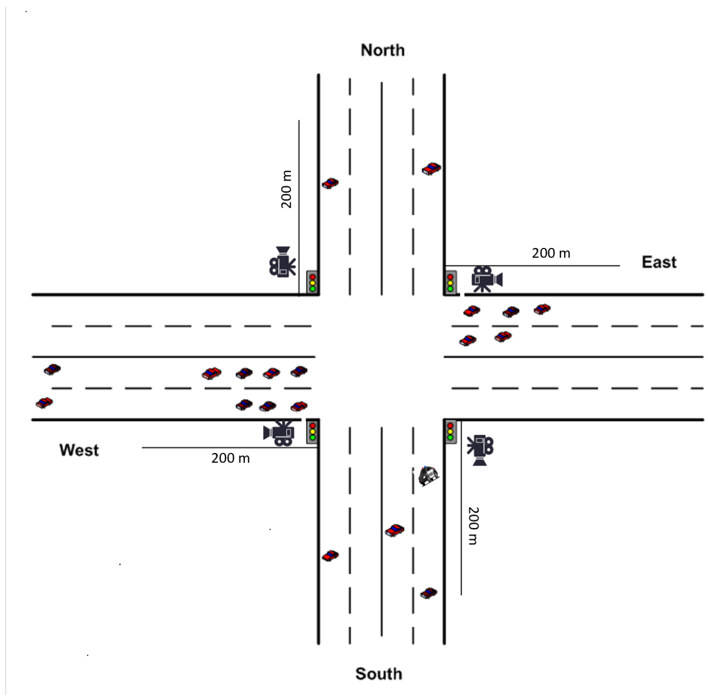


Figure 2.2: Concept Design 3

There are limitations that are include

2.3 Concept Design 3

Concept Design 3 explore the idea of Blockchain Structure

2.3.1 Background

A blockchain is a decentralized ledger of a peer-to-peer network that consists of multiple data sets. These data sets get duplicated and distributed across a large network system consisting of many nodes and create a chain of data packages called blocks where a block comprises multiple transactions. This recording system is hard and impossible to hack and cheat.

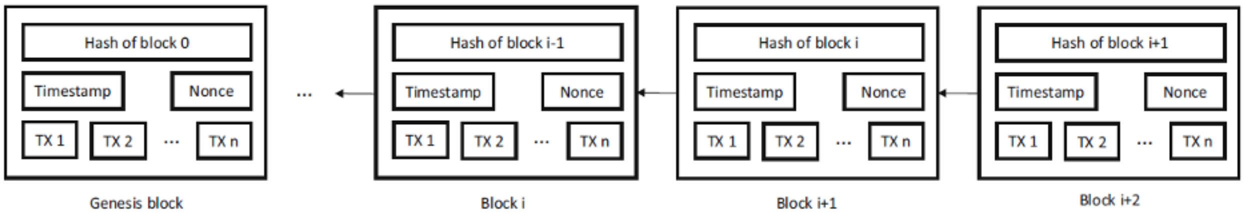


Figure 2.3: Chain of Data

Blockchains use a P2P Protocol which allows nodes to communicate with each other within the network and transfer information about transactions and new blocks[1].

Each block contains:

1. Timestamp.
2. The hash value of the previous block (“parent”).
3. A nonce, which is a random number for verifying the hash.

In blockchain generality, the focus of the entirety of the concept is through processing and analyzing the first block which is called the genesis block. By adding hashing method to the blocks, a more secure connection will be made since:

1. The hash values are unique and infiltration and hacking can be immediately prevented.
2. Changes in a block in the chain would change the corresponding hash value in no time.

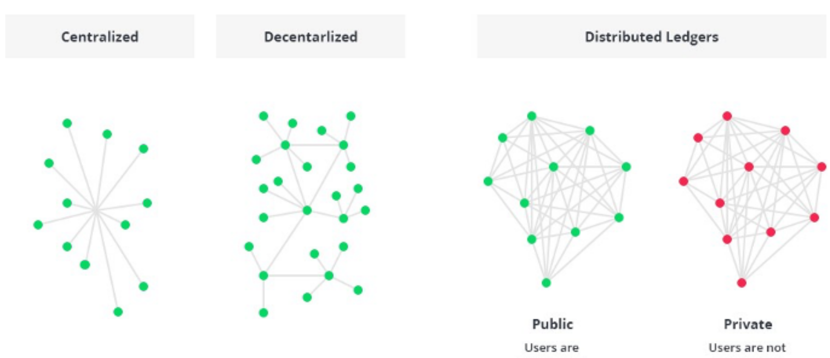


Figure 2.4: Distributed Ledgers

According to Swanson, the blockchain mechanism “is the process in which a majority (or in some cases all) of network validators agree on the state of a ledger. It is a set of rules and procedures that allows maintaining a coherent set of facts between multiple participating nodes”. In our case of creating smart traffic signals, the blocks are the information of the cars and pedestrians that we need to process and analyze for the scheduling system of our traffic lights. The blocks are created by the cars and will contain information about the car including, the speed, location, object detection sensors, distance sensors, etc. The cars will share these data and form a chain of information. The blocks of information will be analyzed and processed and sent the result to the scheduling software of the traffic signals[1].

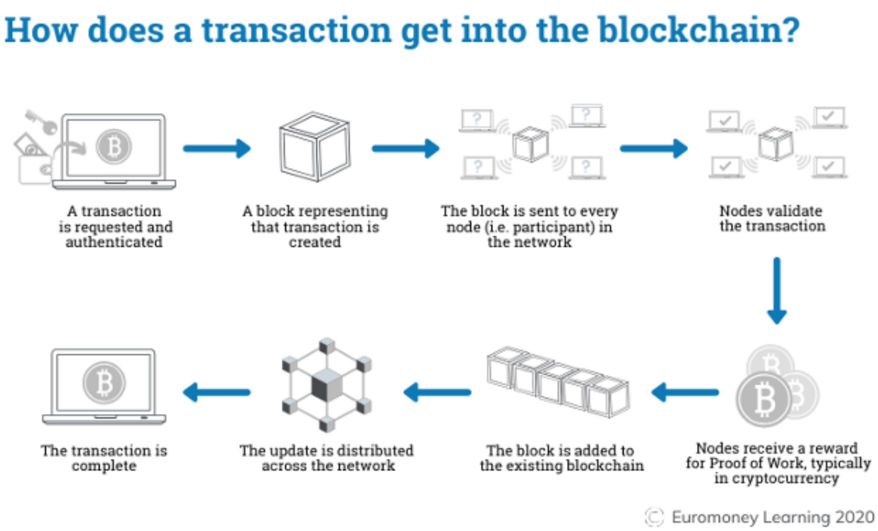


Figure 2.5: Transaction Processing

2.3.2 Helium and LoRa Alliance

Helium is a global blockchain and distributed network which uses hotspots to create public and long-range wireless for LoRa Alliance. This hotspot system compensates the users which install the devices and use them by giving them the native cryptocurrency of the Helium blockchain called HNT. The ability of this system to use the hotspot technology can be used in our system. This device can even work with radio devices which are implemented in all cars. The radio system used in the cars can be used to transfer and receive the blocks of information[2].

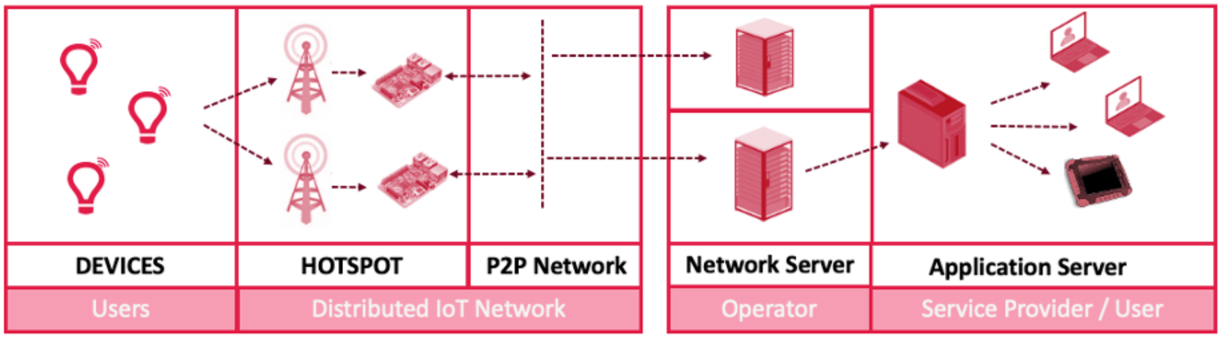


Figure 2.6: Distributed Network

LoRa Alliance is a non-profit organization to help and stimulate the global adoption of the LoRaWAN standard. LoRaWAN is a low-power wide-area end-to-end network architecture. This system is used to connect nodes and devices in regional, national, or even global networks. The system has different layers of protocol architecture (Fig 2.4) with different standards and features to support applications like:

- Internet of Things (IoT)
- Machine-to-machine (M2M)
- Smart city
- Industrial applications

With the low-cost, mobile, and secure bi-directional communication features, this system is the best option to be implemented in our smart traffic signal system[3].

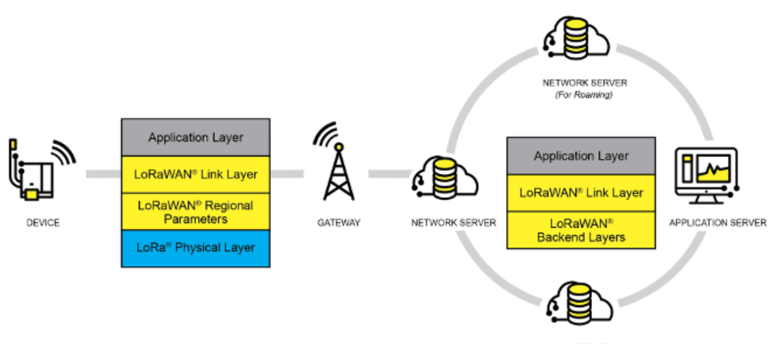


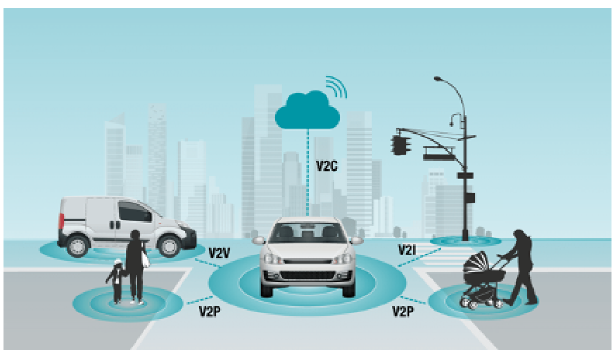
Figure 2.7: LoRaWAN Network Architecture

2.3.3 Connectivity of Cars and Pedestrians

The main aim of this design is to connect all possible nodes via the blockchain. As discussed before we need to create a system that supports V2I, V2V, V2P, and V2X. The vehicle must connect to everything that is using the traffic signal and the signal as well. By using the blockchain method and LoRaWAN system, we can make this connection happen.

Figure 2.8: Connects All Nodes Together

The cars would connect and transfer all of the necessary information for processing and analysis. It is a requirement for the system to have at least 80 to 90 percent of the cars using the LoRaWAN system to implement our algorithm. The pedestrians can be an extra point to have and connected to the main system, but we can have them monitored using a sensor or camera to see, analyze and process their behavior. Moreover, after we have everything connected, the system starts to process everything and creates the best traffic live signal schedule. By doing this, we could easily control the flow of traffic and lower the wait times, since the system has everything it needs to have its best performance[4].



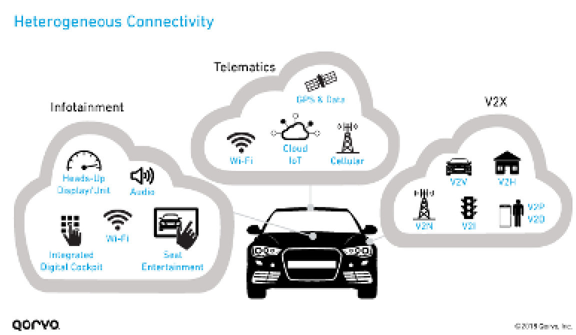


Figure 2.9 and 2.10: Connected and Transferring Information

2.3.4 Exiting Solutions

Some solutions use the same idea of connecting cars, pedestrians, and infrastructure, but our designs are different. In our innovative and new design, we use a blockchain system and LoRaWAN architecture to connect the systems, but the other researched designs use wireless internet connectivity and cellular to connect. In a design called “V2X: Connecting to the Environment,” the system uses two different key standards:

1. IEEE 802.11p: In vehicular environments, to use dedicated short-range communications devices in vehicles and pedestrians, the system uses 802.11p wireless access. This system operates in the 5.9 gigahertz band and covers an approximate range of one kilometer[5].
2. C-V2X LTE: Cellular vehicular-to-everything detects and exchanges information using a low-latency direct transmission. This system uses the low-latency direct transmission in the 5.9 gigahertz in the transportation system band[5].

Many vehicles are equipped with IEEE 802.11p technology and C-V2X is starting to get implemented in the vehicle industry. With this architecture, a large and massive network can be created for our smart cities and can be used to determine the best solution for traffic and traffic signal problems.

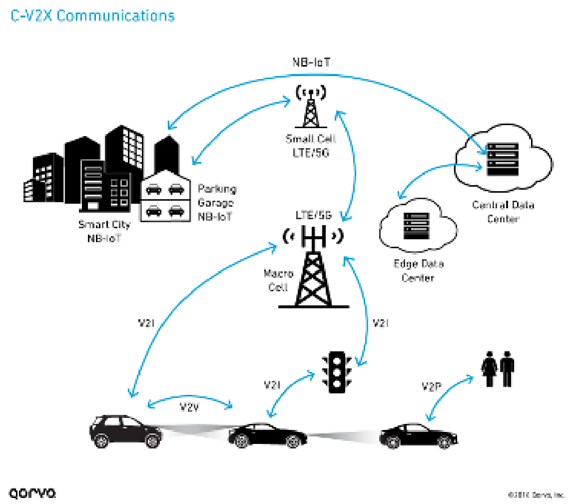


Figure 2.11: Communication

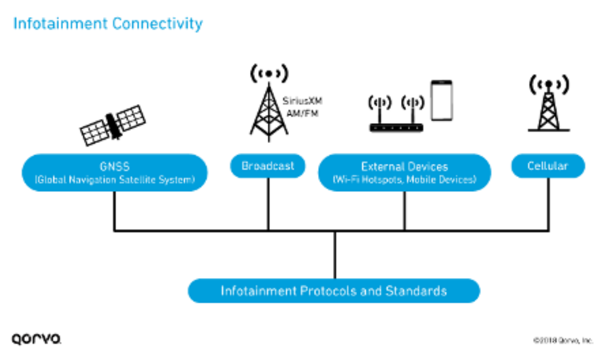


Figure 2.12: Connectivity

2.3.5 Evaluation

This system will be very beneficial to the city, people, environment, and society.

- From a user experience point of view, we would have the easiness of seeing the light status by having the traffic light software implemented in the vehicles so the drivers can see it with no problem and get a warning as fast as possible

Figure 2.13: Traffic light implementation in the vehicles

- The traffic flow would be much better and the waiting times would be lower in the city.

- By having a better traffic flow and lower waiting times, the usage of gas and energy would be lower since the idling time of the cars is lowered and the drivers get to their destination faster so there will be less usage of the heater or cooler and that saves the energy.
- Accidents will be much lower since the mechanism of the design knows all of the cars' information like position and speed and evaluating these parameters would warn the drivers of a possible cause of accidents if they continue driving with the same behaviors.

From a technological and engineering point of view, since we are using the blockchain system, our design is much faster, more secure, and better in performance. The other systems would have security problems since they use an open connection with no security and they need to connect to a bigger software to process and analyze the data. Since our system is using a blockchain architecture and we decided to use the LoRaWAN application, we could run the software and algorithm of our system on the distributed system without getting help from an outside resource, a bigger computer, or even the internet.

Figure 2.14: LoRaWAN application

2.4 Concept Design 4

Concept Design 4 explores the idea of machine learning and image processing

2.4.1 Background

Machine learning is a subcategory of Artificial Intelligence (AI), that is used in computers and systems to give the ability to learn in the system without any need for developing the systems furthermore. Examples of machine learning applications are chatbots, predictive text, language translation apps, suggesting systems, and social media feeds presentation uniquely for you[6].

The goal of creating AI and machine learning systems is to implement intelligent behavior of humans at a computerized level which is giving the ability to recognize tasks, understand tasks, and perform tasks[6].

Traditional programming and developing an application; require detailed instructions and algorithms, so the program can follow them and perform a task. This method is good for doing processes that are not required learning and it can simply do it without the need of performing large-scale processing. But in some cases like image detection, giving the instructions to the system will not be good enough since it takes a lot of time and processing for the system to just figure out if a picture of a human is human or not. These cases need machine learning implementation to perform the task in a shorter time[6].

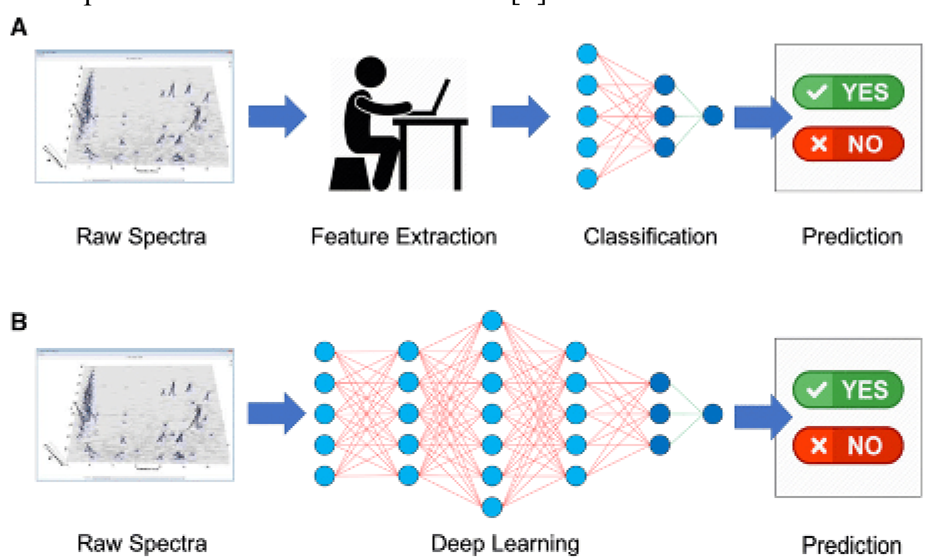


Figure 2.15: Machine Learning Implementation

By applying machine learning to image detection, we can use a method called image processing to detect and process images faster and with less processing time. The ability to learn in the system and pictures gives many advantages. Examples of image processing can be seen in MRI technologies for detecting tumors and abnormalities in the human body. There are many deep learning methods to use image processing using Keras, OpenCV, TensorFlow, etc[7].

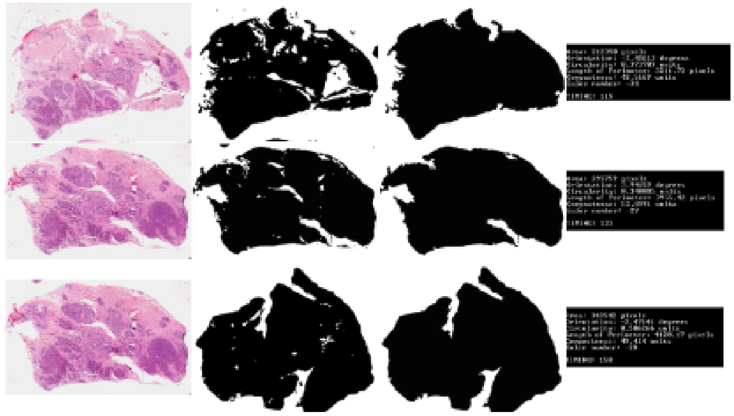


Figure 2.16: Tumor detection using OpenCV

2.4.2 Image Processing in traffic detection

In the automotive industry, we have seen much use of image processing. Creating these systems has helped to achieve autonomous driving, traffic safety, lane recognition for lane keeping, etc. By adding more value to cameras and image processing, we can use them to control traffic flows and detect them and keep the intersections organized. Implementing image processing into traffic signals will be a huge step toward optimizing traffic flow. By using OpenCV, we can detect the cars, their make, model, and even their speed. Below is an example of an OpenCV program and car detection system using jitendrasb24 resources in GitHub[7][8].

```python
#import libraries of python opencv
import cv2
# capture video/ video path
cap = cv2.VideoCapture('cars.mp4')
#use trained cars XML classifiers
car_cascade = cv2.CascadeClassifier('haarcascade_cars.xml')
#read until video is completed
while True:
    #capture frame by frame
    ret, frame = cap.read()
    #convert video into gray scale of each frames
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    #detect cars in the video
    cars = car_cascade.detectMultiScale(gray, 1.1, 3)
    #cv2.im_write(cars)

    #to draw a rectangle in each cars
    for (x,y,w,h) in cars:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2)
        cv2.imshow('video', frame)
        crop_img = frame[y:y+h,x:x+w]

     #press Q on keyboard to exit
    if cv2.waitKey(25) & 0xFF == ord('q'):
        break
#release the video-capture object
cap.release()
#close all the frames
cv2.destroyAllWindows()
```

Figure 2.17: Car detection program using OpenCV

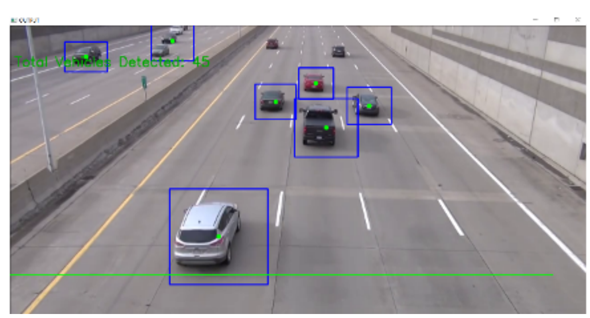


Figure 2.18: Car detection program result

As you can see, we can detect the cars and get their position and amount, and if we want to we can get more information. These data can be used in implementing a smart traffic signal.

2.4.3 Smart Scheduling

Image processing is one of the methods of machine learning used in our design. Image processing is used to gather the data and process and analyze some data gathered from the cameras like detecting crashes and accidents and warning the drivers and pedestrians, but most of the data gathered is not used and is left alone. By using another ability of machine learning we can create data analytic software to make many different scheduling plans for traffic signals. In Machine Learning, Data Analysis is the process of inspecting, cleansing, transforming, and modeling data to reach the goal of creating useful information through decision-making[9][10].

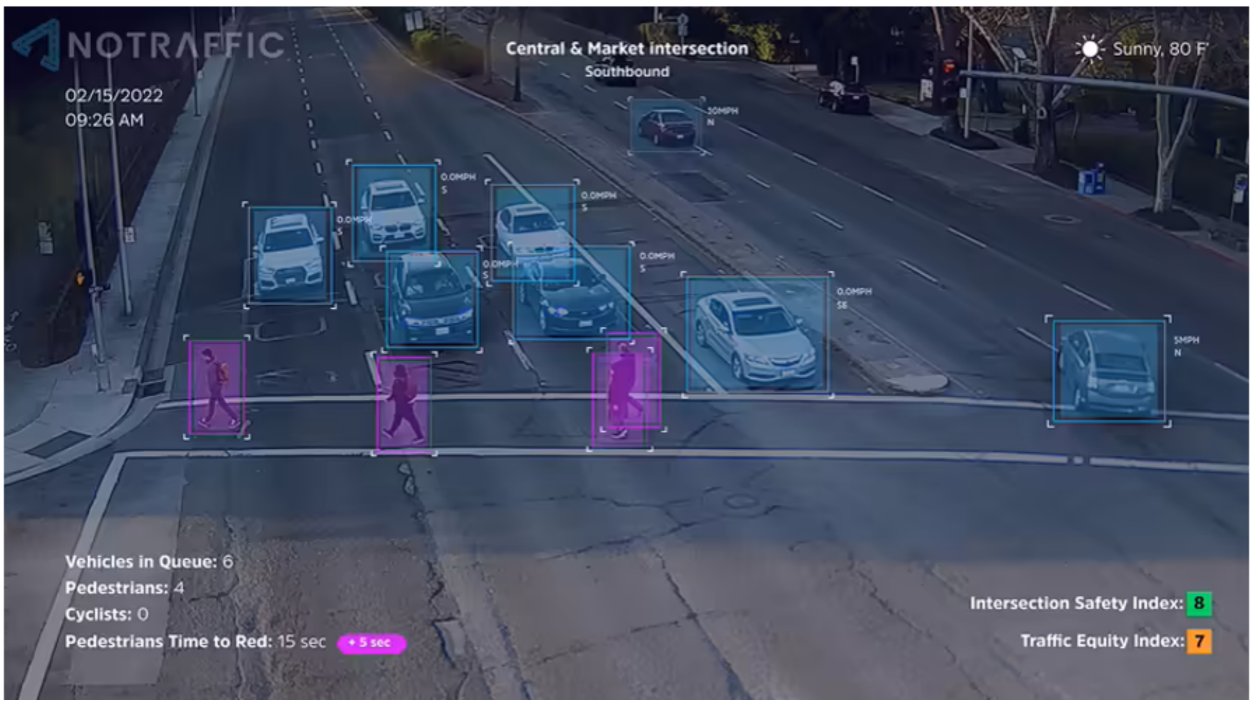


Figure 2.19: Result of Image Processing and Data Analysis using machine learning

2.4.4 Existing solutions

There are systems using the same idea we are trying to implement but in another method. SCATS, SCOOT, and TCS are using the best way to create a smart traffic signal using machine learning.

1. SCATS: An intelligent real-time traffic system that monitors, controls, and optimizes the movement, traffic, and timing of cars and pedestrians to give the best traffic timing. The information it gathers per each traffic signal gets analyzed with the SCATS algorithm and then it'll be executed, then will be sent to the SCATS organization for further analysis[11][12].

2. SCOOT: An adaptive traffic control system that determines the traffic timing plans based on the gathered data of its sensors, pedestrians, and cars for each region differently. Calculating this information and data, it creates the best traffic timing for the located region[11].
3. TCS: Trans Suite is a hybrid traffic control system. Uses second-by-second communication to monitor signal operations. Maintain signal coordination with Field equipment strategy. Does not directly control signal movements, but commands each intersection controller to follow the timing plan that resides within its local database. It verifies that the controller adheres to the commanded timing plan. It can accommodate up to 256 signal timing plans[11].

The one that is closest to our design is SCATS, which uses an intelligent system to create the best possible traffic flow. This technology is capable of ingesting data from sensors into an intelligent system and decision-making engines which uses machine learning to optimize service delivery and network management[12].

2.4.5 Evaluation

As statistics show in other similar programs and designs, we would have more than:

- 28% Reduction in travel time,
- 25% Reduction in stops,
- 12% Reduction in fuel consumption,
- 15% Reduction in emissions

By using a machine learning implemented system. Moreover, the cost of implementing this system is lower than using many expensive sensors, since the traffic signal will only use cameras to detect the traffic and machine learning software to give out the outcome of the processing and analysis[12].

2.5 Sensitivity analysis and ranking of the models against the requirements.

9 point rating scale are as follow:
1-poor
3-Weak
5-Good
7-Very Good
9-Exceptional

		Concept Designs			
Points out of 100	Requirements	Concept Design 1	Concept Design 2	Concept Design 3	Concept Design 4
15	Function after a power outage.	3	5	9	5
15	Detect vehicles/pedestrians arriving at an intersection	7	7	9	7
12	Provide efficient traffic signal timing	7	7	9	7
13	Include CPS indicators that show how many seconds are left in the interval for pedestrian change.	3	5	5	5
15	Reduce waiting time at the traffic lights.	5	7	9	7
10	Higher priority for transit vehicles than normal vehicles.	5	7	7	7
12	Reduce maintenance cost	5	3	7	7
8	Return the status of the system and any fault data to the control center.	3	3	9	7
100	Total Points	38	44	64	52
Rank		4	3	1	2
Which design/s should we continue with?		NO	NO	YES	YES

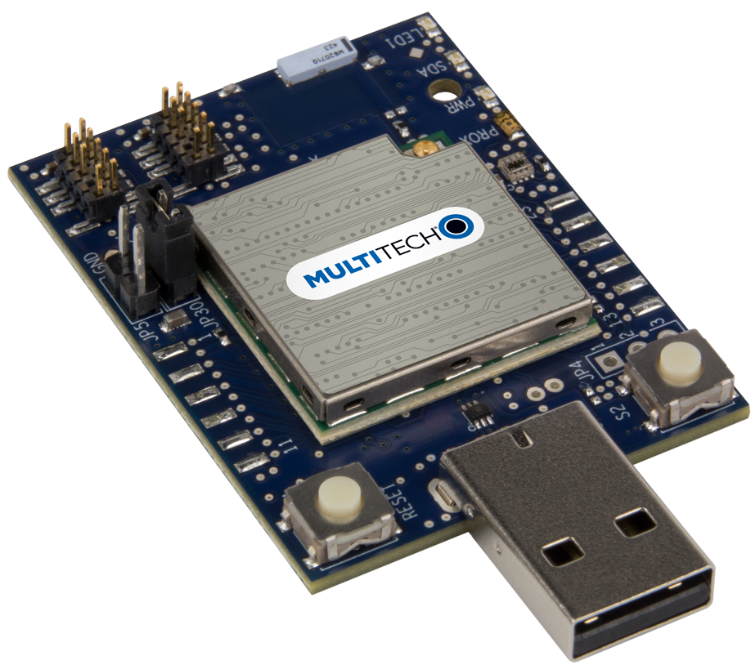
Table 2.1 Concept Screening Matrix Table

Thus, illustrated in Table 2.1 Concept Design 3 and 4 have the highest rating from the 4 concept designs in their total point scores, so the team have come up with the idea of using both the designs by connecting it to each other.

3. Conceptual System Design

3.1 Connecting LoRaWAN devices

The LoRaWAN technology uses an embedded system for implementing its software which is using the Helium blockchain software to run. We can easily buy the devices we need from the LoRa Alliance company or purchase it from Helium itself. There is another method we can follow without purchasing their devices. Since we want to exchange information and install the software on this massively distributed system, a self-made embedded system can work easily. By



creating our embedded system or using cheap ones in the market, we can go on to the next steps of our design.

Figure 3.1: MultiTech xDot Micro Developer Kit

The system we talked about is for vehicles that do not have access to a device to connect. Nowadays cars have more complicated computers and integrated embedded systems in their design and can easily be reprogrammed. Since we have this option, we can install the LoRaWAN software on the vehicles directly and then use the technology to connect to the other cars.

Another problem we have is the connection we need between the cars in the concept design. We explained the other designs that use cellular and IEEE 802.11p wireless connections. But these communication types are not good to choose from since they may encounter many problems including, losing connection, and the complexity of the connection between the systems may slow down the communication timings. Since we are using Helium and LoRaWAN, we can use their new technology for hosting which is using radio-based connections. Every car is equipped with a radio connection and can receive and transmit data. We can use an unused frequency to transmit the data we want between the cars.

For connecting the pedestrians, we have to come up with a new way since humans are not equipped with any radio system or embedded systems. Although, we can use their phones to connect the process will be a big one to do. In this part, we will use Meshed technology which uses the LoRaWAN method to connect the pedestrians. Meshed is an Australian-based LoRaWAN IoT integration company delivering private and public LoRaWAN solutions supported by devices and applications. Meshed technology is being used in many industries as well as smart cities and people counting. By using their technology, we will be able to connect pedestrians to vehicles and infrastructure without giving all of them a device to hold so they could cross a street.

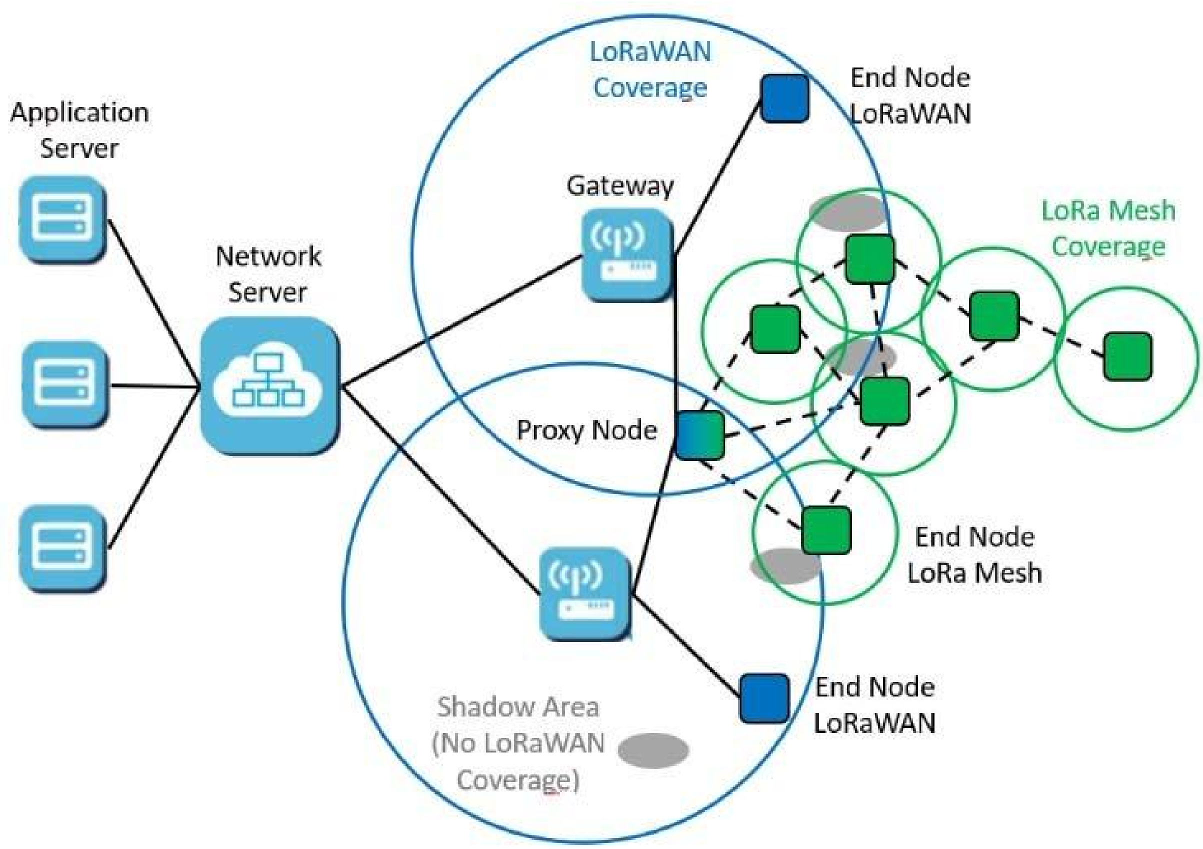


Figure 3.2: Hybrid LoRaWAN and Meshed Network

3.2 Implementation of tMachine learning and LoRaWAN

The system we are going to design will have software running on the LoRaWAN system. The distributed system created with a big processing capability will be a good base for installing our machine learning system including the image processing and the data analytics program. The AI will be working on the system using the data it gathers from the cars using the blockchain to process the data and give the best result for each traffic light.

The algorithm is a bit more complex to explain in this step of the design but we will explain the fundamentals of it. The cameras on each intersection will gather the data of the cars and pedestrians and send it to the distributed system network. Each intersection will have an unique ID for identification in the system. Around five intersections horizontally and vertically will be communicating to each other to perform the best output for the traffic flow. The AI will be analyzing the data it gathered within the five by five intersections and then gives out a plan for them to run it on their system and change the status of the light based on them.

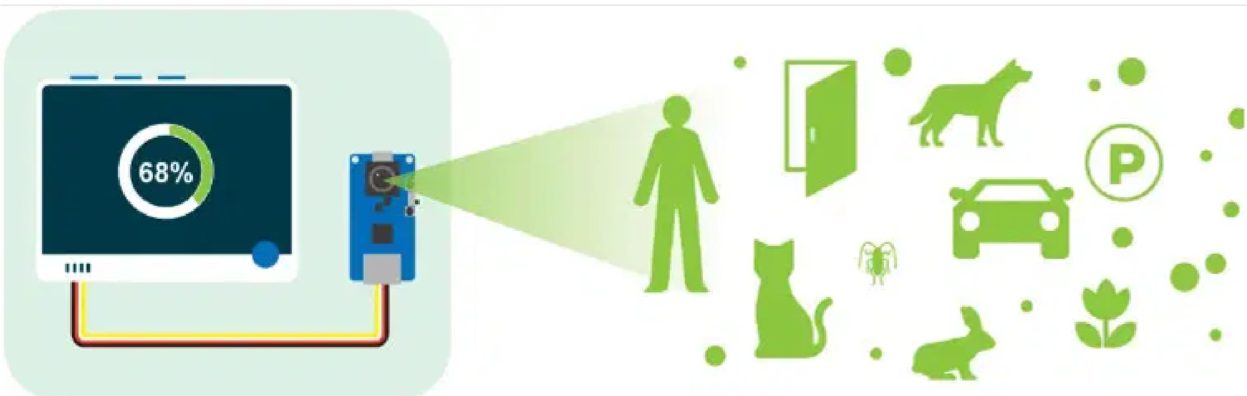


Figure 3.3 AI analyzing the data and giving out an output

4. Definition of Integration Tests

Test Case ID	Test Case Objective	Test Case Description	Expected Result
1	Check if the car is connected to the STS	When a car approaches an intersection it should be connected to STS.	Display the traffic signal status to the car driver on the software.
2	Check if a pedestrian is connected to the STS	When a pedestrian approaches an intersection, the STS should be connected to mobile device of the pedestrian	The STS should start a countdown pedestrian signal.

Table 4.1: Integration Testing

5. Estimated Cost of the project

Estimated cost of the project will be divided into two sections. Section one will include the rough cost estimation of the prototype/simulation. Section two will include rough cost estimation for the prototype/simulation to be scaled to the real world.

5.1 Estimated Cost of Prototype/Simulation

We will be using Carla and PTV Vissim software to simulate our concept design. Both of these software are free to use so there will be no costs associated with the software.

5.2 Estimated Cost of Real World

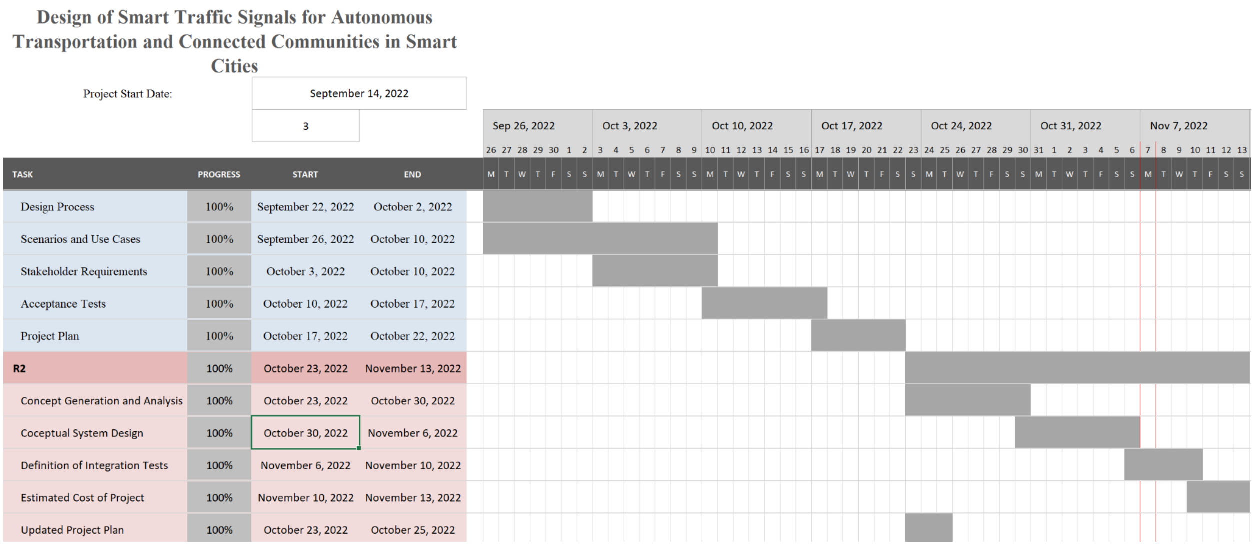
To implement our design in a real world environment, there will be several costs associated with acquiring the hardware and developing a software. The % cost estimation for each process in project development is shown in the table below.

Project Phase	% Cost Estimation
Development	45%
Management	10%
Quality Assurance	20%
Business Analysis	15%
UI Design	10%

Table 5.1: Project Cost Estimation

Furthermore, we estimate that the project will be completed within 4-6 months and the cost would be roughly $2,500.00 - $10,0000.00.

6. Updated Project Plan



Design of Smart Traffic Signals for Autonomous Transportation and Connected Communities in Smart Cities

Project Start Date: September 14, 2022

3

TASK	PROGRESS	START	END
Design Process	100%	September 22, 2022	October 2, 2022
Scenarios and Use Cases	100%	September 26, 2022	October 10, 2022
Stakeholder Requirements	100%	October 3, 2022	October 10, 2022
Acceptance Tests	100%	October 10, 2022	October 17, 2022
Project Plan	100%	October 17, 2022	October 22, 2022
R2	100%	October 23, 2022	November 13, 2022
Concept Generation and Analysis	100%	October 23, 2022	October 30, 2022
Coceptual System Design	100%	October 30, 2022	November 6, 2022
Definition of Integration Tests	100%	November 6, 2022	November 10, 2022
Estimated Cost of Project	100%	November 10, 2022	November 13, 2022
Updated Project Plan	100%	October 23, 2022	October 25, 2022

Figure 6.1: Project Plan

7. Contribution Matrix

Table 7.1: Summary of work % contributed by each team member for report 2

Task	People		
Report 2	Saro Karimi	Tirth Patel	Vatsal Patel
	33.33%	33.33%	33.33%

8. References

[1] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, *http://cs.unibo.it/~danilo.montesi/CBD/Articoli/2017Blockchain.pdf*, Mar. 2017.
[2] “Home: Helium documentation,” *home | Helium Documentation*, 2018. [Online]. Available: https://docs.helium.com/.
[3] “Lorawan for developers,” *LoRa Alliance®*, 29-Sep-2021. [Online]. Available: https://lora-alliance.org/lorawan-for-developers/.
[4] “What is lorawan® specification,” *LoRa Alliance®*, 25-Jun-2022. [Online]. Available: https://lora-alliance.org/about-lorawan/.
[5] *Qorvo*, 31-Oct-2018. [Online]. Available: https://www.qorvo.com/design-hub/blog/v2x-in-the-connected-car-of-the-future.
[6] S. Brown, *Machine learning, explained*, Apr. 2021.
[7] N. Maan, “Vision and image processing in automotive Adas/AD Development,” *LinkedIn*, 23-May-2021. [Online]. Available: https://www.linkedin.com/pulse/vision-image-processing-automotive-adasad-development-nakul-maan/.
[8] J. Singh, “JITENDRASB24/car-detection-opencv: Detection of cars from video, photo and Real Time,” *GitHub*, 06-Aug-2021. [Online]. Available: https://github.com/jitendrasb24/Car-Detection-OpenCV.
[9] P. Dharshini, *Must-Know Statistical Data Analysis Techniques in Machine Learning!*, Jun. 2021.
[10] M. Paterson, “Machine learning gives Santa Cruz traffic the Green Light,” *Cloud Brigade*, 24-Jan-2022. [Online]. Available: https://www.cloudbrigade.com/smart-traffic-using-machine-learning/.
[11] City of Toronto, “Types of traffic signal systems,” *City of Toronto*, 29-Aug-2022. [Online]. Available: https://www.toronto.ca/services-payments/streets-parking-transportation/traffic-management/traffic-signals-street-signs/traffic-signals-in-toronto/types-of-traffic-signal-systems/.
[12] “Scats,” *Home*. [Online]. Available: https://www.scats.nsw.gov.au/home.