ÇANKAYA UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

Project Report

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 nearly 40 hours each year in traffic which could be used properly. In cities, traffic management is mostly governed by traffic police and signal lights that contribute to the long wait for vehicles due to an inefficient configuration. Also, in many urban areas, most of the traffic signal lights are based on a fixed cycle protocol, which is a reason for the inefficient configuration. That's why there is a need to automate and improve these traffic systems. The mixed work of machine learning, artificial intelligence (AI) and computer vision can be desirable to develop a scalable traffic system which can solve such problems as timings of the traffic lights, traffic jams etc. 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. Machine learning, AI and Vision-based technologies have the potential to cover all

these parameters. This paper provides an insight into how these technologies can be applied and the difficulties which we will have to face.

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 goal of traffic and transportation research is to optimize the transportation flow of people and goods. As the population grows, the traveler amount increases while resources are not sufficient for this. Therefore, intelligent control of traffic became a really serious issue. Traffic in the urban areas is generally based on fixed cycle traffic signal lights, which is not properly configured in many cases and causes unnecessary extra waiting times for the vehicles [1]. The traditional traffic signal monitoring techniques include fix-interval monitoring, daytime monitoring, vehicle actuated and semi-actuated monitoring, green wave monitoring, area dynamic monitoring and area static monitoring. However, there is no system which we can call a capable and effective system or can be adopted in the real world effectively. In order to find an answer to this problem, a lot of researchers' groups have performed a lot of studies and researches. The application of image processing methods in automated traffic systems has been reviewed to optimize methodologies for traffic controls in recent years. 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 when 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 vehicles in surveillance data is one of the most common and approved problems in computer vision. Here we have made an attempt to detect the actual vehicle and count of the vehicles and detect the speed of them. These observations are not limited to a single traffic signal. We introduce a new concept of synchronous data sharing with the same directional backwards-forward traffic signals that provide better results for automation. The cameras that are positioned at the top view and use the footage as input can give us feedback. Therefore, cameras give the system great installation flexibility and increase the durability, efficiency and portability of the system.

This paper aims to propose an intelligent synchronized traffic signal monitoring system to improve traffic monitoring and enhance the way we are dealing with some other problems and situations related to transportation. It will be in the intelligent traffic system. Thus, an intelligent traffic monitoring system will be formed of two segments (or subsystems): a control system and a monitoring system will be able to integrate together to help the traffic system to take smart decisions efficiently. This research will be examining the additional component (monitoring system) by using synchronous data sharing between its control system. This system will be capable of identifying different types of vehicles, their speed, and determining the crowded traffic jam case, normal traffic jam case and empty traffic jam case. The input of this system will be a sequence of images of vehicles of the crossing to be monitored. Technically, the proposed method has two stages: the training stage and the recognition stage which uses one video camera to every crossing of the area. These cameras are fixed in a suitable position and use multi-connect architecture based multi-line reference techniques [2] to detect the level of the crowd in the lane.

1.1. Previous Works Of Smart Traffic Systems

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

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

The most common way is to set up a traffic signal at each road junction to manage access to the critical section (the junction). Several variants of priority access and critical sections are implemented. One of the variations is the round tape, which 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 closing the road by turning on the red light, and repeat the process for the next street, etc. This is the least efficient approach to traffic management; however, it is the most widely used system in the world, except in several developed countries.

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].

Another study is presented in [11] that uses multi-agent communication based on edge computing architecture and IoT for traffic light control. The authors propose a multi-agent reinforcement learning system (MARL) for global traffic sign management. Similar studies based on MARL are presented in [12] ,[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 level of congestion on the roads. For example, in [21], an adaptive algorithm is presented and evaluated. In this study, the objective is to use V2V so that each vehicle can estimate the level of traffic congestion and can redirect itself to the least congested route. Another study based on V2V communication is proposed in [22]. A traffic optimization framework based on vehicle redirection to reduce traffic congestion is provided in [23]. In another study in [24], a system based on V2I communications is proposed.In addition, this study considers the protection of incident detection and the spread of different 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.

The Modular Timed Synchronized Petri Net model is proposed in [27] for the management of traffic signs in order to reduce the environmental impact. The maximum Pareto flow algorithm is presented in [28] and the cell genetics algorithm is proposed in [29]. There are several research articles on the use of cameras to count the number of vehicles for traffic management and optimization. A recent study on the use of the Internet of smart cameras is presented in [30]. The solution is based on WSN(Wireless sensor network) and VANET(Vehicular Ad-hoc Network) by implementing a very large number of cameras connected in a dedicated infrastructure. The video feed from the cameras is sent to centralized servers for processing and extracting useful traffic information that can be used to check traffic signs. Further studies based on the WSN are provided in [31, 32]. Several IoT(Internet of Things)-based approaches are proposed as in [33, 34]. The authors of [35] propose the use of expert systems and AI to process the images extracted from the cameras for traffic management. A pheromone-based multi-agent system based on the use of cameras and sensors is proposed in [13].

An interesting study which selects the best charging station for electric vehicles according to traffic conditions in order to minimize travel time is presented in [36]. A parallel algorithm for synchronizing intersections in large and dense areas which suggests improving Bus Rapid Transit based on average speed is proposed in [37] A similar study based on a hybrid heuristic approach is presented in [38]. Finally, in [39], the authors suggest the use of vehicle speed at the intersection 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.

Maximum machines in the world are still blind. [3] Today there is no need to drive cars manually only because of computer vision based self-driving cars. Without this technology, they cannot understand the difference between crumpled paper and rock. Where they can run over and where they should be avoided. Computer vision helps the machine to understand anything like we humans do from identifying people, naming objects, understanding relations, actions, intentions and emotions. Computers learn from a data set of images as self-training data and design a model which helps to predict. This data set is where huge data information helps the machine to train itself and learn in time. About 3 billion photos are shared by users every day. Today humans upload a vast number of images daily on the internet that can be used for prediction or machine training. Recognising or classifying an object is different and to make a prognostication is something different. For which high-level algorithms of machine learning are required. The traffic management system consists of CCTV's and sensors that can improve computer performance to make real-time decisions. The CCTV could identify the number of vehicles, lightweight vehicles and high weight vehicles. These can assist in determining the peak time in which there is a higher possibility of traffic.

Over the past several decades, an enormous amount of standard transformations have been encountered in the field of computing. Earlier, everything was built on logic and mathematical problems. This logic cannot be used to solve real-world intricacies.An improved system is to be designed which will learn by itself and take decisions at the real-time. Machine learning can be categorized into sections based on the learning methods used. These methods are, Supervised machine learning algorithms , Unsupervised machine learning algorithms , Semi - supervised machine learning algorithms, Reinforcement machine learning algorithms.To briefly mention these methods, Supervised machine learning algorithms can apply what has been learned in the past to new data using labeled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. In contrast, unsupervised machine learning algorithms are used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. Semi-supervised machine learning algorithms, systems that use this method are able to considerably improve learning accuracy. Usually, semi-supervised learning is chosen when the acquired labeled data requires skilled and relevant resources in order to train it / learn from it. Reinforcement machine learning algorithms is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behavior within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal.

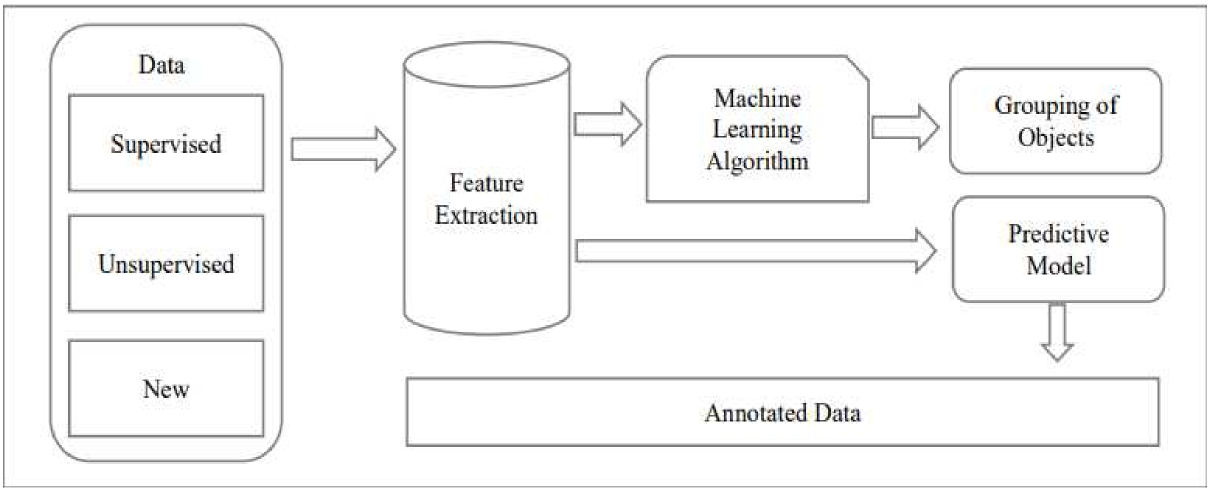


Fig.1 Workflow of Machine Learning

Machine learning is a part of artificial intelligence which is able to develop machines that can learn from their surroundings, from people and from mistakes. Recommendation systems, Autopilot system for planes, translating one language to another using computers and maps are all examples of the systems which are developed by machine learning.

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 work on 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 admin will be able to check out the logs anytime for the identification of errors or logs in the system.

3.3.1.2. Hardware interfaces

- Server Side

  The PC will have at least 32bit architecture. The PC 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.

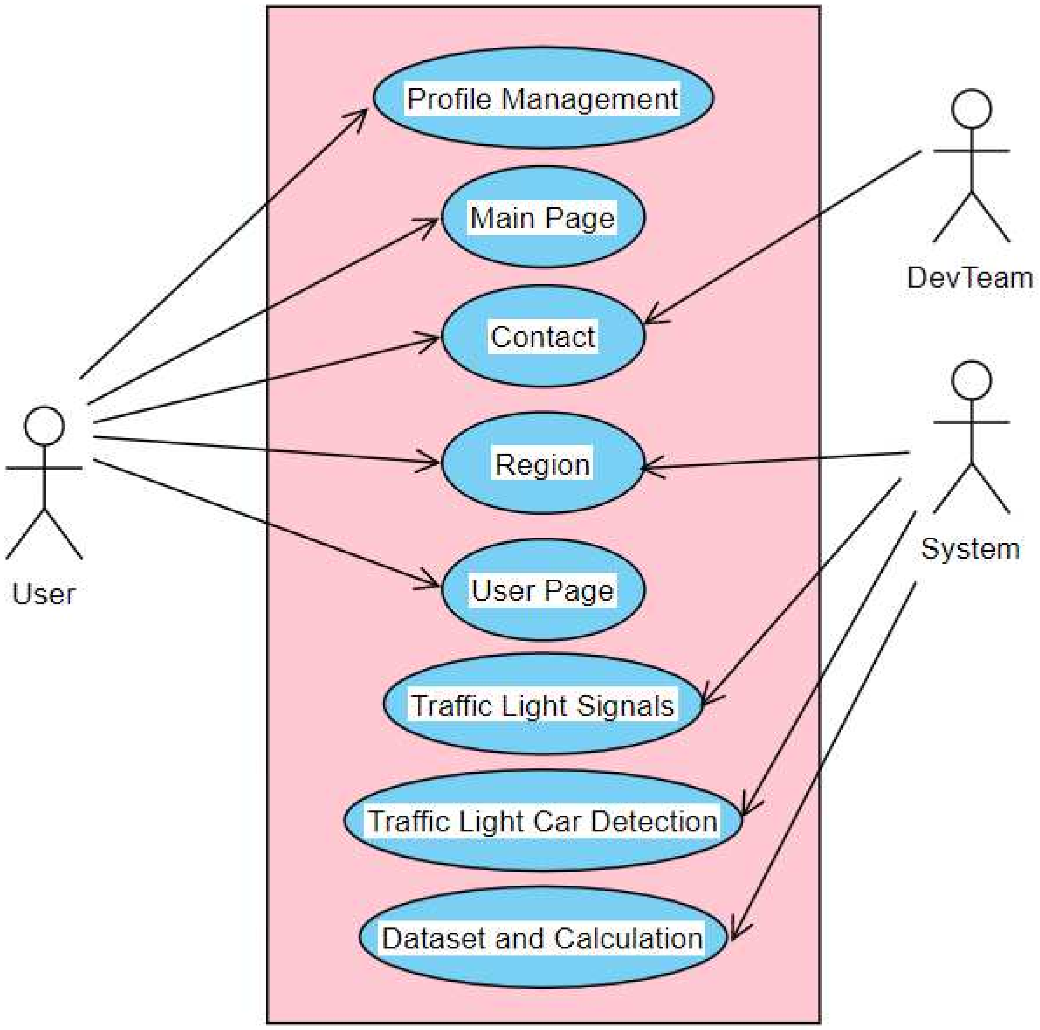
3.3.1.3. Software interfaces

- An external firewall can be attached to the application in order to prevent unauthorized access to the system. Also, a user authentication- Login system should be there to identify the System administrator.

3.3.1.4. Communications interfaces

- There are no external communications interface requirements.

3.3.2 Functional Requirements

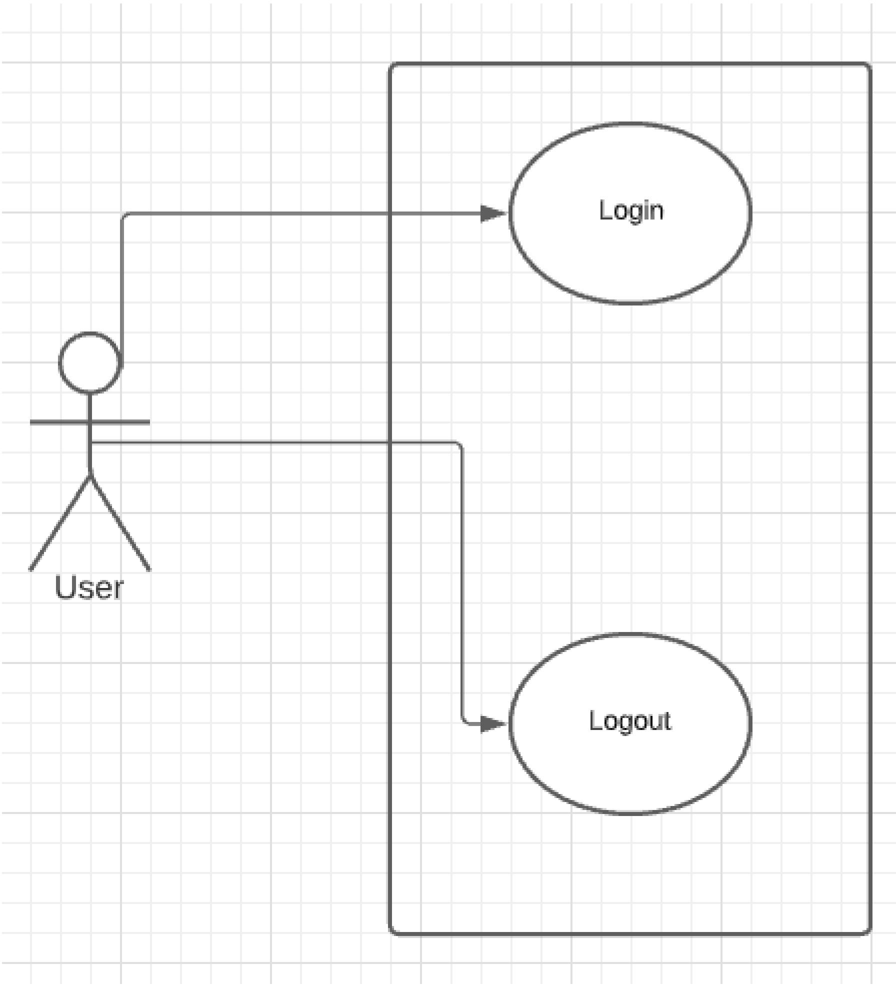


3.3.2.1. Profile Management Use Case

Use Case :

- Login as User
- Logout

Diagram :



Profile Management Use Case diagram

Brief Description :

In Profile Management diagram explains the basic operations which are related to logining 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 shall login to the system by using his/her password.

    1.1. If the password is invalid for the username, the user should re-login.

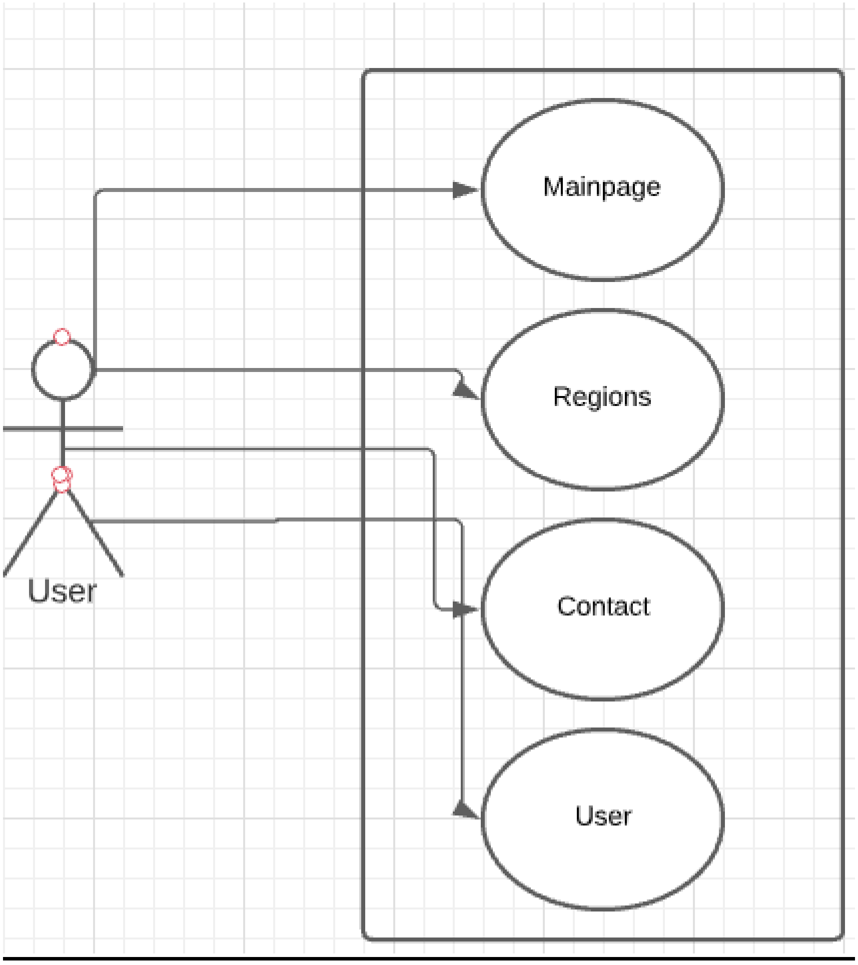
2. The user can logout from the system

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 :

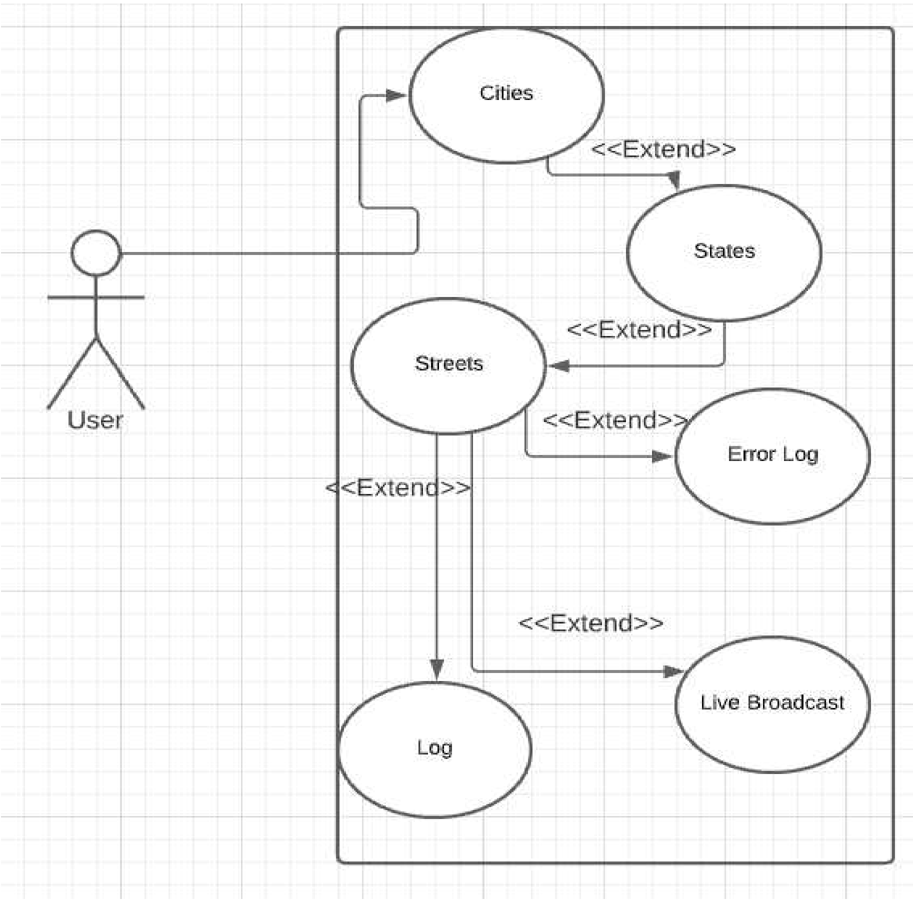
1. The user sees all the details about the system in the main page once he/she is logged in.
2. If the user clicks the Regions button , he/she will be redirected to the Region's page.
3. If the user clicks the Contact button , he/she will be redirected to the Contact page.
4. If the user clicks the User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

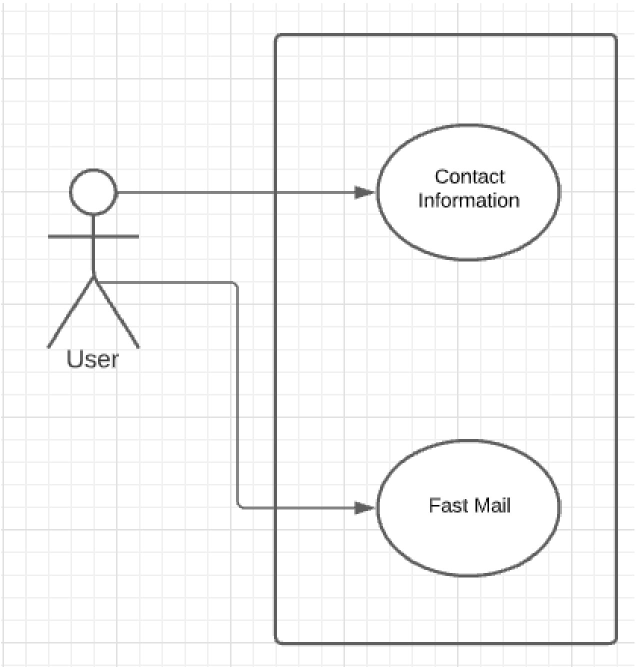
1. The user sees city, state and street options.
2. If the user clicks the City option, he/she will be able to select cities.
3. If the user clicks the State option, he/she will be able to select states.
4. If the user clicks the Street option, he/she will be able to select streets.
5. The user sees 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 :

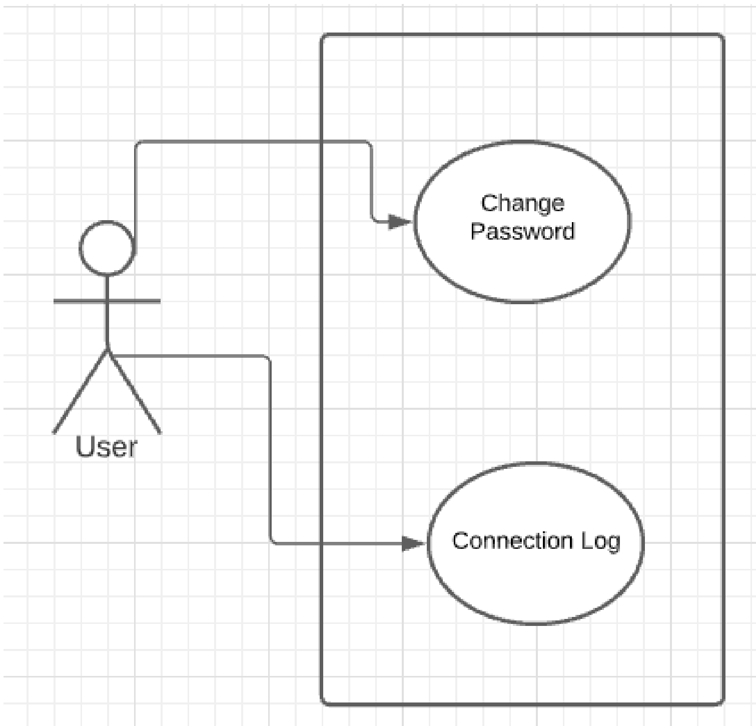
1.The user sees the Contact information and Fast mail part.
2.If the user clicks the Contact information button , he/she will see contact information about the DevTeam.
3.If the user clicks the Fast Mail button , he/she will be able to send a message to the DevTeam by typing only the email address.

3.3.2.5. User Page Use Case

Use Case :

- Change Password
- Connection Log

Diagram:



User Page Use Case Diagram

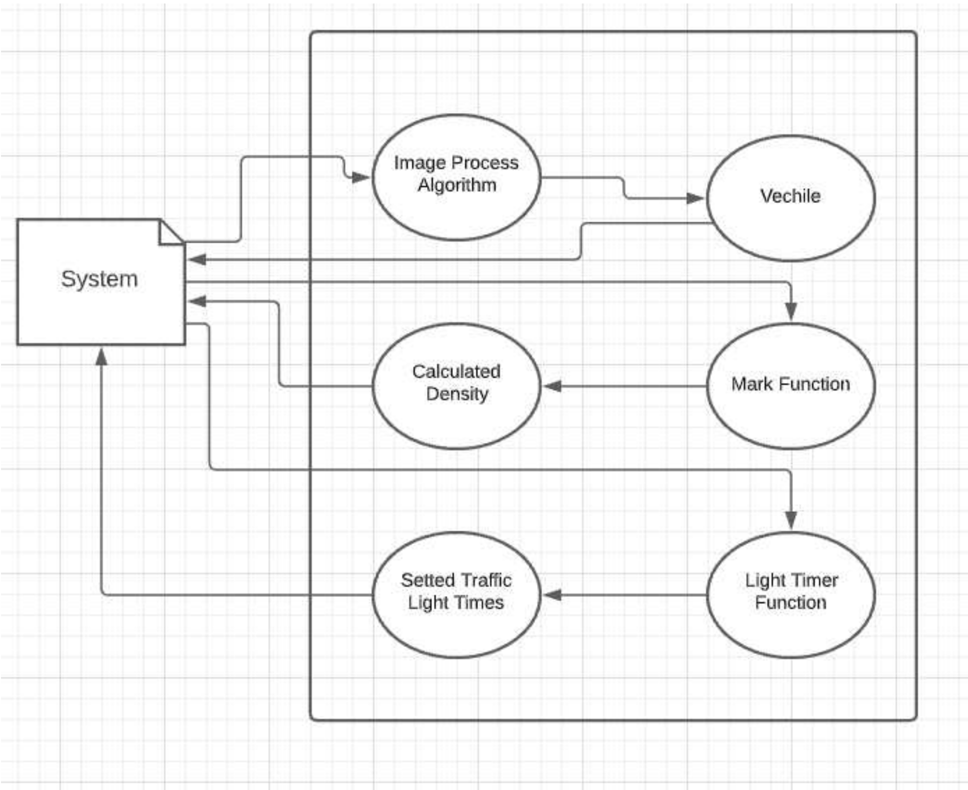
Brief Description:

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

Brief Description:

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.

Initial Step by Step Description:

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 every module must be good.
- The overall performance of the software will enable the users to work efficiently.
- System should give better resource sharing results.
- Performance of the results and data application should be efficient and fast enough.

3.3.4 Software System Attributes

There are requirements that are not functional in nature. Specifically, these are the constraints 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 system's app interface has to be user-friendly and easy 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 shall maintain data integrity by keeping backups of all updates to the database for every record transaction.

3.3.5 Safety Requirement

- Since this system affects the daily traffic population, the system should be secure enough and shouldn't allow intruders to make changes or install bugs in it. Only System administrators should be authorized to access, modify, halt and carry out any other functions.

4. Software Design Document

4.1. INTRODUCTION

4.1.1 Purpose

One of the major problems encountered in large cities is that of traffic congestion. High traffic density was caused by the influx of vehicles as a result of breakdown in other transport sectors and is most prevalent in the '+' road junctions.Several measures had been deployed to address the problem of road traffic congestion in large cities.namely among these are: the construction of flyovers and bypass roads, creating ring roads, posting of traffic wardens to trouble spots and construction of conventional traffic light based on counters. These measures however, had failed to meet the target of freeing major '+' intersections resulting in loss of human lives and waste of valuable man hours during the working days. The main purpose of this Smart Traffic Systems Project Document is to provide convenience to drivers by minimizing the duration and end of traffic lights.

4.1.2 Scope

This document describes the implementation details of the Smart Traffic Systems Program and also describes a solution to road traffic problems in large cities through the design and implementation of an intelligent system; based on AI and ML. By using Smart Traffic System Program users 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.

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

The remaining chapters and their contents are listed below.

Section 2 is the Architectural Design which describes the project development phase. Also it contains a class diagram of the system and architecture design of the system which describes actors, exceptions, basic sequences, priorities, pre-conditions and postconditions. Additionally, this section includes an activity diagram of the scenario generator.

Section 3 is Use Case Realization. In this section, a block diagram of the system, which is designed according to use cases in the SRS document, is displayed and explained.

Section 4 is related to the Environment. In this section, we have shown the sample frames of the environment from the prototype and have described the scenario.

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. ARCHİTECTURE DESIGN

4.2.1 Simulation Design Approach

To automate traffic signal monitoring systems, we are building a system using artificial intelligence, Machine Learning and computer vision technologies which outperform together.

Cameras mounted on every traffic signal record thousands of hours of footage 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 make the prediction on traffic situations.

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 imagine traffic at each street and control the traffic lights accordingly 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 determine traffic situations by observing everyday data. Sometimes during overcrowding situations, signals based on hard-coded time interval techniques do not provide excellent results. This is because of the fixed time interval, every street can have a different traffic situation every hour even every minute which fails time-based signals and can result in traffic bottlenecks.

Applying machines to do this job can be very effective and reliable. The proposed system analyses predicting future traffic and current traffic that help to make decisions on which time and where a signal release operation should be performed. The proposed system will be a synchronised intelligent traffic signal monitoring which will improve the way of traffic monitoring and enhance the way we are dealing with other problems and the situations related to transportation. 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. Fig X shows the proposed model architecture

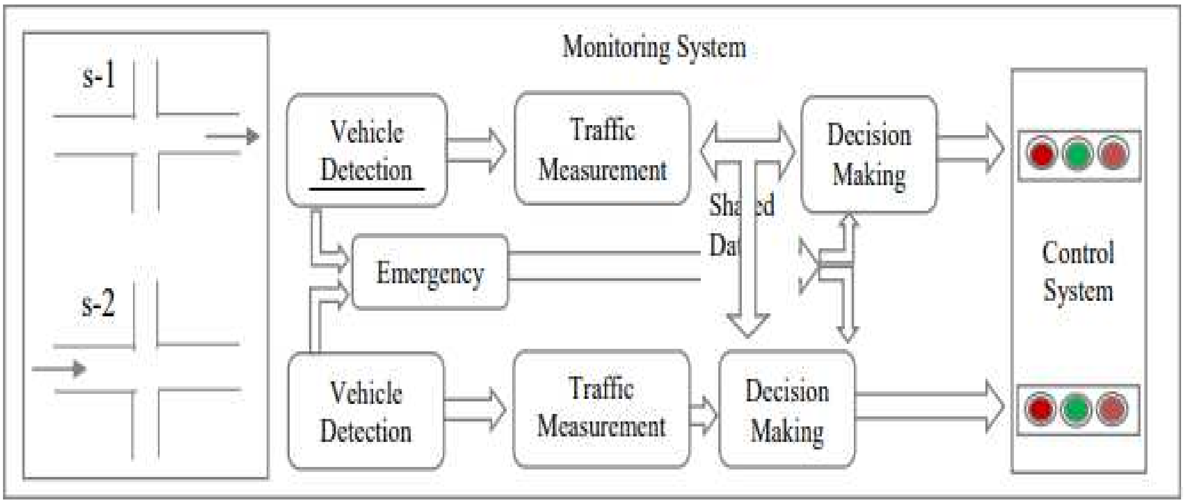


Fig - 1 Simulation Design

4.2.1.1 Class Diagram

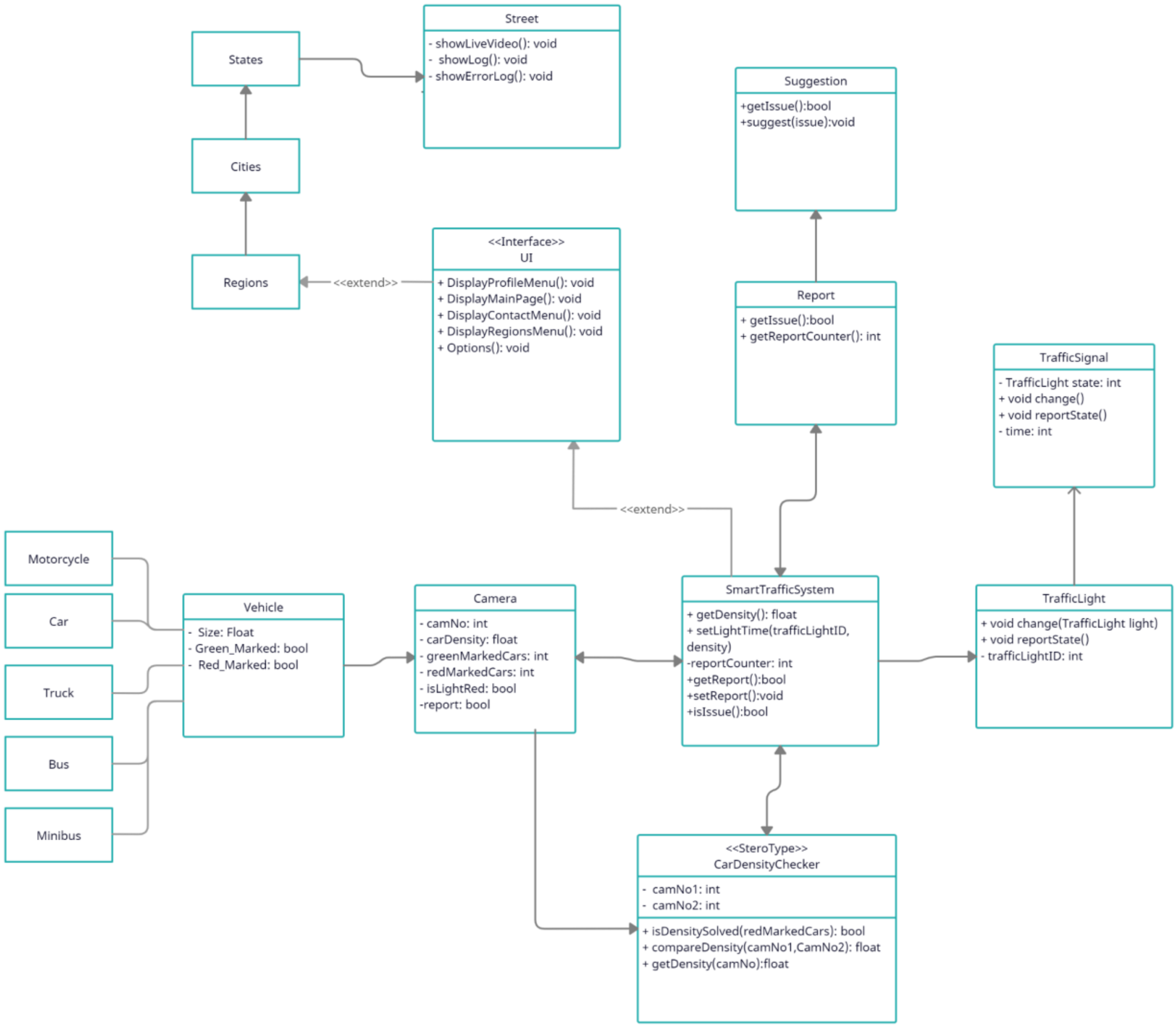


Figure 2: Class Diagram of Smart Traffic System project

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

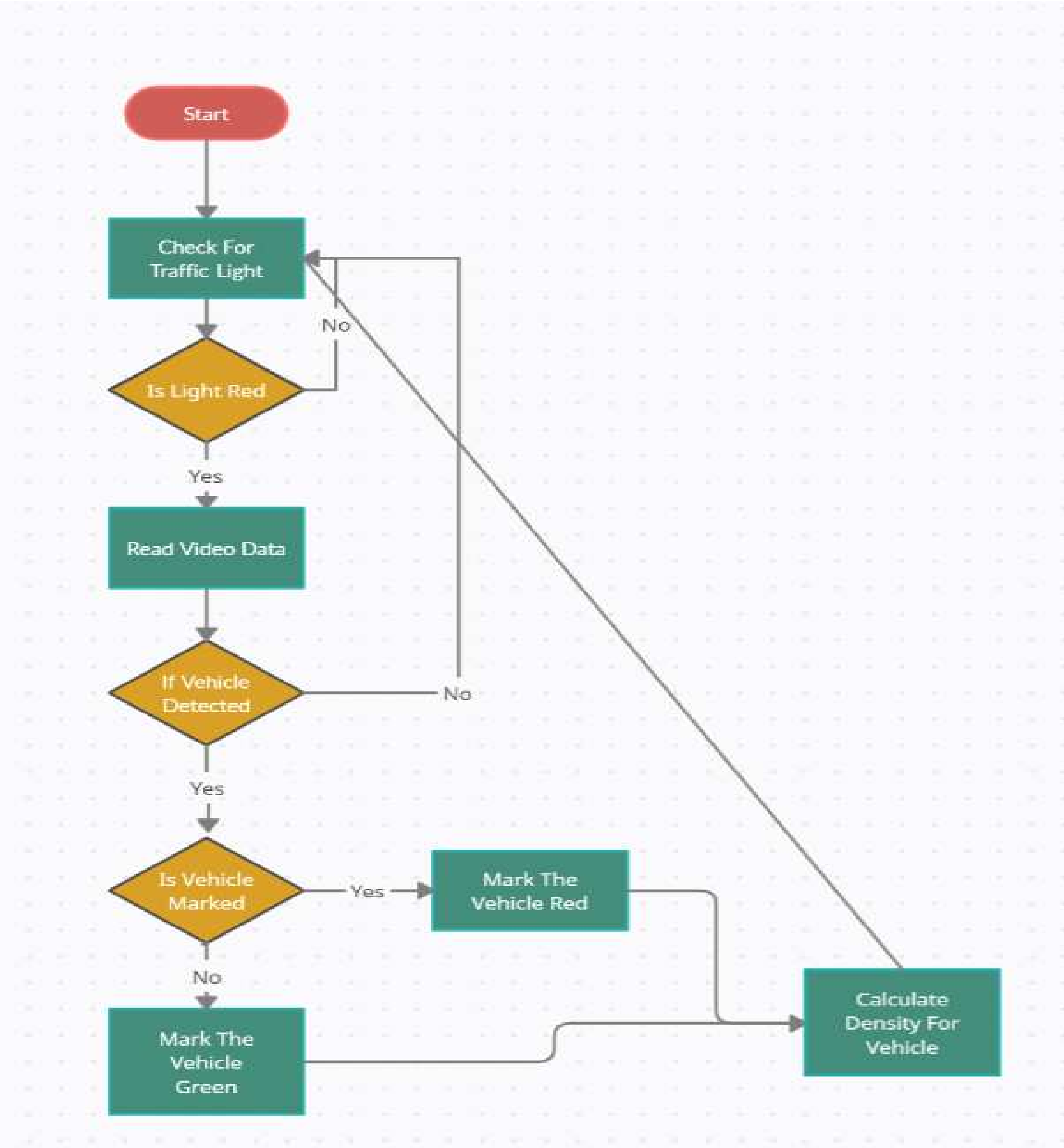


Figure 3: Activity Diagram of Light Check and Vehicle Detection

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:

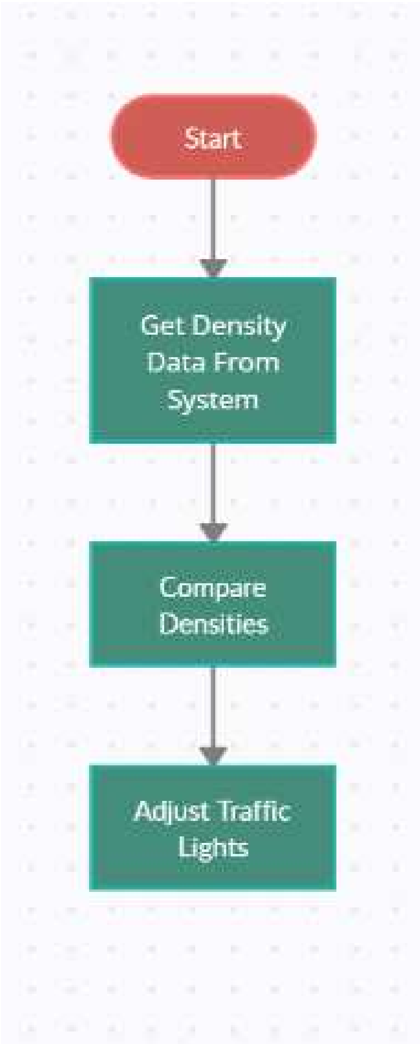


Figure 4: Activity Diagram of Setting Light Time

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

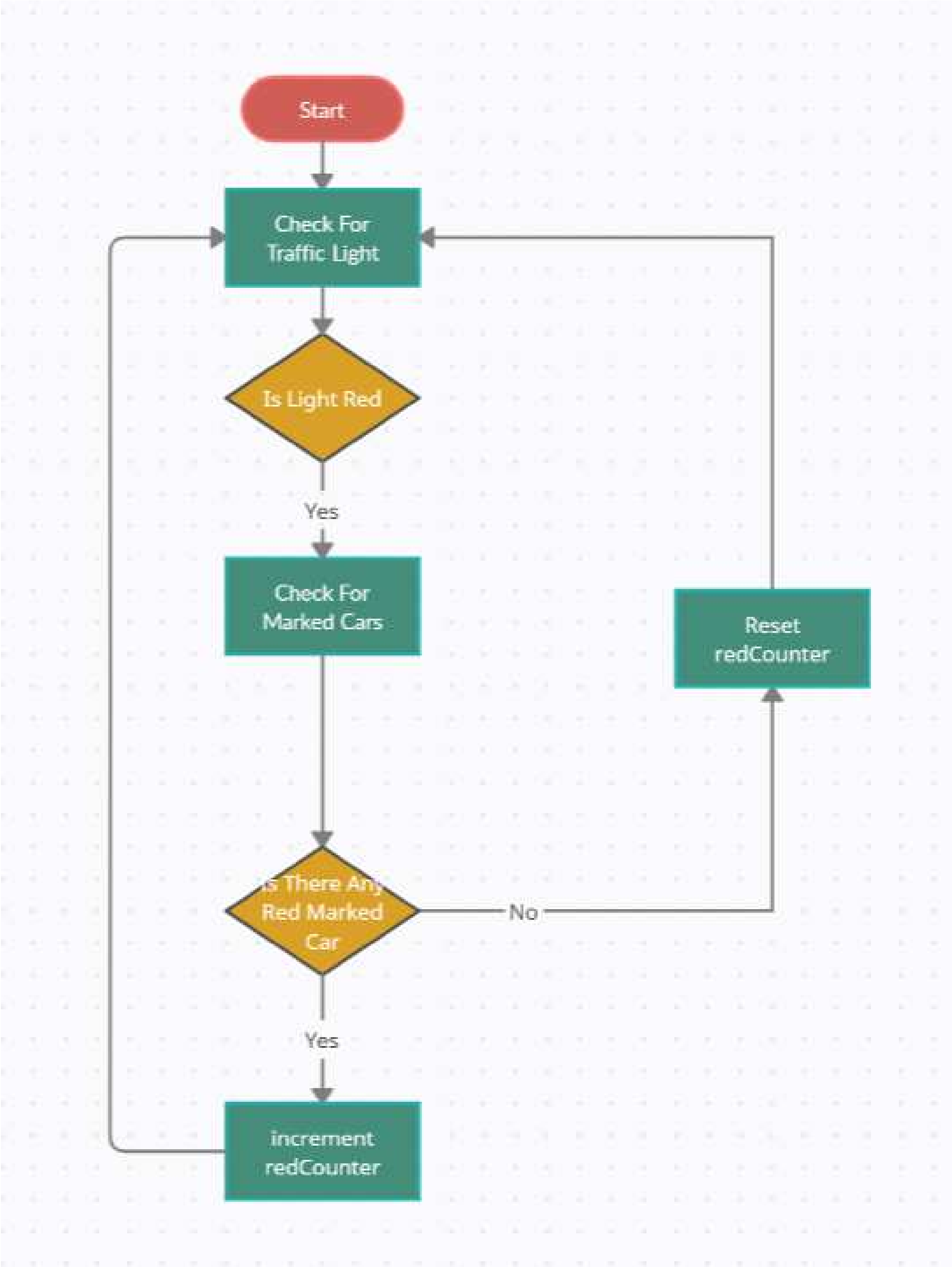


Figure 5: Activity Diagram of Red Mark detection

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

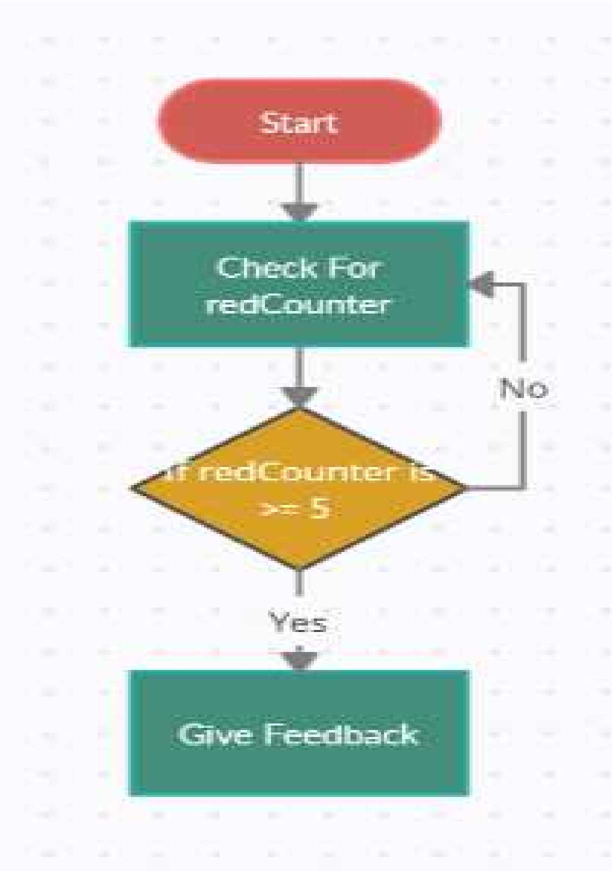


Figure 6: Activity Diagram of Give Feedback

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

Steps of Troubleshooting

Machine Learning steps that will be done consist of three stages: pre-processing, training process and the process of identification. Broadly speaking, the system will be designed as follows:

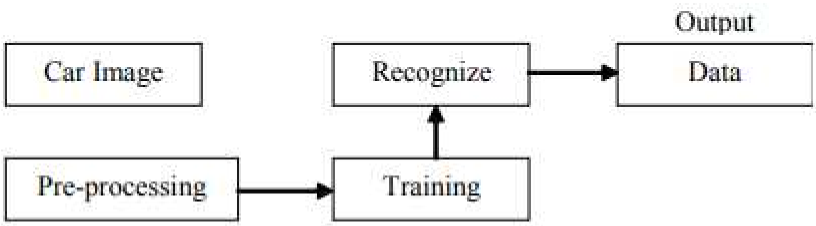


Figure 7: Diagram of Car Recognition System

The first stage reading camera inputs the vehicle images into the system. After the image is inputted, the system will read and perform a pre-processing process. After the preprocessing was successfully executed then entered the training stage, which is the system learning phase in order to recognize an object (in this case car type). After that the next process will be done by a recognized process, namely the process of introduction of a system that has previously been trained. After all the process is successful the result will show the introduction of the type of car.

4.3.1 Datasets

Datasets which are used for machine learning research in supervised learning set labels to the data. The vehicle detection system has a decent amount of datasets for more accurate results and decreasing the percentage of the error that might occur during the machine learning process. The dataset is divided into a training set and a testing set. The ratio between the training and the testing set might vary depending on the difficulty of the task.

4.3.1.1 Training dataset

The training dataset represents the set of training samples. It is a collection of positive and negative data. Usually they are grouped by the same number of components or features. .The training dataset is an essential part of the future feature extraction needed for vehicle detection.

Car image example from training set

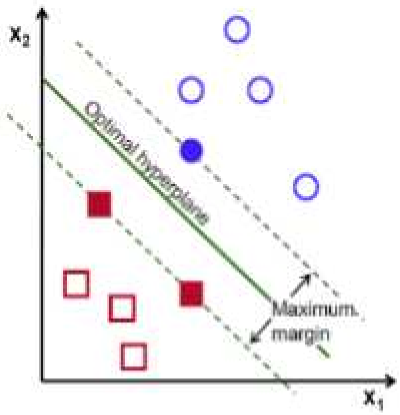
Non-car image example from training set

4.3.1.2 Testing dataset

The testing dataset is defined as a data which provides the unprejudiced evaluation of the final model which is fit to the training dataset. It is the gold standard which is applied in practice after the model is trained completely.

4.3.2 Training and classification

Classifier training is the most vital step of detecting the car, based on the feature set, for the computer to learn to recognize whether the vehicle is in an image or not. It is called classifying them as class labels, for example vehicle and non-vehicle. (OpenCV library) There are two main steps to achieve the desired goal: training and classification itself. Training is the process of taking the known content, which belongs to specified classes, which results in creating a classifier on this content. Classification is the next step which includes taking the training set and applying it to the unknown content. The output results depend on the quality of the training process, which is iterative while the classification is a one-time process to run on unknown content.



Support Vector Machine (SVM) is one of the most popular supervised binary classification algorithms to perform classification whether it is a vehicle or not after being trained with a prepared dataset. Figure 6 shows the main idea lying behind the SVM. There are two class exits (e.g. vehicles and non-vehicles) represented by two different types of dots. During the training the algorithm is provided with a vast number of examples from two classes. The algorithm’s task is to separate those classes. Figure 6 clearly shows the line between the two classes, which is called optimal hyperplane. The support vector machine will find the suitable plane that separates a maximum of two classes.

Figure 8: Support Vector Machine (SVM)

4.3.3 OpenCV library

OpenCv (Open Source Computer Vision Library) is a powerful open source software library. It consists of more than 2500 optimized computer vision and machine learning algorithms. The main purpose of using those algorithms is face detection and recognition, object identification, human actions classifications in videos, and camera movements tracking. The library is developed in C and C++ to enhance the computation efficiency, supported by main operating systems.

Vehicle detection system is created based on the OpenCv library. The main purpose of using it is transforming the received frames, by analyzing them and making a representation of them. Class cvSVM::CVM was used to achieve the classification of two classes: vehicle 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

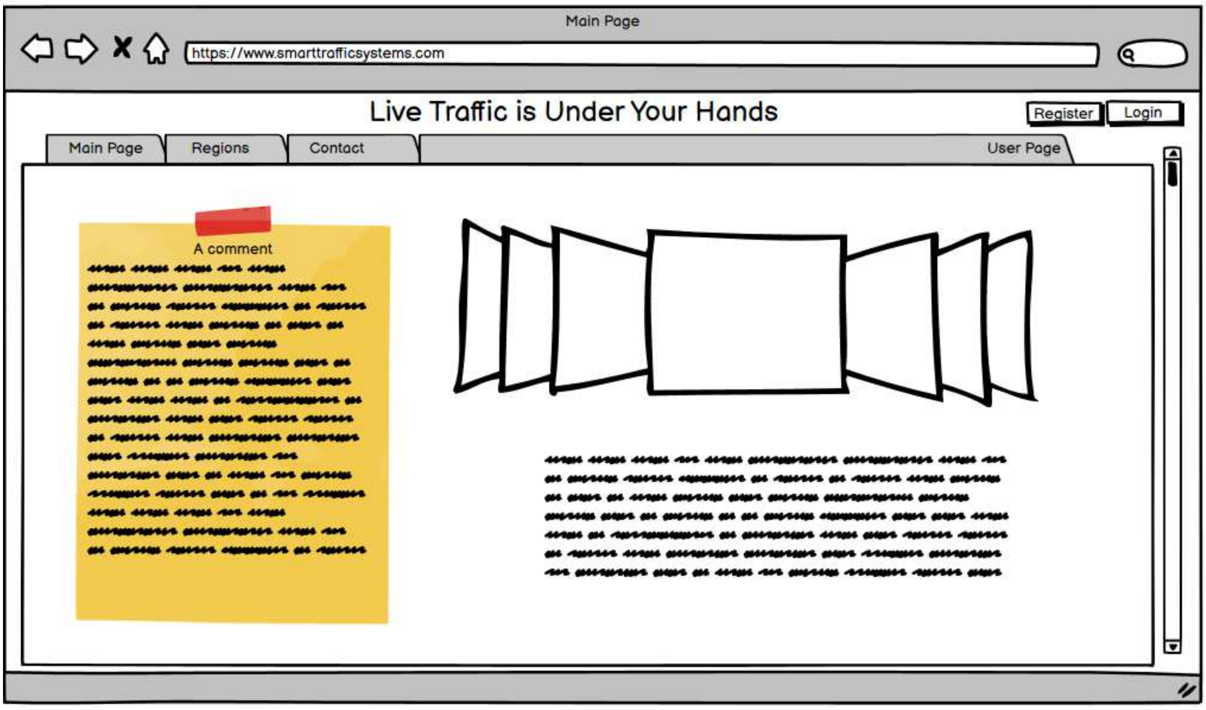


Figure - 9 : UI Design of Main Page (No Login)

After Log In

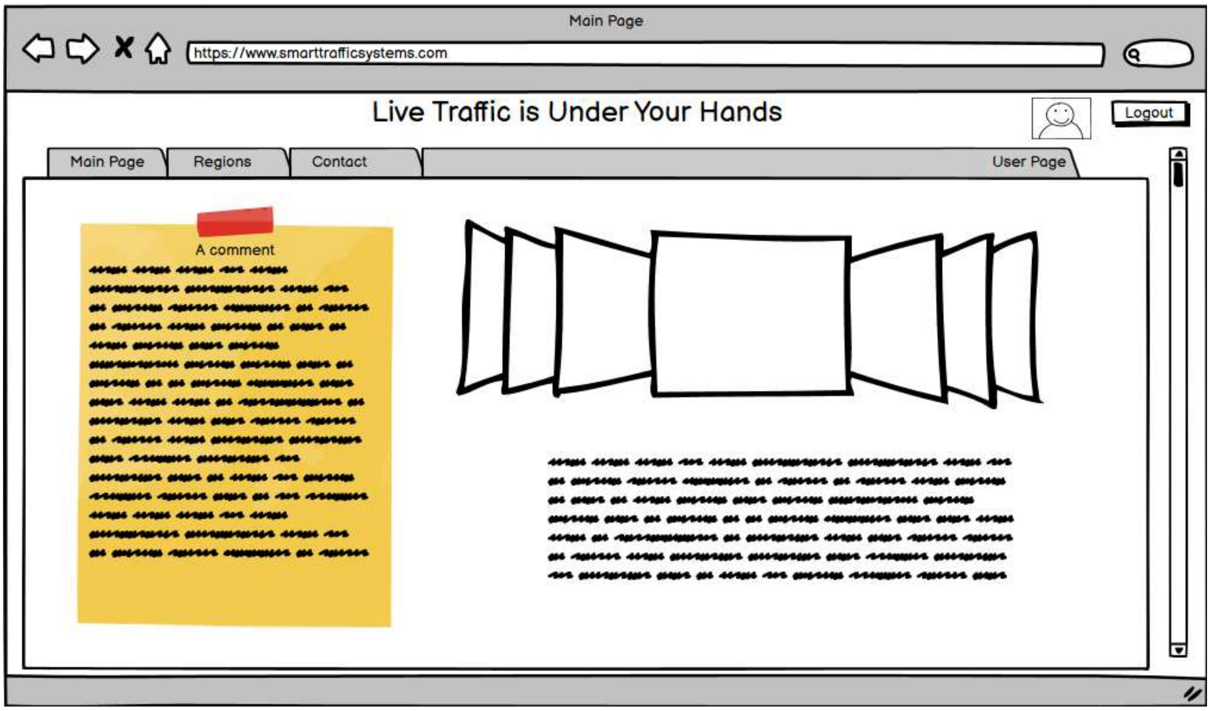


Figure - 10 : UI Design of Main Page (User Login)

Regons Dropdown Menu

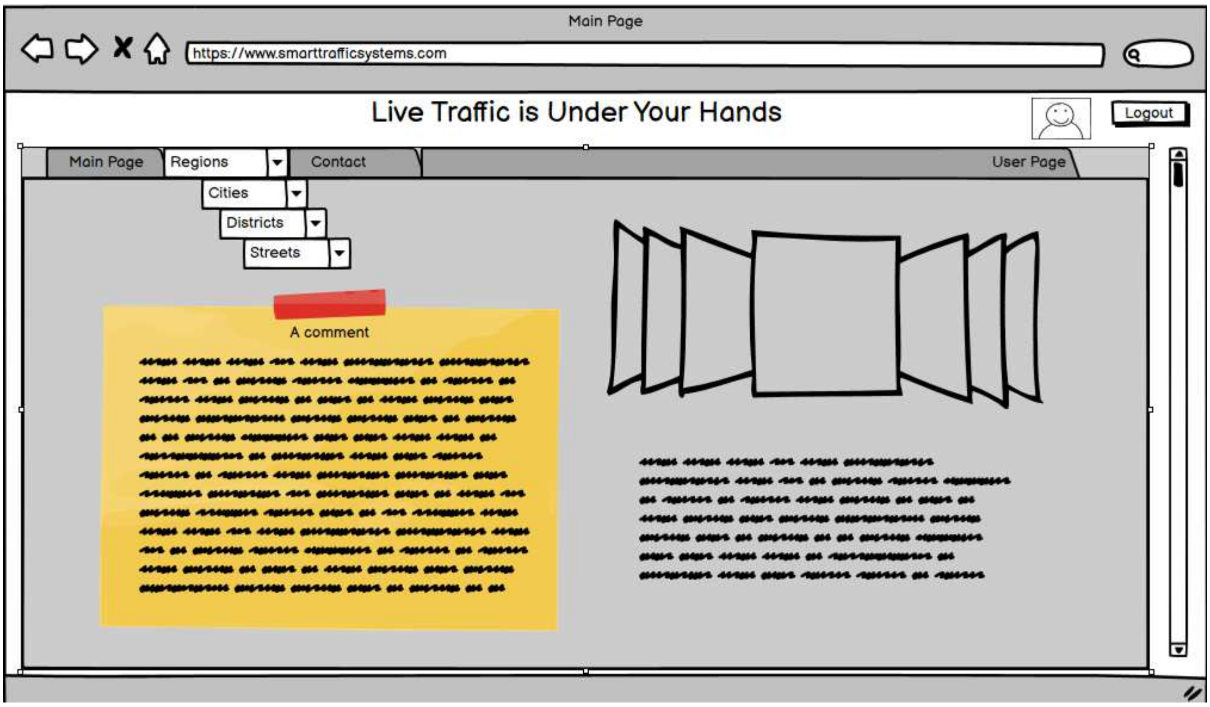


Figure - 11 : UI Design of Regions Page

Contact Page

Figure - 12 : UI Design of Contact Page

User Page

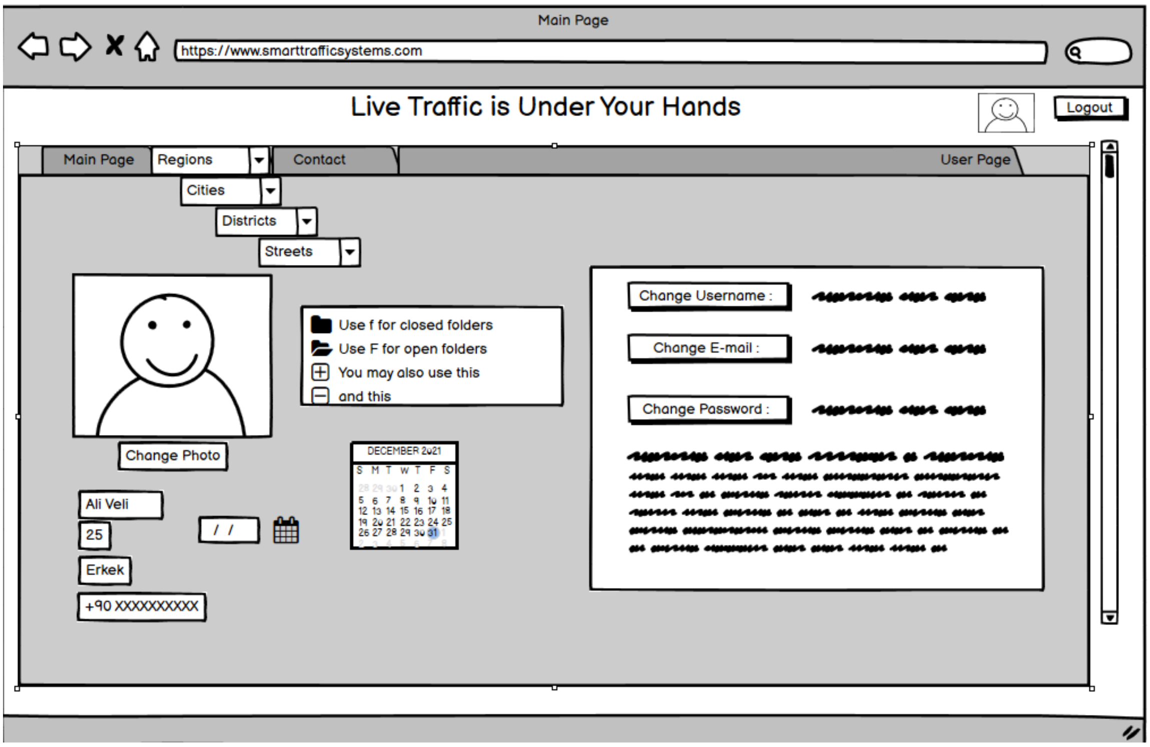


Figure - 13 : UI Design of User Page

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.