ÇANKAYA UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

Project Report v0.2

Jan 7, 2022

Smart Traffic Systems Project

Mete Korucu

Öykü Yatıkkaya

Oğuzcan Baldemir

Baran Azak

Kadir Gökalp

TABLE OF CONTENTS

Abstract

These days, people waste most of their time waiting for traffic which this time could rather be used properly.Most cities all around the world, traffic is mostly just managed by traffic police and signal lights that contribute to the long wait we mostly have because of their inefficient configuration. Also, most of the time the traffic lights use fixed cycle protocol. This is an example for an inefficient configuration as well. That’s why there is a need to automate and improve these traffic systems. The mixed work of machine learning, artificial intelligence and computer vision can be used and integrated in the traffic system to get rid of long hours spent waiting in traffic. In addition, the system will identify the cars and their amount that are stuck under the red light for a longer time than the other cars.

Özet

Günümüzde insanlar her yıl yaklaşık 40 saatini trafikte boşa harcıyor. Şehirlerde trafik yönetimi çoğunlukla trafik polisi ve verimsiz bir konfigürasyon nedeniyle araçların uzun süre beklemesine neden olan sinyal lambaları tarafından yönetilir. Ayrıca, birçok kentsel alanda, trafik sinyal ışıklarının çoğu, verimsiz konfigürasyonun bir nedeni olan sabit bir döngü protokolüne dayanmaktadır. Bu yüzden bu trafik sistemlerini otomatikleştirmeye ve iyileştirmeye ihtiyaç vardır. Trafik ışıklarının zamanlamaları, trafik sıkışıklığı vb. gibi sorunları çözebilecek ölçeklenebilir bir trafik sistemi geliştirmek için makine öğrenimi, yapay zeka (AI) ve benzeri özellikler kullanılabilir. Buna ek olarak, sistem araçları tanımlayacak ve diğer araçlara göre daha uzun süre kırmızı ışıkta kalan araç miktarını hesaplayabilecek. Makine öğrenimi, AI ve Görsel tanıma tabanlı teknolojiler, tüm bu parametreleri kapsama potansiyeline sahiptir. Bu makale, bu teknolojilerin nasıl uygulanabileceği ve karşılaşacağımız zorluklar hakkında bir fikir vermektedir.

1.Introduction

The main goal of the project that we are making is to change people’s lives for good purposes and make their lives easier. The population grows every single day. Therefore, smart traffic control can be a big problem. To find an answer to this problem, a lot of researchers' groups have done lots of studies. We also have the problem that some green lights occur for a very short amount of time and if there is a traffic jam, the same car might wait in the same red light for a long time. To prevent this problem, we can use computer vision technology to detect the cars that are stuck under the same light more than once and reduce the time of that red light the second time it occurs. By doing this, we give the chance to those cars which are stuck under the same light more than once. Thus, detecting all vehicles (cars, trucks, motorcycles etc.) is a very common problem. In this project, we want to detect the vehicles and also the count of them. By doing this, the system will decide the time of the traffic lights to occur and which road has the most traffic jams in that area.

There will be some cameras which will be put in the street and from the top, the cameras will detect the vehicles and then give feedback to the system. Overall, this system will be a system that can identify all crowded or empty streets, vehicles, vehicles’ speeds.

1.1. Previous Works Of Smart Traffic Systems

The problem of traffic road management has been an active problem for countless years. Many attempts to propose a solution for this problem of intelligent traffic control have been found in the past.

For example, in [4], the design of Intelligent Traffic Light Controller Using Embedded System is developed. They utilize simple means to count the quantity of vehicles. In [5], It is proposed to create an intelligent traffic light system to prevent traffic accidents. In [6], an intelligent traffic light control method for crossroads based on extension theory is presented. An Artificial Intelligent (AI) Approach for Intelligent Traffic-Light Control is proposed in [7].

The most common method is to install a traffic signal at each road intersection to control access to the critical section (the junction). Several variants of priority access and critical sections are implemented. The round tape, for example, assigns a constant time for the open road (by turning on the green light), then a fixed time for the transition period (by turning on the yellow light), then closes the road (by turning on the red light), and repeats the process for the next street, and so on. Except in a few developed countries, this is the least efficient method of traffic management; however, it is the most widely used system in the world.

In [8] an algorithm for managing the operation of a single traditional traffic light signal for an intersection with four-lane roads is presented, which proposes an adaptation of the traffic to the traffic conditions. Although it is stated that the proposed algorithm is adaptive, it takes into account fixed time periods similar to the study presented in [9]. Similarly, a calendar-based approach for progress reporting is proposed in [10].

[11] presents another study that uses multi-agent communication based on edge computing architecture and IoT for traffic light control. For global traffic sign management, the authors propose a multi-agent reinforcement learning system (MARL). Similar MARL-based studies are presented in [12] and [13]. Other agent-based approaches are given in [7,14]. In [15] to use Q Learning to maximize the number of vehicles crossing intersections. Similarly, [16] proposes deep reinforcement learning. In [17] the approach to the optimization of ant colonies is proposed and in [18] the approach to the optimization of artificial bee colonies is proposed. The social IoV proposed for traffic management [19] is shown in [20].

Several studies are based on the amount of traffic on the roads. In [21], for example, an adaptive algorithm is presented and evaluated. The goal of this study is to use V2V so that each vehicle can estimate the level of traffic congestion and redirect to the least congested route. [22] proposes yet another study based on V2V communication. [23] provides a traffic optimization framework based on vehicle redirection to reduce traffic congestion. Another study in [24] proposes a system based on V2I communications. Furthermore, this study takes into account the protection of incident detection as well as the spread of various types of attacks. Several experiments for the control of floating data-based traffic signs (FCD) are reported in [25]. FCD has also been used in [26] for vehicle tracking data management techniques.

[27] proposes the Modular Timed Synchronized Petri Net model for traffic sign management in order to reduce environmental impact. [28] presents the maximum Pareto flow algorithm, and [29] proposes the cell genetics algorithm. Several research articles have been published on the use of cameras to count the number of vehicles for traffic management and optimization. [30] presents a recent study on the use of smart cameras on the Internet. The solution is based on WSN (Wireless Sensor Network) and VANET (Vehicular Ad-hoc Network) by connecting a large number of cameras in a dedicated infrastructure. The video feed from the cameras is routed to centralized servers for processing and extraction of useful traffic data that can be used to check traffic signs. [31, 32] provide additional studies based on the WSN. As shown in [33, 34], several IoT (Internet of Things)-based strategies are proposed. The authors of [35] propose using expert systems and artificial intelligence to process images extracted from camera systems for traffic management. [13] proposes a pheromone-based multi-agent system that is based on cameras and sensors.

[36] presents an interesting study that selects the best charging station for electric vehicles based on traffic conditions in order to minimize travel time. [37] proposes a parallel algorithm for synchronizing intersections in large and dense areas, with the goal of improving Bus Rapid Transit based on average speed. [38] presents a similar study based on a hybrid heuristic approach. Finally, the authors of [39] propose that vehicle speed at the intersection be used as an optimization parameter for traffic light control.

TABLE I

RECENT ACADEMIC STUDIES IN VEHICLE DETECTION AND TRAFFIC LIGHT SYSTEM

Paper	**Year**	**Method**	**Feature**	**Classification**	**Tracking**
[40]	2018	Deep Convolutional Neural Networks	Colour point	Pixel calculation	Space tracking
[41]	2016	Computer vision techniques, including thresholding, hole filling and adaptive morphology operations	Grayscale comparison	Gaussian mixture	Reference line model
[42]	2015	Probabilistic prior maps and dense Hog	Colour	Majority pixel count	Hidden markov model
[43]	2014	Color thresholding, BLOB analysis	HoG	Support vector machine	Correlation tracking

[44]	2012	BLOB analysis	2D Gabor wavelet	Nearest neighbour	-
[45]	2012	Color difference enhancement, neighborhood image filling, radial symmetry	Colour	Colour	Spatial-temporal consistency check
[46]	2011	Prior knowledge based state detection of vehicle	LBP feature	Support vector machine	-
[47]	2010	Color thresholding, morphological operations	Haar feature	AdaBoost trained classifier	CAMSHIFT
[48]	2009	Gaussian-distributed classifier, BLOB analysis, temporal information	Colour	Global contradiction solving scheme	Temporal filtering / decision scheme

1.2. Proposed System and Challenges

To make a smart traffic system, we are building a new system using artificial intelligence and CCTV(Camera-Real time video) image processing together. Cameras mounted on every place that have traffic lights record thousands of hours of video daily, which contain very useful information and can be used to reduce the overcrowding on the streets. With the help of the video footage data, data set of traffic videos and machine learning algorithms, it is possible to manage traffic light duration. Pattern recognition can be used to identify the vehicle type, count number of vehicles, and categorize lightweight and heavy vehicles. This would improve the system to control the traffic lights to decrease the traffic. The machine learning algorithm can perform better as the training data sets contain enough data to predict. These data sets are growing every day which can help our system to give suggestions about improving roads like “Build a bridge here” etc. Sometimes during the overcrowding situations, managing traffic light duration is not enough. So we are building a suggestion AI for improving roads . Applying machines to do this job can be very effective and reliable. The proposed system will be a synchronized intelligent traffic light which will improve overcrowding. The proposed model will be composed of two components: a monitoring system and a control system. Both components are able to integrate together and use data of the same directional backward signals simultaneously to make intelligent decisions efficiently. Figure shows the proposed model architecture.

2. Literature Search

2.1. Computer Vision and Machine Learning

Computer Vision is the field of computing with technologies that allow computers to identify and process preferences like humans. From dying to being processed, unintended, planning to achieve goals and aiming for them. The image in question may be the result of video, videos from multiple cameras of the same type, acquired with a 3D scanner, or enhanced devices such as an ultrasound device.Computer Vision studies first started in the 1950s. Commercially, it was first used in an application that perceived the difference between printing and handwriting in the 1970s. The real development in the field of Computer Vision has taken place in recent years. Both the rapid increase in the processing power of computers and the widespread use of the internet have resulted in a significant increase in the visual data we have.

Most of the machines we still use don't use image recognition and AI. These days we have self-driving cars that use computer vision which lets us not use the car manually. If they didnt use these technologies, they wouldn't know the differences between a person and a tree. And wouldn't know where they should drive and where they should avoid going. these technologies such as computer vision lets the system identify objects and people and relations of the environment. Computers learn the difference between these objects and people from image datasets, using them as training data and they design a model which helps the computer to predict the outcome. These datasets have huge amounts of data that can help the computer learn and train itself in time. about 2 billion photos are shared by users every day. And these images are used for prediction and training the models that we use to predict. The traffic management system has CCTV’s and sensors that helps the system see, identify and make momentary decisions. These sensors could be used to identify light weight/high weight vehicles and the number of the vehicles that are waiting on the road or just passing by. This knowledge can help us determine the times where there will be a high possibility of traffic.

Over the past several decades, the field of computing developed enormously. Earlier, everything was about logic and mathematical problems. This was an obstacle for solving real life intricacies. This was why we needed improved systems, to learn on its own and take decisions at the real time on its own. Machine learning varies based on the methods used. These methods are categorized as supervised, unsupervised, semi-supervised, reinforcement machine learning algorithms. Supervised machine learning algorithms can be used when we have learned in the past and we apply it to a new data using labeled examples to predict events of the future. Unsupervised machine learning ae mostly used when the information we have informations that are neither classified or labeled for training. Semi supervised machine learning method is mostly chosen when the labeled data needs skilled or relevant resources to train and learn. Reinforcement machine is a method that şnteracts with the environment by producing actions and discovers errors or rewords.

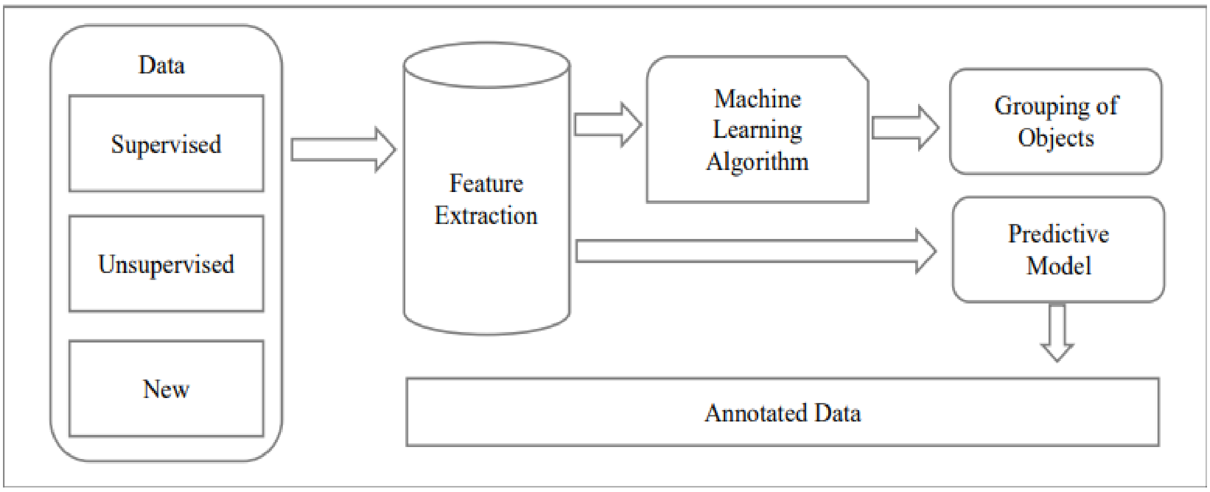


Fig.1 Workflow of Machine Learning

Machine learning is a branch of artificial intelligence that can create machines that can learn from their surroundings, people, and mistakes. Machine learning systems include recommendation systems, autopilot systems for planes, computer-aided translation of one language to another, and maps.

3. Software Requirements Specification

3.1. INTRODUCTION

The following subsections are an overview of the entire Software Requirements Specification (SRS) document.

3.1.1 Purpose of the System

The purpose of this document is to present a detailed description of the Smart Traffic Systems Project system to the Users who will work with this system. It will explain what the system will do, the purpose and features of the system, the interfaces of the system. This document will be a reference to who would want to use the system but that can find it hard to understand.

3.1.2 Scope

This document gives the requirements specification for the Smart Traffic Systems program. It provides a description of the behavior of the system as well as the flow of events for each use case. Using this system, the user will be able to see the current flow of the traffic lights, check for errors and logs, watch the broadcast of the chosen streetlight and give feedback.

3.1.3 Overview

The remainder of this document will be used to specify the requirements of the system. It will also be used to illustrate the various use-cases of the program.

3.1.4 Glossary

Term	Definition
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operare. For example, this document.
IP	An IP address is a unique address that identifies a device on the internet or a local network. IP stands for "Internet Protocol".
ML	Machine Learning
AI	Artificial Intelligence
Stakeholder	Any person with an interest in the project who is not a developer.
Database	Collection of information in a structured form.
User	A person who can gather information and give feedback to the admin by using functions that are provided by this app.

3.1.5 References

[1]"IEEE Recommended Practice for Software Requirements Specifications," in IEEE Std 830-1998 , vol., no., pp.1-40, 20 Oct. 1998, doi: 10.1109/IEEESTD.1998.88286.

3.2. OVERALL DESCRIPTION

The document follows the IEEE standards, yet some of the sections are discarded as they are not compatible for this project.

3.2.1. Product Perspective

This system is meant to simulate a managed system to reduce the likelihood of traffic while remaining as efficient as possible using ML. It calculates traffic densities using image processing, compares the densities of each road, and decides on the time the lights will be green and red. Using AI, the system will send feedback to the users if a road has traffic for a long time. The system will be used by cars/drivers and can be managed by system users.

3.2.2. User Characteristics

There are two user classes for the Traffic Controller system.

User

Users can log in, check traffic lights by video broadcasts, see if everything is going properly, and contact Devteam to help improve the system.

Devteam

They are enabled to fix problems and add new features/tools in the system. They get feedback from users who check if the system has any flaws.

System

System will do image processing, vehicle detection and light signal changing using different use cases automatically and continuously

3.2.3 General Constraints and Assumptions

- The camera which we are going to implement into the system must recognize cars, pedestrian well enough to perform algorithms much more correctly in order to smooth traffic flow.
- The interface of the resulting system will be easy to use and accessible without a time or location constraint.

3.3. REQUIREMENTS SPECIFICATION

3.3.1 External Interface Requirements

3.3.1.1. User interfaces

- The user interface will be compatible with Windows.
- The user shall log in to the system by using his/her password or log out from the system.
- The user can interact with the objects in the main menu.
- The administrator will be able to inspect the logs at any time to identify errors or logs in the system.

3.3.1.2. Hardware interfaces

- Server Side

  The server will have at least 32bit architecture. The server will have a hard disk with enough capacity to hold all data. And also, since image processing is resource-hungry, devices should have modern RAM and CPU.

- Client Side

  Any PC, which can support the Windows environment and access to the ethernet/wifi, is acceptable.

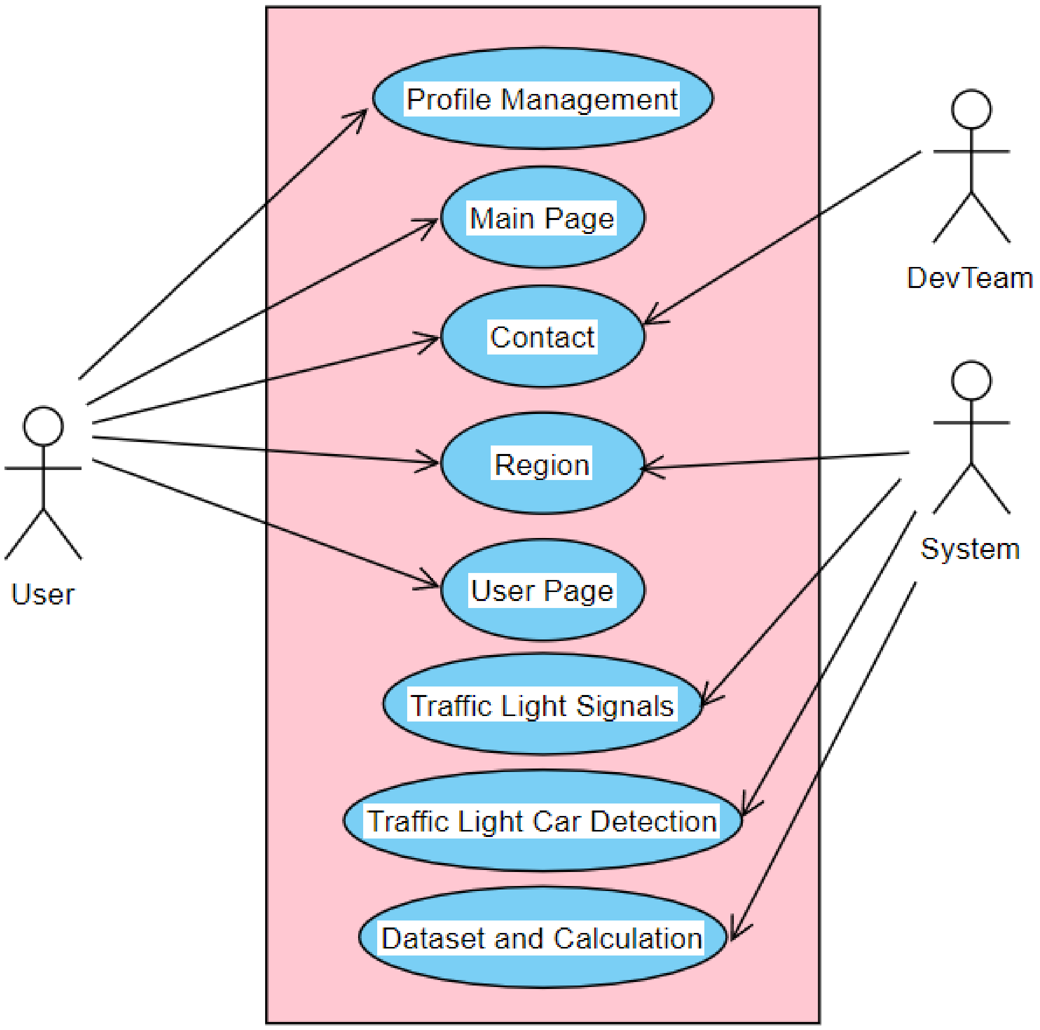
3.3.1.3. Software interfaces

- To prevent unauthorized access to the system, an external firewall can be connected to the application. There should also be a user authentication-login system in place to identify the System administrator.

3.3.1.4. Communications interfaces

- There are no external communications interface requirements.

3.3.2 Functional Requirements

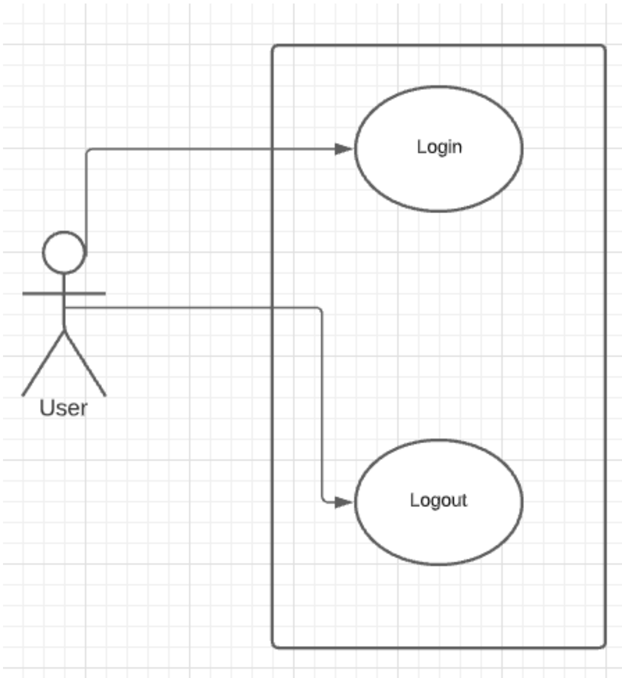


3.3.2.1. Profile Management Use Case

Use Case :

- Login as User
- Logout

Diagram :



Profile Management Use Case diagram

Brief Description :

The Profile Management diagram explains the basic operations associated with logging into the system as a user. The user is able to use the following function : Logout. Thus , the user can also use the Login function .

Initial Step by Step Description :

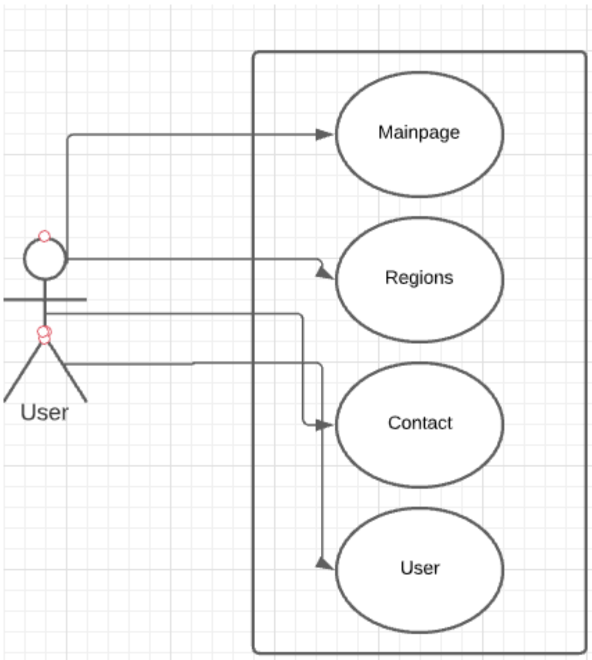
1. The user must log in to the system using his or her password.
   1.1. If the username's password is invalid, the user should re-login.
2. The user has the option to log out of the system.

3.3.2.2. Main Page Use Case

Use Case :

- Reviewing the main page
- Redirecting to Regions Page
- Redirecting to Contact Page
- Redirecting to User Profile Page

Diagram :



Main Page Use Case Diagram

Brief Description :

The Main Page diagram explains the basic navbar options. When the user enters the system , he/she can see all the options within the system. The user can execute functions of the Main Page , Regions Page , Contact Page and User Profile section .

Initial Step by Step Description :

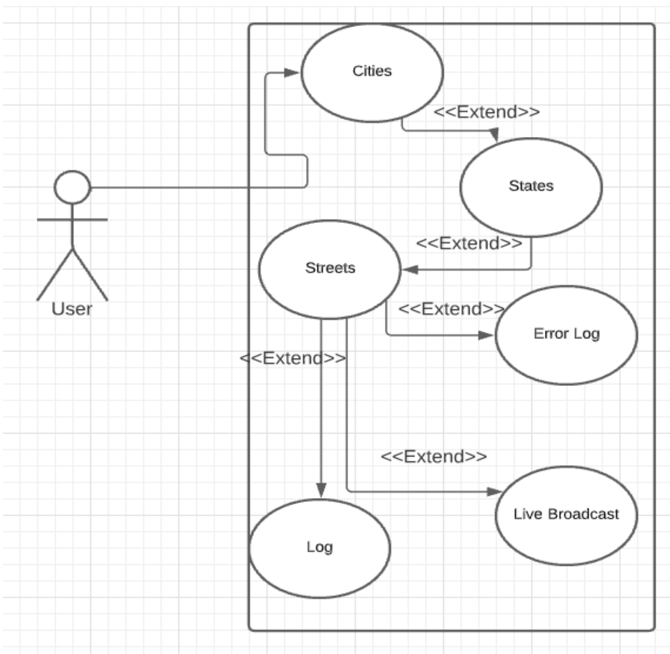
1. The user sees all the details about the system in the main page once he/she is logged in.
2. If the user clicks the Regions button , he/she will be redirected to the Region's page.
3. If the user clicks the Contact button , he/she will be redirected to the Contact page.
4. If the user clicks the User button he/she can see options such as Profile and Logout.

3.2.3. Regions Page Use Case

Use Case :

- Cities Option
  - States of the City Option
    - Streets of the States Option
      - Error Log
      - Log
      - Display Live Broadcast

Diagram :



Region Page Use Case Diagram

Brief Description :

The Region's Page diagram (Figure 3) explains the basic operations which are related to city, states and streets information from the address the user is looking for. As the user enters the region's page there are fields to fill about city, states and street information. After the user enters the address information data of traffic lights are coming out. If there is still traffic despite the traffic lights being adjusted, it is displayed as an error log. At the same time, the user can access the live broadcast of the address that he/she entered.

Initial Step by Step Description :

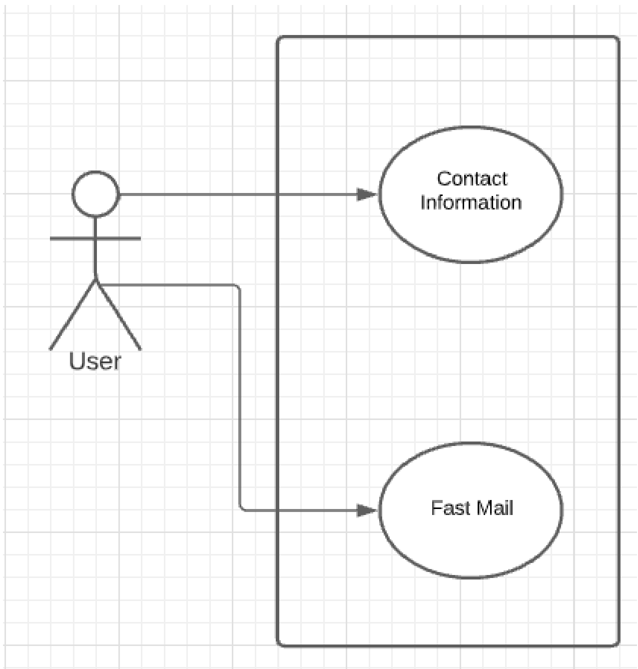
1. The user sees city, state and street options.
2. If the user clicks the City option, he/she will be able to select cities.
3. If the user clicks the State option, he/she will be able to select states.
4. If the user clicks the Street option, he/she will be able to select streets.
5. The user sees datas of the traffic lights which the user entered before.
6. If the user clicks the Live Broadcast option, he/she will be able to see the street of that traffic lights that he/she entered.
7. If there are any occurring errors on the traffic lights, users will be able to access these errors by error log options.

3.3.2.4. Contact Page Use Case

Use Case :

- Contact Information
- Fast Mail

Diagram :



Contact Page Use Case Diagram

Brief Description :

The Contact Page diagram explains the basic operations which are related to Devteam information and contact information. The contact information contains the information needed to reach the Devteam (Fax no, phone num. , e-mail address etc.).In Fast mail part, the user can reach for help or any problem simply by typing his/her email address and typing his/her message.

Initial Step by Step Description :

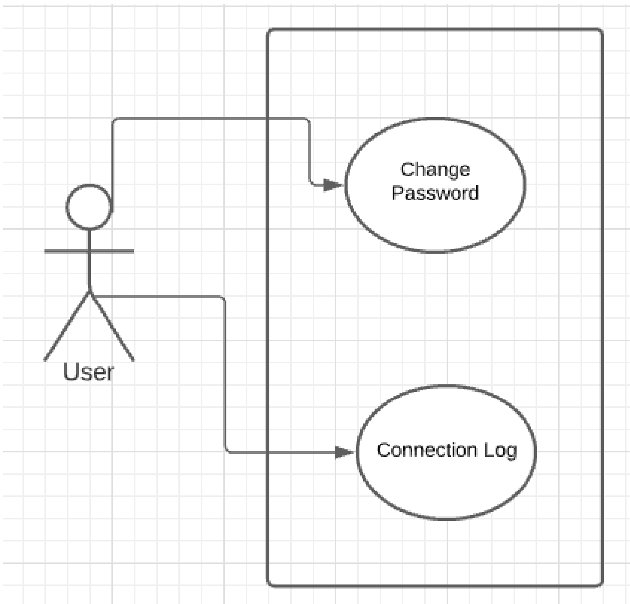
1.The user sees the Contact information and Fast mail part.
2.If the user clicks the Contact information button , he/she will see contact information about the DevTeam.
3.If the user clicks the Fast Mail button , he/she will be able to send a message to the DevTeam by typing only the email address.

3.3.2.5. User Page Use Case

Use Case :

- Change Password
- Connection Log

Diagram:



User Page Use Case Diagram

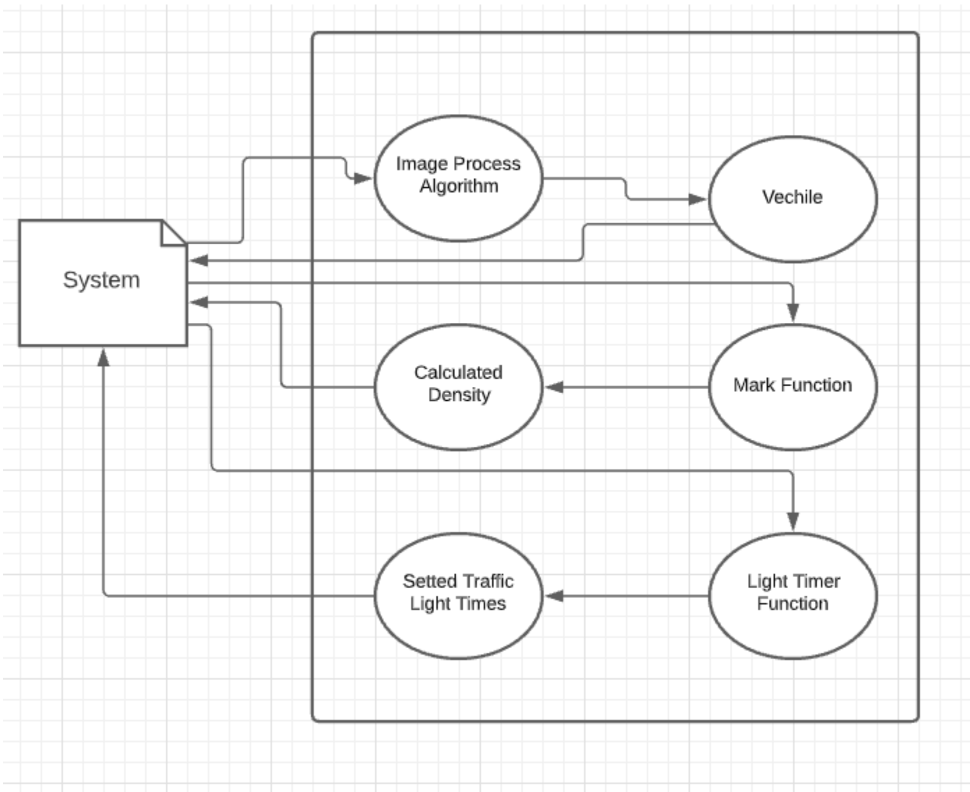
Brief Description:

The Region's Page diagram explains the basic operations which are related to changing password and connection log. When a user clicks the change password button, the user is able to change his/her password. And when the user clicks the connection log user is able to see the last connection from where and which IP.

Initial Step by Step Description:

1. The user sees change password and connection log options.
2. If the user clicks the change password option, he/she will be able to change the password.
3. If the user clicks the connection log option, he/she will be able to see the last connection from where and which IP.

3.3.2.6. System Use Case



3.3.2.6.1 Traffic Car Detection Use Case

<u>Brief Description:</u>

The system uses image processing to detect cars that are waiting at a red light, label them depending on their size, mark them green or red and send the information to the system with the information of the traffic lights number.

<u>Initial Step by Step Description:</u>

1. The system detects cars at a red light within a 30-meter range.

2. Detected cars get labeled with their size and get marked green.
3. Sends labeled and green marked car information back to the system.
4. Detects the cars that couldn't pass at the green light and marks them red when the lights turn red.
5. Sends labeled and red-marked car information back to the system.
6. Goes back to step 1 until every car on the road is marked and labeled.

3.3.2.6.2 Dataset and Calculation Use Case

Brief Description:

Collects and keeps the data of the cars waiting on each road from the system. Changes marks and labels to numerical expressions for each car and calculates traffic density for every road. Depending on the traffic density of the roads, the greater value of density will be saved as a float number and gets passed to the system with the traffic light number.

Initial Step by Step Description :

1.Gets datas from the system for each traffic light and saves them in datasets with attributes of mark colors and labels.

2. Changes labels and marks to numerical values. calculates the traffic density for every road.

3. Combines the traffic densities of opposite roads.

4. Compares traffic density values and keeps the higher value as a float. When a road is on red light, the density of the road will be compared with this float value.

5. Sends the traffic density values to the system

3.3.2.6.3 Traffic light Signals Use Case

Brief Description :

Changes the duration of red and green lights for every road depending on the traffic density value it gets from the system.

3.3.3 Performance Requirements

- The performance of the functions and each module must be satisfactory.
- The software's overall performance will allow users to work more efficiently.
- The system should produce better resource sharing outcomes.
- The performance of the results and data application must be efficient and quick.

3.3.4 Software System Attributes

There are some requirements which aren't strictly functional. These are the constraints that the system must work within.

3.3.4.1. Portability

- The System's app should work on Windows.

3.3.4.2. Performance

- The system should be able to load and save all video data in the database. Loading and saving operations should be as fast as possible and accurate for better productivity.
- The system should support all individuals connecting the system at the same time. Also they should be able to gather data and upload data at any time.

3.3.4.3. Usability

- The app interface of the system must be user-friendly and simple to use.

3.3.4.4. Adaptability

- If there is any condition, for example there are some errors or video bugs. the system should know what to do in that case.
- If there is nothing the system can do, the system must report to the people that are responsible for errors.

3.3.4.5. Scalability

- Since this system uses Machine Learning and AI concepts, the more time passes the system recognizes cars, people in the camera much more easily. Because of that ML algorithms work much smoother as the time passes.

3.3.4.6. Data integrity

- The system must maintain data integrity by making backups of all database updates for each record transaction.

3.3.5 Safety Requirement

- Because this system affects the daily traffic population, it must be secure and not allow intruders to make changes or install bugs in it. Only system administrators should be able to access, modify, halt, and perform any other functions.

4. Software Design Document

4.1. INTRODUCTION

4.1.1 Purpose

Traffic congestion is one of the most serious issues that large cities face. The surge of vehicles as a result of breakdowns in other transportation sectors has resulted in high traffic density, which is especially noticeable around '+' road junctions. Several strategies have been implemented to alleviate traffic congestion in major cities. The construction of flyovers and bypass highways, the creation of ring roads, the assignment of traffic wardens to trouble zones, and the erection of traditional traffic lights based on counters are just a few examples. However, these initiatives fell short of the goal of liberating key '+' crossings, resulting in the loss of human lives and the squandering of important man hours during working days. This Smart Traffic Systems Project Document's major goal is to minimize the traffic problem as much as possible.

4.1.2 Scope

This article provides the Smart Traffic Systems Program's implementation specifics as well as a solution to road traffic problems in large cities by designing and implementing an intelligent system based on AI and ML. Users of the Smart Traffic System Program will be able to view the current traffic flow, check for problems and logs, watch the broadcast of the selected streetlight, and provide feedback.

4.1.3 Glossary

Term	Definition
GUI	Graphical User Interface
UI	User İnterface
OpenCV	Python library for image/video recognition
SVM	Classification algorithm
AI	Artificial Intelligence
ML	Machine Learning

4.1.4 Overview of Document

Below is a list of the remaining chapters and their contents.

Architectural Design is the second section, and it describes the project development phase. It also includes a system class diagram and an architecture design that describes actors, exceptions, basic sequences, priorities, pre-conditions, and post-conditions. The scenario generator's activity diagram is also included in this section.

Use Case Realization is the third section. A block diagram of the system is illustrated and explained in this section, which is designed according to use cases in the SRS paper.

The fourth section is about the environment. We've presented example frames of the environment from the prototype and detailed the situation in this part.

4.1.5 Motivation

We are a group of university students in the computer engineering department who are interested in Artificial Intelligence and Machine Learning technologies. As a group, we have participated in various AI and ML conferences to listen to different opinions about experts. Attending such conferences has allowed us to look at some problems from different angles. In addition, we took ML and AI lessons as a group. These lessons were the biggest factor in making this article. The articles we wrote and the projects we prepared during lessons took us to a higher level in AI and Ml. Because ın every project you do, new information and results about AI and ML technology emerge. AI and ML is a world that is without end and where each new knowledge creates another knowledge.

4.2. ARCHITECTURE DESIGN

4.2.1 Simulation Design Approach

We're developing a system that combines artificial intelligence, machine learning, and computer vision technologies to automate traffic light monitoring systems. Every day, cameras mounted on traffic signals collect thousands of hours of footage that contains extremely useful information and can be utilized to minimize street congestion. It is feasible to produce traffic condition predictions using video footage data, a data set of traffic films, and machine learning algorithms.

Pattern recognition can be used to identify vehicle types, count the number of vehicles on the road, and classify light and heavy vehicles. This would increase the system's ability to predict traffic on each street and adjust traffic lights accordingly to reduce congestion. Because the training data sets contain enough data to forecast, the machine learning system can perform better. These data sets are expanding on a daily basis, and they can assist our system in determining traffic problems by observing daily data. Signals based on hard-coded time interval approaches do not always deliver outstanding outcomes in overcrowding circumstances. Because of the set time period, every street can have a distinct traffic scenario every hour, if not minute, causing time-based signals to fail and traffic jams to occur.

Using machines to complete this task can be very efficient and dependable. The suggested system analyzes forecasting future and present traffic to assist in determining when and where a signal release operation should be carried out. The suggested system will be a synchronized intelligent traffic signal monitoring system that will improve traffic monitoring and the way we deal with various transportation-related challenges and circumstances. A monitoring system and a control system will be the two components of the suggested model. Both components can communicate with one another and use data from the same directed backward signals at the same time to make intelligent decisions. The proposed model architecture is shown in Figure 1.

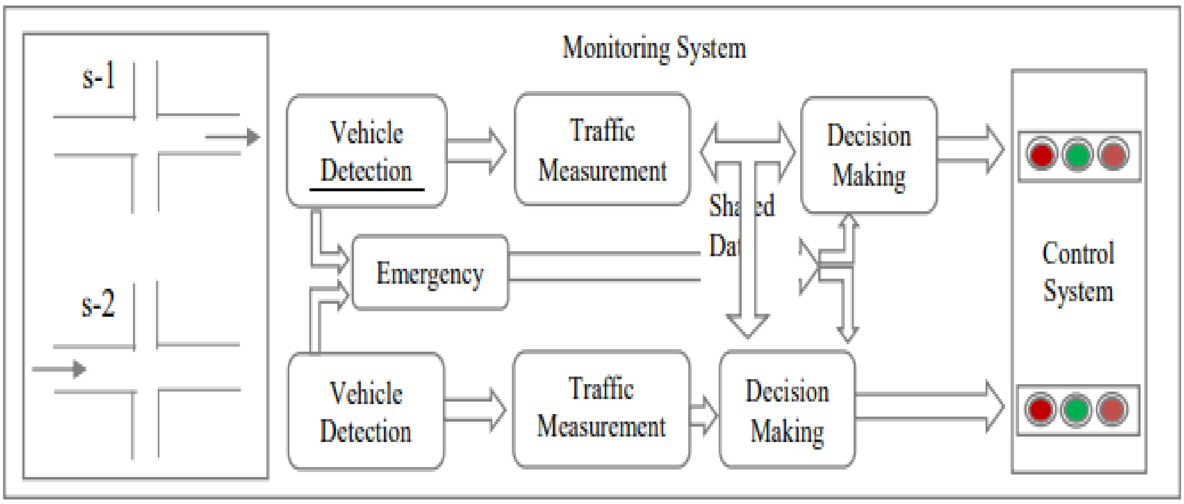


Fig - 1 Simulation Design

4.2.1.1 Class Diagram

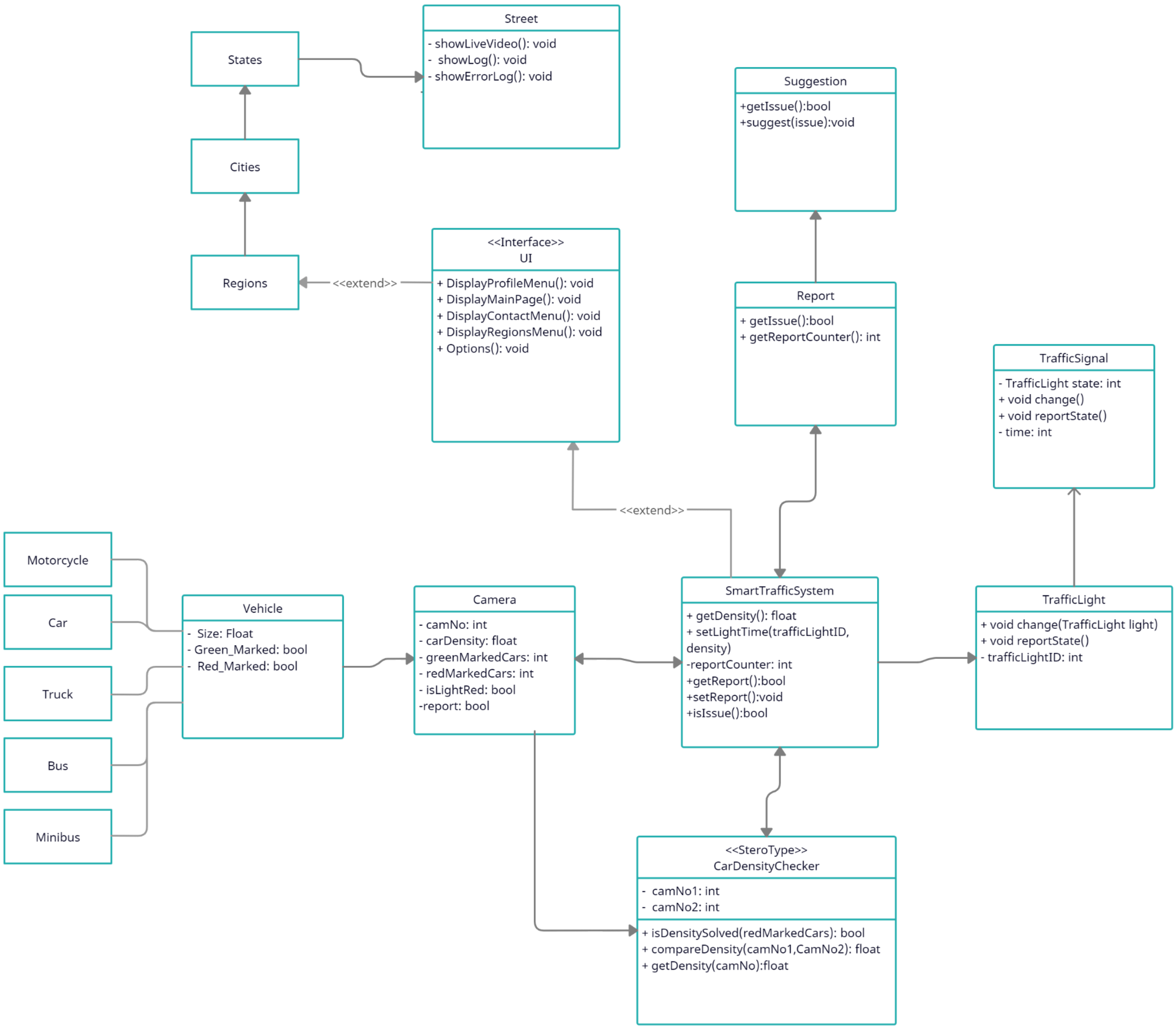


Figure 2: Class Diagram of Smart Traffic System project

Figure 2 displays information about connections between the systems within the simulation.

- SmartTrafficSystem class is the main system, which contains other systems. It is responsible for connections between other systems such as TrafficLight, Camera, Interface,CarDensityChecker.
- Camera class collects all information from the streets and feeds the system with information like, car density, marked vehicles -green,red- and also the current state of the light.
- Vehicle class that is derived from the camera class represents information about vehicles which are used in traffic and has the information of the car density value of each vehicle type.
- TrafficLight class has information of traffic light's ID, current state of the traffic light and timer on the light. UI class represents the User Interface which the Actors of the system will encounter.
- The CarDensityChecker class does the calculation of data and compares the density of roads. After doing that, the class sets the timer on the light.

- Report class uses information from the camera, it checks for red_marked vehicles. Red_marked means that if the vehicle is red_marked, the vehicle waited at the light more than 1 times. If red_marked vehicles get higher from the threshold. It reports an issue and makes suggestions to correct that situation.

4.2.2 Architecture Design of System

Light Check and Vehicle Detection

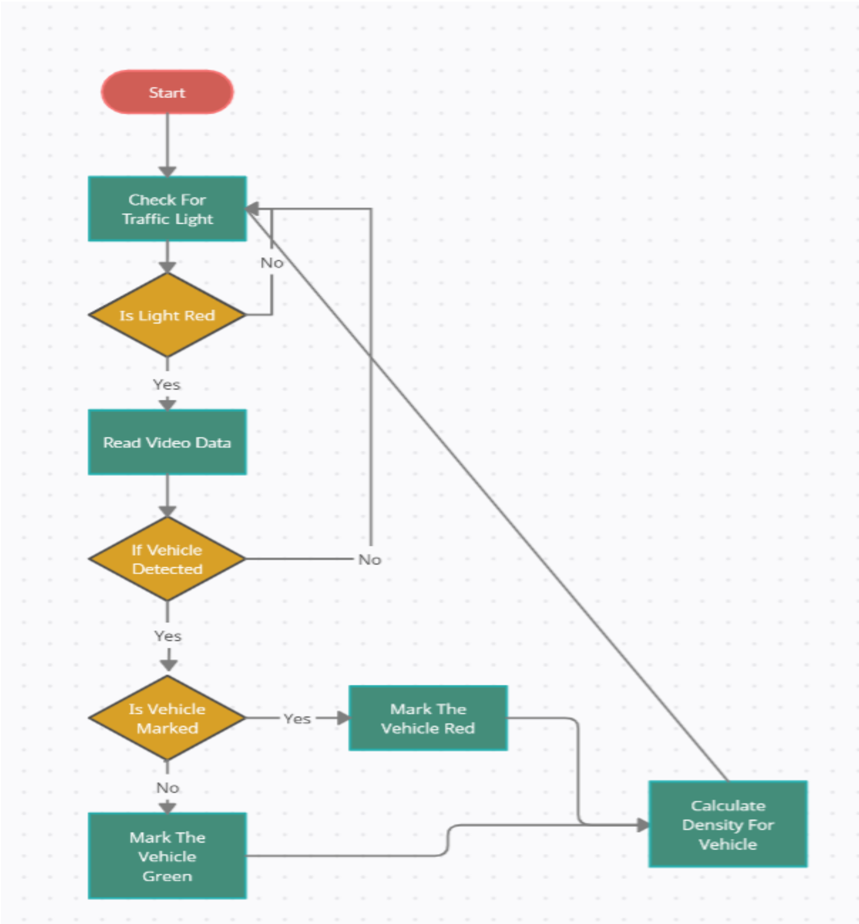


Figure 3: Activity Diagram of Light Check and Vehicle Detection

Summary: This system is used by System. System gets data from the traffic light and camera. After getting data from the system do a calculation to density ratio on the street and set the time of lights.

Actor: System

Precondition: System must be runned, camera must be setted.

Basic Sequence:

1. System checks for red lights.
2. System reads data from the video.
3. If a vehicle exists, mark it green or red.
4. Calculate density based on marks.

Exception:

Cameras can’t see well

Database connection can be failed

Bad learning on train

Post Conditions: System gets density.

Priority: High

Setting Light Time:

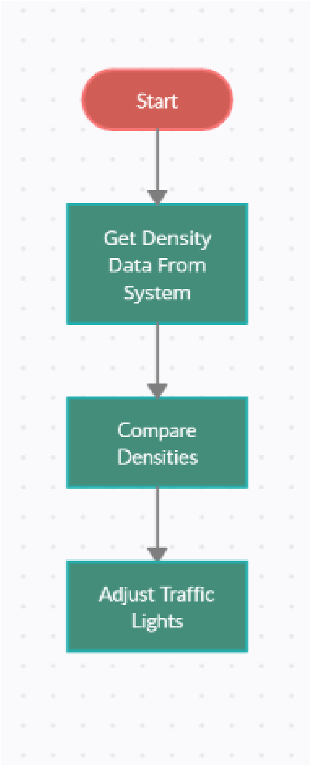


Figure 4: Activity Diagram of Setting Light Time

Summary: This system is used by System. System gets density data from the check system. Compares densities based on roads and sets the traffic light times.

Actor: System

Preconditions: System must be runned, data flow needs to be clear.

Basic Sequence:

1. System gets density from the check system.
2. System compares density of roads
3. Set traffic light time based on density ratio.

Exception: Database connection can be failed.

Post Condition: None.

Priority: High.

Red Mark Count

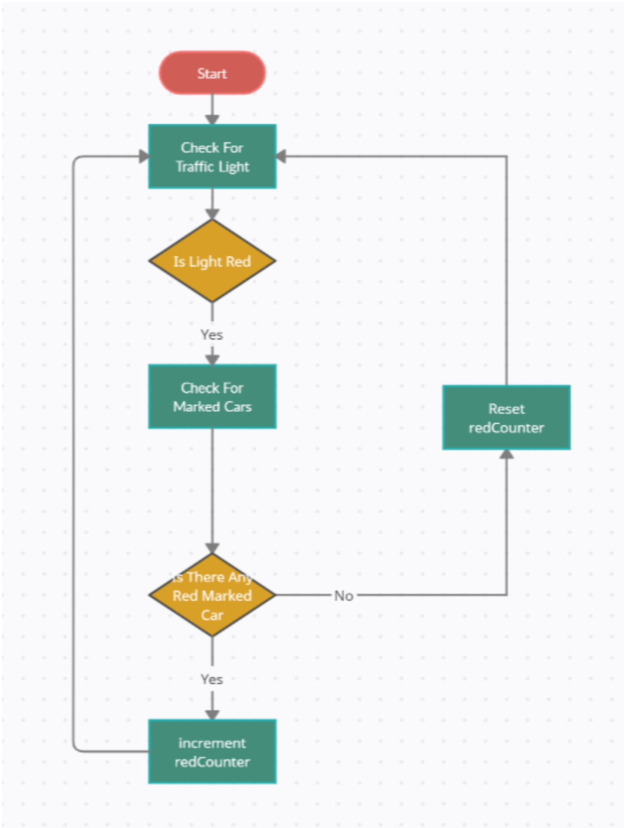


Figure 5: Activity Diagram of Red Mark detection

Summary: This system is used by System. System gets red marked vehicle data from the check system. Compares densities based on roads and sets the traffic light times.

Actor: System

Preconditions: System must be runned, data flow needs to be clear.
Basic Sequence:

1. System gets red marked vehicle data from the check system.
2. System change counter based by if condition

Exception: Database connection can be failed.

Post Condition: Feedback System

Priority: High.

Feedback System

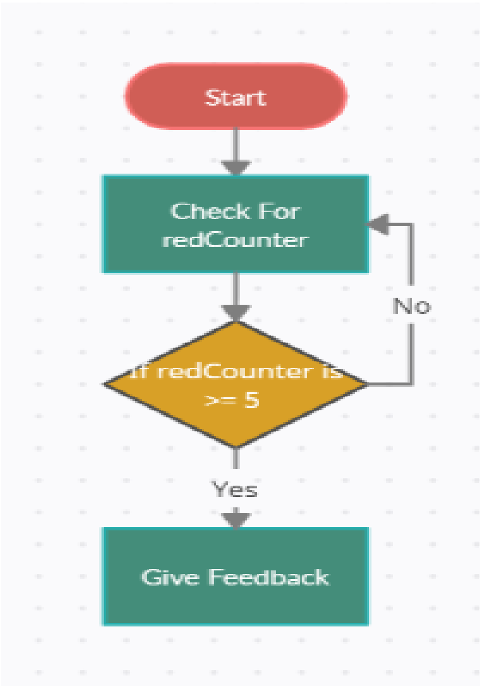


Figure 6: Activity Diagram of Give Feedback

Summary: This system is used by System. System gets counter data from the Red Mark Count System. Check counter value.

Actor: System

Preconditions: System must be runned, data flow needs to be clear.

Basic Sequence:

1. System gets counter data from the Red Mark Count System.
2. System gives feedback about to make upgrade this street

Exception: Database connection can be failed.

Post Condition: None.

Priority: High.

4.3. Vehicle Recognition

Troubleshooting Procedure

The machine learning steps for car recognition will be broken down into three stages: pre-processing, training, and identification. In general, the system will be constructed as follows:

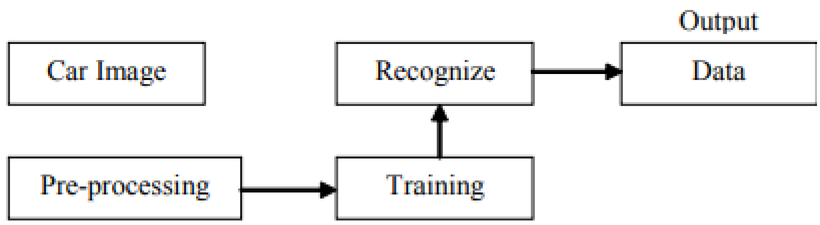


Figure 7: Diagram of Car Recognition System

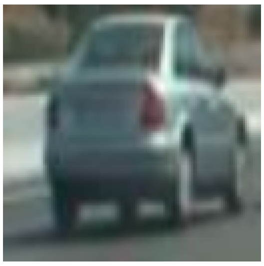
The vehicle images are entered into the database via the first stage reading camera. The system will read the image and undertake a pre-processing step after it has been entered. Following the successful completion of preprocessing, the system began the training step, which is the phase during which the system learns to recognize an item (in this case car type). Following then, a well procedure, namely the introduction of a previously taught system, will be used to carry out the next step. The result will display the introduction of the type of car after the entire process has been completed successfully.

4.3.1 Datasets

In supervised learning, datasets used for machine learning research assign labels to the data. The vehicle detection system contains a large number of datasets, which allows for more accurate findings and a lower proportion of error throughout the machine learning process. A training set and a testing set are created from the dataset. Depending on the task's difficulty, the ratio between the training and testing sets may change.

4.3.1.1 Training dataset

The set of training samples is represented by the training dataset. It is a compilation of both positive and negative information. They are usually grouped together by the same number of components or features. The training dataset is an important component of future feature extraction for vehicle detection.



Car image example from the training set

Non-car image example from the training set

4.3.1.2 Testing dataset

The testing dataset is described as data that allows for an objective assessment of the final model after it has been fitted to the training dataset. It is the gold standard that is used in practice after the model has been fully trained.

4.3.2 Training and classification

In detection of the car classifier training is essential and can be done by its set of features, so that the computer can distinguish if the car is an image or not. This is classified by class labels such as vehicle and non-vehicle (OpenCV library). Training and classification are the substantial steps to reach the wanted goal. Process of step named training starts with getting a specified class's known content hence producing a classifier concerning this content. The quality of the process of training influences the output results iteratively while this classification of running on unknown content is a on-time process.

After being trained with a prepared dataset, Support Vector Machine (SVM) is one of the most popular supervised binary classification algorithms for deciding whether or not a vehicle exists. Figure 6 depicts the central concept underlying the SVM. There are two types of exits (for example, vehicles and non-vehicles), each represented by a different type of dot. During training, the algorithm is given a large number of examples from two classes. The task of the algorithm is to separate those classes. Figure 6 clearly shows the optimal hyperplane, which is the line that connects the two classes. The support vector machine will select the best plane that separates no more than two classes.

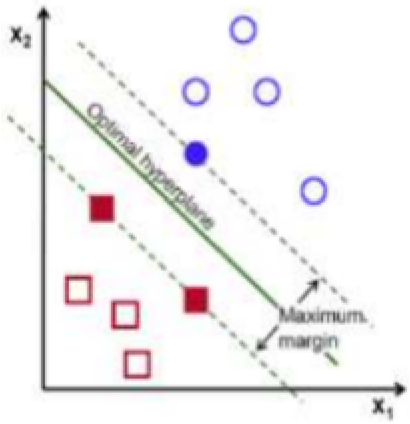


Figure 6: Support Vector Machine (SVM)

4.3.3 OpenCV library

OpenCv (Open Source Computer Vision Library) is a solid open - source software library. It contains around 2500 computer vision and machine learning algorithms that have been optimized. Face detection and recognition, object identification, human activity classifications in movies, and camera movement tracking are some of the applications of these techniques. The library is written in C and C++ to improve calculation efficiency and is compatible with most operating systems.

The OpenCv library is used to build the vehicle detection system. Its major aim is to modify received frames by analyzing them and creating a representation of them. cvSVM::CVM was used to classify two types of objects: vehicles and non-vehicles.

4.4. GUI Design

GUI design is responsible for interaction between the actors and the system. In this GUI design , there are also some sub-systems which are Main Page , Contact, Region , User Page . Main Page is a start page that the user sees as soon as the page is opened. He/She can see all the menus and information about the site , login,logout or register to the system. Region Page is the page that everyone can select the region, city , district and street . As the user clicks that Region Button , there is a search box in every section of the menu . The user can search the region,city,distinct,street etc. By doing this , there will be a live broadcast playing in the middle of the page and the user will be able to see the traffic situation on that specific street . Contact Page , is the page where there will be a contact form for users to type whatever they want like requests,complaints,collaborations etc.In the User Page section , the user can edit his/her page , change settings, photo or general information .

Main Pag

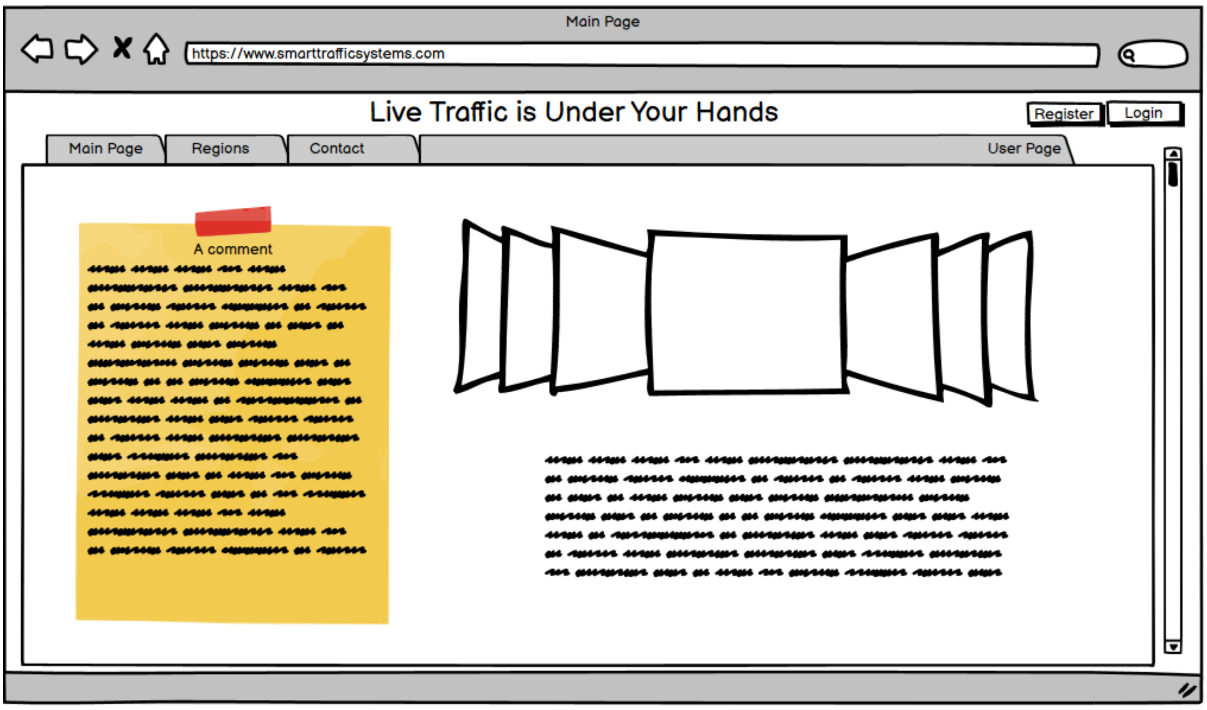


Figure - 9 : UI Design of Main Page (No Login)

After Log In

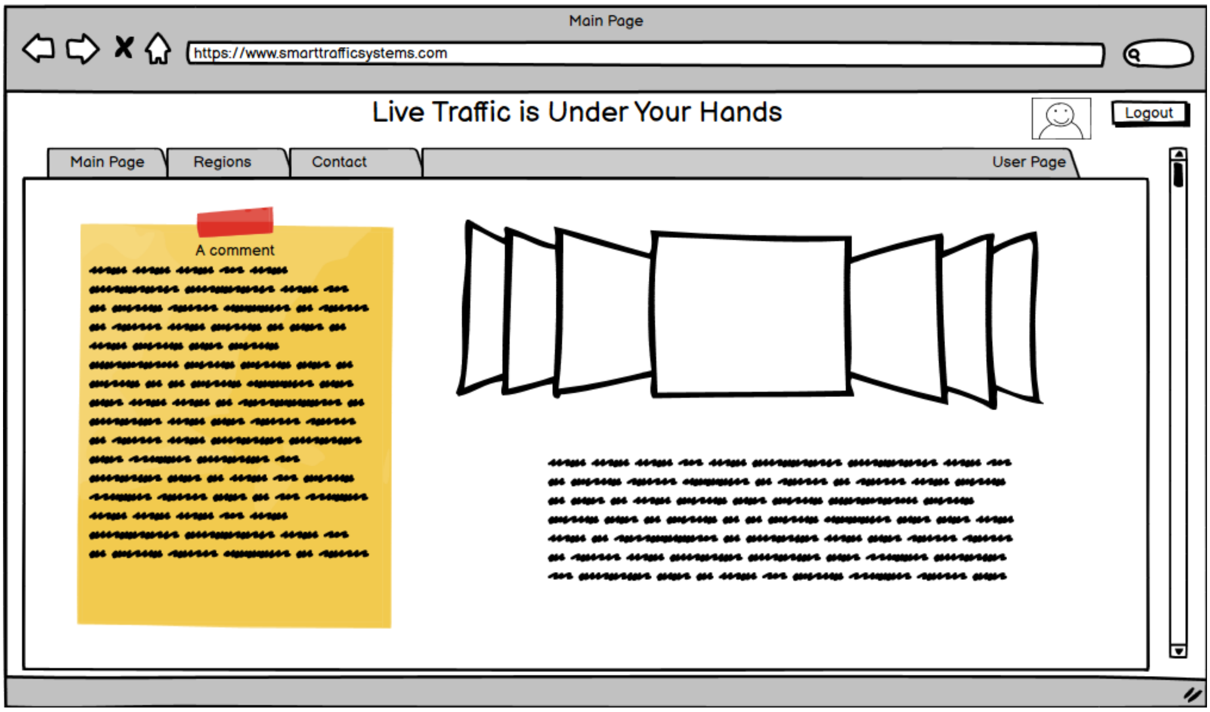


Figure - 10 : UI Design of Main Page (User Login)

Regons Dropdown Menu

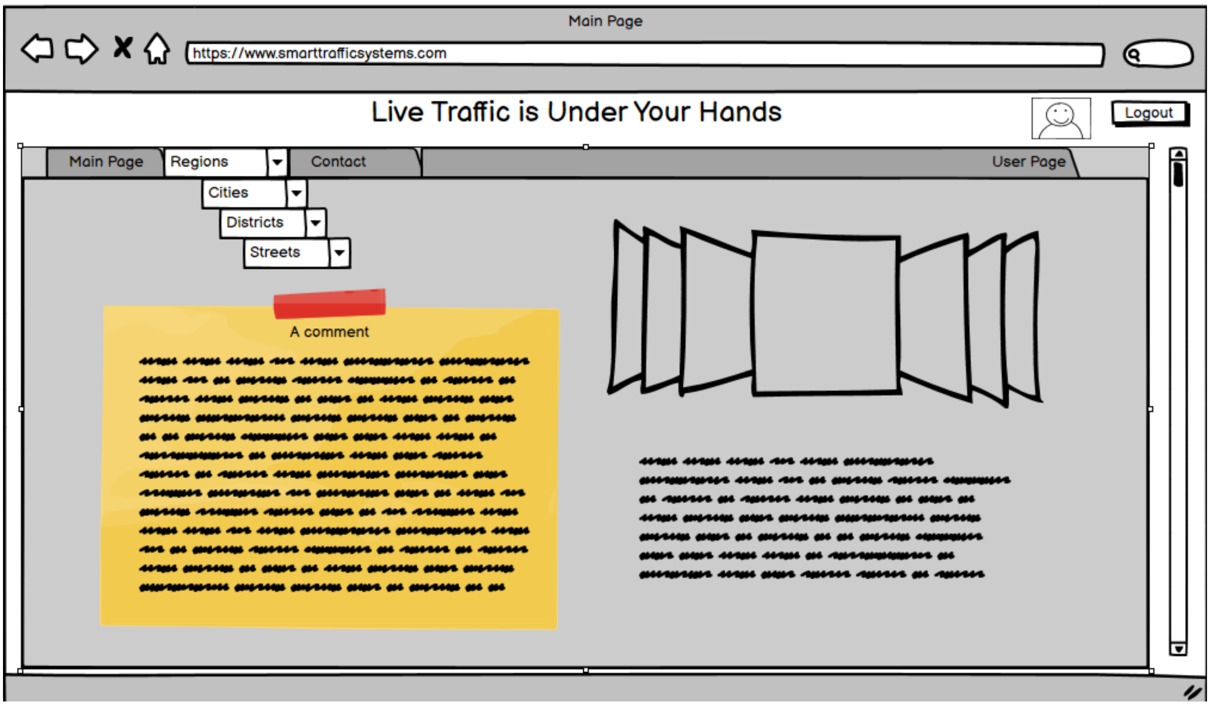


Figure - 11 : UI Design of Regions Page

Contact Page

Figure - 12 : UI Design of Contact Page

User Page

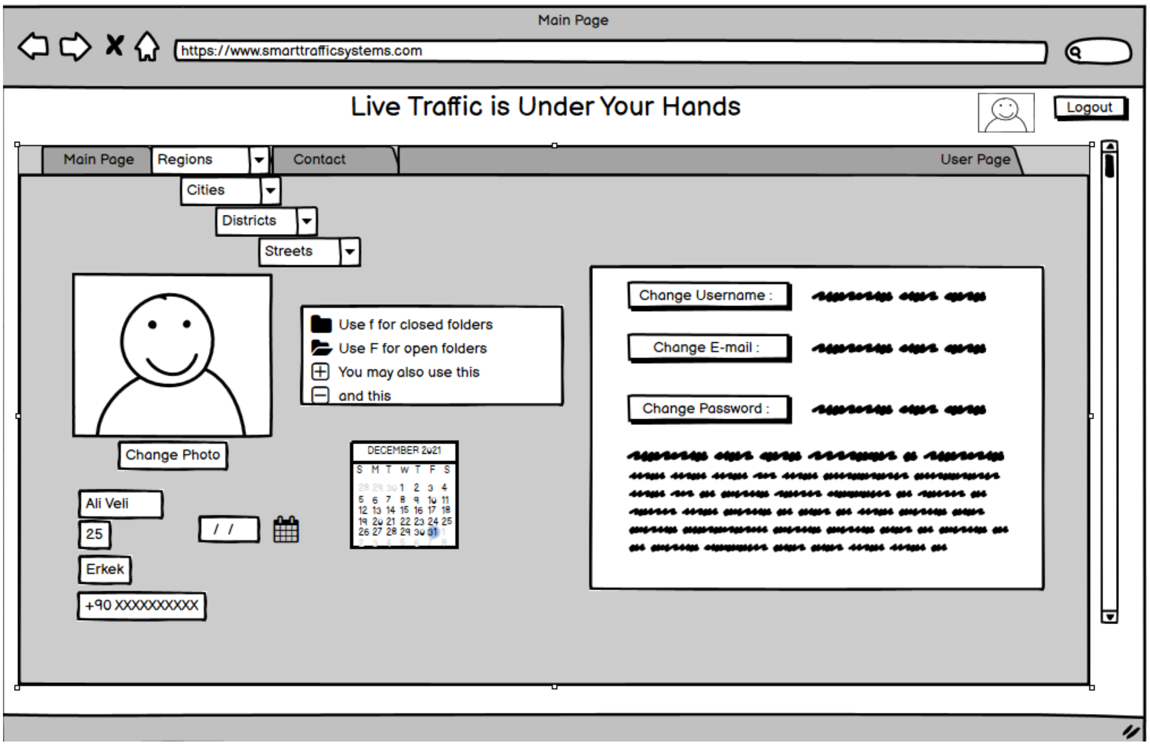


Figure - 13 : UI Design of User Page

5. Conclusion

Traffic these days has a big impact on our lives. We waste our precious time waiting because of poorly built systems that could be modified and derived to a better system using AI, machine learning and such so that it is more efficient. More efficiently built systems will help us with our traffic problem and let us have less time spent on roads.

6. References

1. Emad Issa Abdul Kaream and Aman Jantan, “Intelligent Traffic Light Control Using Neural Network with Multi-Connect Architecture”, National Conference on Information Retrieval and Knowledge Management (CAMP08) in Kuala Lumpur,Malaysia, 2008.
2. Emad Issa Abdul Kaream, k. N. M., Hussein A. Moussa, “Gray Image Recognition Using Hopfield Neural Network With Multi- Bitplane and Multi-Connect Architecture”, Proceedings of the international Conference on Computer Graphics, Imaging and Visualisation (CGIV'06) IEEE, 2006.
3. S. S. Chavan, R. S. Deshpande, and J. G. Rana, “Design of intelligent traffic light controller using embedded system,” in *Proceedings of the 2009 Second International Conference on Emerging Trends in Engineering & Technology*, pp. 1086–1091, Maharashtra, India, December 2009.
4. K. Matsuzaki, M. Nitta, and K. Kato, “Development of an intelligent traffic light for reducing traffic accidents,” in Proceedings of the 2008 International Conference on Control, Automation and Systems, pp. 443–447, Hanoi, Vietnam, December 2008.
5. K.-H. Chao, R.-H. Lee, and K.-L. Yen, “An intelligent traffic light control method based on extension theory for crossroads,” in Proceedings of the 2008 International Conference on Machine Learning and Cybernetics, pp. 1882–1887, Kunming, China, July 2008.
6. V. Hirankitti and J. Krohkaew, “An agent approach for intelligent traffic-light control,” in Proceedings of the First Asia International Conference on Modelling & Simulation (AMS’07), pp. 496–501, Phyket, Thailand, March 2007.
7. M.-D. Pop, “Traffic lights management using optimization tool,” Procedia-Social and Behavioral Sciences, vol. 238, pp. 323–330, 2018.
8. H. Joo, S. H. Ahmed, and Y. Lim, “Traffic signal control for smart cities using reinforcement learning,” Computer Communications, vol. 154, no. 3, pp. 324–330, 2020.
9. M. Radivojević, M. Tanasković, and Z. Stević, “The adaptive algorithm of a four way intersection regulated by traffic lights with four phases within a cycle,” Expert Systems With Applications, vol. 166, p. 2021, 2019.
10. M. Drapalyuk, S. Dorokhin, and A. Artemov, “Estimation of efficiency of different traffic management methods in isolated area,” Transportation Research Procedia, vol. 50, pp. 106–112, 2020.
11. A. Yousef, A. Shatnawi, and M. Latayfeh, “Intelligent traffic light scheduling technique using calendar-based history information,” Future Generation Computer Systems, vol. 91, pp. 124–135, 2019.
12. Q. Wu, J. Wu, J. Shen, B. Yong, and Q. Zhou, “An edge based multi-agent auto communication method for traffic light control,” Sensors, vol. 20, no. 15, pp. 4291–4316, 2020.
13. T. Wu, P. Zhou, K. Liu et al., “Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks,” IEEE Transactions on Vehicular Technology, vol. 69, no. 8, pp. 8243–8256, 2020.
14. K. L. Soon, J. M.-Y. Lim, and R. Parthiban, “Coordinated traffic light control in cooperative green vehicle routing for pheromone-based multi-agent systems,” Applied Soft Computing, vol. 81, Article ID 105486, 2019.
15. N. Kumar and S. S. Rahman, “Deep reinforcement learning with vehicle heterogeneity based traffic light control for intelligent transportation system,” in Proceedings of the 2019 IEEE International Conference on Industrial Internet (ICII), pp. 28–33, Orlando, FL, USA, November 2019.
16. P. M. Kumar, U. Devi Gandhi, G. Manogaran, R. Sundarasekar, N. Chilamkurti, and R. Varatharajan, “Ant colony optimization algorithm with internet of vehicles for intelligent traffic control system,” Computer Networks, vol. 144, pp. 154–162, 2018.
17. K. Gao, Y. Yicheng Zhang, A. Sadollah, and R. Rong Su, “Improved artificial bee colony algorithm for solving urban traffic light scheduling problem,” in Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), pp. 395–402, San Sebastián, Spain, June 2017.
18. K. M. Alam, M. Saini, and A. E. Saddik, “Toward social internet of vehicles: concept, architecture, and applications,” IEEE Access, vol. 3, pp. 343–357, 2015.
19. M. S. Roopa, S. Ayesha Siddiq, R. Buyya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, “Dynamic management of traffic signals through social IoT,” Procedia Computer Science, vol. 171, pp. 1908–1916, 2019.
20. T. S. Gomides, R. E. De Grande, A. M. D. Souza, F. S. H. Souza, L. A. Villas, and D. L. Guidoni, “An adaptive and distributed traffic management system using vehicular ad-hoc networks,” Computer Communications, vol. 159, pp. 317–330, 2020.
21. A. Bazzi, A. Zanella, and B. M. Masini, “A distributed virtual traffic light algorithm exploiting short range V2V communications,” Ad Hoc Networks, vol. 49, pp. 42–57, 2016.
22. Z. Cao, S. Jiang, J. Zhang, and H. Guo, “A unified framework for vehicle rerouting and traffic light control to reduce traffic congestion,” IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 7, pp. 1958–1973, 2017.
23. V.-T. Ta and A. Dvir, “A secure road traffic congestion detection and notification concept based on V2I communications,” Vehicular Communications, vol. 25, Article ID 100283, 2020.
24. V. Astarita, V. P. Giofrè, G. Guido, and A. Vitale, “A review of traffic signal control methods and experiments based on floating car data (FCD),” Procedia Computer Science, vol. 175, pp. 745–751, 2020.
25. S. Brakatsoulas, D. Pfoser, and N. Tryfona, “Practical data management techniques for vehicle tracking data,” in Proceedings of the 21st International Conference on Data Engineering (ICDE’05), pp. 324-325, Tokyo, Japan, April 2005.
26. H. Lamghari Elidrissi, A. Nait-Sidi-Moh, and A. Tajer, “Knapsack problem-based control approach for traffic signal management at urban intersections: increasing smooth traffic flows and reducing environmental impact,” Ecological Complexity, vol. 44, Article ID 100878, 2020.
27. S. Kamishetty, S. Vadlamannati, and P. Paruchuri, “Towards a better management of urban traffic pollution using a pareto max flow approach,” Transportation Research Part D: Transport and Environment, vol. 79, Article ID 102194, 2020.
28. A. Villagra, E. Alba, and G. Luque, “A better understanding on traffic light scheduling: new cellular GAs and new in-depth analysis of solutions,” Journal of Computational Science, vol. 41, Article ID 101085, 2020.
29. W. C. Tchuitcheu, C. Bobda, and M. J. H. Pantho, “Internet of smart-cameras for traffic lights optimization in smart cities,” Internet of Things, vol. 11, Article ID 100207, 2020.
30. G. Padmavathi, D. Shanmugapriya, and M. Kalaivani, “A study on vehicle detection and tracking using wireless sensor networks,” Wireless Sensor Network, vol. 2, no. 2, pp. 173–185, 2010.
31. B. Zhou, J. Cao, and H. Wu, “Adaptive traffic light control of multiple intersections in WSN-based ITS,” in Proceedings of the 2011 IEEE 73rd Vehicular Technology Conference (VTC Spring), Budapest, Hungary, May 2011.
32. L. F. P. Oliveira, L. T. Manera, and P. D. G. Luz, “Smart traffic light controller system,” in Proceedings of the 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), pp. 155–160, Granada, Spain, October 2019.
33. I. M. Albatish and S. S. Abu-Naser, “Modeling and controlling smart traffic light system using a rule based system,” in Proceedings of the 2019 International Conference on Promising Electronic Technologies (ICPET), pp. 55–60, Gaza, Palestinian Territories, October 2019.

34. P. Barbecho Bautista, L. Lemus Cárdenas, L. Urquiza Aguiar, and M. Aguilar Igartua, “A traffic-aware electric vehicle charging management system for smart cities,” Vehicular Communications, vol. 20, Article ID 100188, 2019.
35. S. Nesmachnow, R. Massobrio, E. Arreche et al., “Traffic lights synchronization for Bus Rapid Transit using a parallel evolutionary algorithm,” International Journal of Transportation Science and Technology, vol. 8, no. 1, pp. 53–67, 2019.
36. X. Cabezas, S. García, and S. D. Salas, “A hybrid heuristic approach for traffic light synchronization based on the MAXBAND,” Soft Computing Letters, vol. 1, Article ID 100001, 2019.
37. J. Li, Y. Zhang, and Y. Chen, “A self-adaptive traffic light control system based on speed of vehicles,” in Proceedings of the 2016 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 382–388, Vienna, Austria, August 2016.