A
Dissertation On

IoT-driven Smart Parking system and Traffic Management for Urban cities

Submitted in Partial Fulfilment of the Requirement
For the Award of Degree of

Master of Technology

In

Software Technology

By

Santosh Kumar Gupta
University Roll No. 2K14/SWT/512

Under the Esteemed Guidance of
Mr. Rajesh Kumar Yadav
Assistant Professor, Computer Science & Engineering, DTU

COMPUTER SCIENCE & ENGINEERING DEPARTMENT
DELHI TECHNOLOGICAL UNIVERSITY
DELHI – 110042, INDIA

DELHI TECHNOLOGICAL UNIVERSITY

DELHI-110042

DECLARATION

I hereby declare that the thesis entitled "**IoT-driven Smart Parking system and Traffic Management for Urban cities"** which is being submitted to the **Delhi Technological University**, in partial fulfillment of the requirements for the award of the degree of **Master of Technology in Software Technology** is an authentic work carried out by me. The material contained in this thesis has not been submitted to any university or institution for the award of any degree.

DATE:

SIGNATURE:

SANTOSH KUMAR GUPTA

2K14/SWT/512

DELHI TECHNOLOGICAL UNIVERSITY

DELHI-110042

CERTIFICATE

This is to certify that thesis entitled "**IoT-driven Smart Parking system and Traffic Management for Urban cities",** is a bona fide work done by Mr. Santosh Kumar Gupta (Roll No: 2K14/SWT/512) in partial fulfillment of the requirements for the award of **Master of Technology Degree in Software Technology** at Delhi Technological University, Delhi, is an authentic work carried out by him under my supervision and guidance. The content embodied in this thesis has not been submitted by him earlier to any University or Institution for the award of any Degree or Diploma to the best of my knowledge and belief.

DATE:

SIGNATURE:

MR. RAJESH KUMAR YADAV
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING.
DELHI TECHNOLOGICAL UNIVERSITY, DELHI 110042

ACKNOWLEDGEMENT

I take this opportunity to express my deep sense of gratitude and respect towards my guide **Mr. Rajesh Kumar Yadav, Department of Computer Science & Engineering.**

I am very much indebted to her for her generosity, expertise and guidance i have received from his while working on this project. Without his support and timely guidance, the completion of the project would have seemed a far-fetched dream. In this respect I find myself lucky to have my guide. He has guided not only with the subject matter, but also taught the proper style and techniques of documentation and presentation.

Besides my guide, I would like to thank entire teaching and non-teaching staff in the Department of Computer Engineering, DTU for all their help during my tenure at DTU. Kudos to all my friends at DTU for thought provoking discussion and making stay very pleasant.

SANTOSH KUMAR GUPTA

M.TECH, SOFTWARE ENGINEERING

2K14/SWT/512

ABSTRACT

In new generation, this is an era of Internet of things (IoT) and Artificial Intelligence (AI). Present time the concept of smart cities more popular in all over the world. IoT is an important for achieving the idea of smart cities. For making better in infrastructure of any city, many efforts are ongoing in this IoT sector. Also, in any city we are facing many problems such like congestion of traffic, very less parking facility and safety when travel during road.

As world is moving fast and current transport system is not providing seamless transportation system users. Extra jam on traffic produced to delays while we are going to home or work or any other places. This is also a reason of time delay and fuel burn. Currently, parking is also a very serious problem for each person who has vehicle.

With use of IoT, we proposed a solution which help to solve these problems. In this thesis, we have developed an algorithm which was helpful for traffic congestion and problem of parking system.

For reading car number plate, we proposed an algorithm of License Number Plate (LNP) detection on the basis of advanced algorithm for convergence of grey scale image & morphologic effect.

For traffic management, reducing time in Intersection, we proposed advanced techniques with use of AI (Fuzzy system) and IoT sensors. Maximum number of vehicles can have passed in given time duration.

Also, we presented an IoT based on cloud adding parking system. System used IoT module for using to monitor the availability of parking space.

In future, we will propose a security and privacy of data for Smart parking and for traffic control enhancement of AI techniques for vehicle overspeed detection and auto detection of car accident and contact SOS number.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 GENERAL CONCEPTS

Today, peoples are connecting through internet. IOT connects all things, so it is called “the Internet of Things”. Things Connected: communicated between physical world and information world.

IoT’s concept started when communication devices are start identification of things. Tracking, controlling and monitoring could be able to easy when we used remote computer with the help of IoT. Things may be any gear, VR, clock, TV, washing machine, mobile and any electronic system which able to connect. If it can behave like alive via commutation, sense and computation then these merged devices able to interact with any objects or any persons.

As we know that any developer can create and host their application with help of cloud computing. When cloud merge with IoT, then it works as a perfect system. In this way it enhanced a new technology named as Cloud of Things (COT) with the merging of both IoT and cloud.

With help of COT, we can control, access and monitor the things .As cloud provide high accuracy, that result on real time, we added or removed any node in IoT system.

1.1.1 Approach for IoT -driven smart parking system (ISPS):

This ISPS system provided Reservation-based parking facilities and location of parking spot inside parking area. Reservation from remote location is very helpful for saving time, fuel and noise less. We proposed a solution using reservation, which built on advanced technology.

It aimed check parking spaces and provide a better parking service to drivers and provider. It is also helpful for communication among different service providers and fulfill the need to any particular driver.

It reduced driver's count who was searching for parking as well. It is also helpful for traffic management. For this, we designed our system as per control signals to help driver to easily parking selection in a large parking system. It able to helpful in taken entry or exit record by using our advanced algorithm of license plate detection. Also, this system provided drivers to shortest path inside parking area during from entry gate to parking space location and from parking space location to exit gate.

1.1.2 Approach for traffic management:

For the traffic management, in our system, we used a technique based on fuzzy system added with IoT system, which was very helpful for traffic lights control on an intersection. Its saved time if in any direction road is free (no traffic), then it decreased remaining time of traffic light and provided run traffic to other direction road. Its main goal to pass maximum vehicles in a given time or reducing waiting time.

1.2 MOTIVATION

In present time, this is an era of digitalization and smartness. In the world every product become smarter than previous time. If we talk about smart city, smart home or any electrical devices. Every person wants to relax in own life and think about many different thinks for become smarter or use the things as a smarter way. IoT become very helpful to achieve this goal. The concept of smart home is already implemented in homes. We are seeing that each electronic device is communicating with other electronic device with help of internet. Also, AI implement with IoT. That result, it helps to one step more to become smart home.

Now, work is ongoing to make cities smart. There are many projects ongoing inside the world. The dream of smart cities cannot achieve without making smart parking system and smart traffic management.

Currently, we are facing many problems [2] like more traffic jam, limited parking space or not proper utilization of parking space, road safety, toll collection, etc. The idea of smart cities seems to true with help of IoT.

This was the main motivation point of our thesis. Here we had worked for the below points:

1. Algorithm enhancement for License number plate detection
2. IoT based smart parking system model with reservation -based with use of cloud DB.
3. A technique based on fuzzy system interrogated with IoT for traffic management.

1.3 RELATED WORK

1.3.1 Related to Smart parking system

At present time, there was many related works on parking system. We focused on implementing an IoT based parking system which use a wireless sensor network. This system detects parking related information and provide parking service at real time. Other then it, we introduced advanced algorithm of LNP detection and direction of parking spot inside parking area, which provided us a smart system to helpful for allocate different parking areas. There were many related working systems as mention below:

1.3.1.1 Cyber-Physical System (CPS)

As per the current research, CPS was an important area. It helps to provide helps and problems, as per the research between academia and industry [25]. CPS was a physical based engineered system that are responsible for monitored, coordinated and controlled operations, which was used for computing and communication core [20]. With the help of this system, we able to find that how humans control and stabilized interaction with physical things of world.

CPS had great social and economic impact in competitiveness [15]. Researchers had different focus view in area of CPS, which included transport system [22], defenses, healthcare systems [16], etc.

Result of survey in US, the transportation consumption of total energy reached 29.2% in 2008, which was the highest record since 1973 and 88% of these trips are by car in U.S. [3]. We can say that daily commuting represents an important role in transportation consumption. CPS's research focused on the smart transport to reduced energy consumption and provided driverless systems. In [17], researcher investigated and found the optimal solution in field of transportation for reducing strategies and identifies optimal ways. It was helpful to reduce transport energy consumption and pollution control.

Since CPS could be physical systems and real-time system [20] that could be interacting with physical world, there were many research challenges for CPS, including:

1. CPS composition of things
2. Security and robustness of system
3. Hybridge and controlled system
4. Computational capabilities
5. System with real time syncing
6. Education and training
7. Model-based and sensors based
8. Verification, validation, and certification

Transportation cyber-physical system [10][27][26][12] is an important application of CPS. It provided people to reach with security and safety. The air transportation network was based on transportation CPS. If the air traffic was suffering adverse weather and density problem, then CPS helps to provide an adaptive flight planning capability with seamless integration. In [27], the researcher told about the impact of the mobile internet for take a change the transportation cyber-physical system.

1.3.1.2 VANET-Based Smart Parking

Vehicular Ad-Hoc Network (VANET), VANET-Based was another option for smart parking systems. In [8], the authors discussed a smart parking system based on wireless sensor network (WSN) technology. WSN provided parking reservation, remote monitoring of parking and automated guidance service. This system was very helpful for find vacant parking spaces. Rongxing Lu et al. [18] introduced a new Smart PARKing (SPARK) scheme, which was based on VANET. It was very helpful for large parking lots and it provided drivers with accurate and convenient parking. It also includes intelligent antitheft protection, real-time parking navigation and friendly parking information dissemination. In [11], authors describe about the dissemination algorithm for discovery of free parking spaces via VANET.

1.3.1.3 Parking Lot Detection

The parking lot detection had been developed by Jake Reisdorff et al. [21]. Main objective of this project to providing an information about that a free parking space is available or not. It was accessible by a webpage. They used a mounted camera to take images from a static position and send images to a web server. After image processing application to detect if the parking spaces was available or not and send information to the web application. Based on information, the web application determined the number of unoccupied spaces, and show a map of parking garage with availability information.

1.3.1.4 IoT based Smart Parking System

This was developed by Khanna and Anand [1]. This project was applicable for small parking area. They provided a mobile application to easy to be booking a parking spot and payment facility option e wallet or credit card. This was not more suitable for large parking system. It was not detected any car License plate during entry and exits. No navigation option and Not a centralized system.

1.3.2 Related to License number plate detection

Currently, there are more work ongoing in this field. Automated object detection algorithm is challengeable research area. There are many techniques [7] like Edge based, Histogram, Color feature, Morphology, Neural Network, Image transform etc.

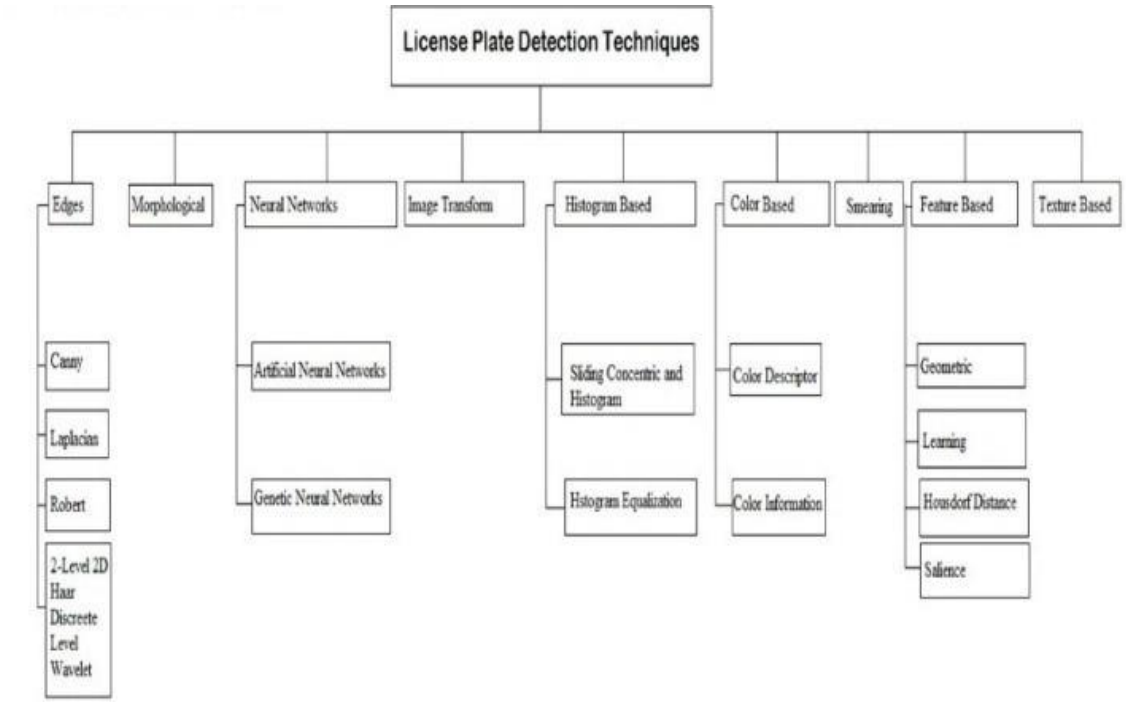


Figure 1.1: Different license plate detection techniques

We know that license number plate is a unique for each vehicle.

In IoT Driven automated License plate detection [3]. In this method taken images from digital camera sensing images to determined and picked out the highest energy frequency areas. Images might be either to pick the vehicle license plates or the vehicles out from the images. This method could be used to reduce big data volume needed to be stored in urban surveillance system.

In VLPR system [28], the removal of a license plate region from an image, which influences the accuracy of the VLPR systems.

One more way was vertical edge matching [29]. The idea was that firstly, locate the two vertical edges of a license plate, then four sides detected. It helped to extracted LNP with more accurate. It was using the contrast between the grayscale image's value, [30] and proposed a vertical edge- based license plate number recognition technique.

Morphology [31] based license plate detection method was to extract important features of contrast as guidance to search the license plates [31].

Color based methods used of the colors of the vehicle license plate. In [32], a color- based method combined with the texture characteristics was proposed to try to detect license plate from the color image.

Neural network techniques were another recognition method of vehicle license plate. It used to provide good response to the license plate images. In [33]it, genetic algorithm (GA) was use for training process and combination of the statistic features with structure features.

1.3.3 Related to Smart traffic management for intersection of traffic lights

There were many ways to determine the traffic load on any direction load, one of them was Video-based system. It used in detecting traffic parameters, which had been a focus of extensive research due to an increase in road traffic jams in intersections, see [35].

In other way, vehicle detection could be divided into either monocular-vision or stereo-vision detection. This technique used appearance-based and motion-based techniques, [36]. Monocular vehicle detection used an appearance-based technique, which was usually done by recognizing vehicles directly from images (depends on pixels). Stereo-vision systems used motion-based techniques. In it 3D information is directly measured using multi-view geometry for scene and motion characteristics. 3D points were used to tracking and characterizing static from moving objects.

Machine learning was also applied in vision-based systems. It was used for solving many problems like detection, tracking, plate number localization of vehicles, [37].

1.4 PROBLEM STATEMENT

Features considered we planned to make use of following additional features apart from the ones mentioned till now:

We enhanced on algorithm of License plate number during convergence in Greyscale image. In this, we added Distortion correction, Morphologic operation, Noise reduction before applying convergence in Greyscale image. It provided more accurate in license plate detection.

In parking system, we added License number plate detection during entry and exits gate. It helped to real time tracking in cloud database as well as helped in accurate billing. Also, provided navigation map option inside parking place.

In traffic management system, for intersection, we provided IoT based algorithm integrated with neural network (Fuzzy logic). As a result, this system detected traffic load on a path is very much accurate. Hence, this increased efficiency and accuracy of traffic road lights controller in a smart way.

1.5 SCOPE OF THE THISIS

In this Project, we had connected all parking spot of a city by using cloud-based data base. We had provided facility to driver that they reserved parking either from a remote location or form a parking space by using IoT and sensor system, which provided real time status for parking spot.

Also, using advanced algorithm of license plate detection during time of entry /exits, provided more accurate, save time and support auto billing system. Inside parking area, from entry location to parking spot or parking spot to exit location, display a shortest path.

In traffic management for intersection we provided a technique which handle a traffic system in a smart way via using AI and IoT.

The purpose of this thesis is to help a city become a smart city with use of IoT based smart parking system and traffic management. By using our proposed method (Using advance algorithm of image conversance in a greyscale image) for detection of License plate in any condition is more accurate detection. Also, AI + IoT based technique helpful for traffic control signal of intersection.

CHAPTER 2: LITERATURE REVIEW

2.1 IOT-CLOUD INTERIGATION

It is possible to building a city as a smart city with help of IoT. Main problem of smart cities are vehicles parking and traffic systems.

Currently, as increasing the numbers of car's user that result there are many problems to finding an available parking space. By enhancing parking resources, we can find the solution of this situation, but it is not the best solution because its leading to reduction in searching times, traffic congestion and road accidents. This become successful with the help of IoT based new applications such as low-cost and energy efficient embedded system. With help of sensor technology, monitoring in and around cities become easy.

We required sensors deployment on parking spot for providing real time information about available parking. It's also helpful for monitoring the occupancy and provide quick data processing. We required cloud-based database, which helped a user current available parking space on real time.

Cloud computing and IoT have together provided a big evolution. On one hand, cloud provides many advanced features like big storage, data processing and memory and time saving. On the other hand, IoT provides the features connecting to real world. Some of advantages are given below after merging Cloud and IoT.

- Storage capacity
- Computation power
- Communication resources
- Availability
- Interoperability

2.2 FUZZY LOGIC

Fuzzy Logic (FL) is a part of AI. FL is a method which is used for arranging human reasoning. The Idea of FL provided the way of decision making in humans, which included all intermediate possibilities between digital values YES and NO. It used to handle the concept of partial truth,

2.3 WIRELESS SENSOR NETWORK (WSN)

WSN is a network of devices where all the data can have collected from different sources and proceed in the destination with help of gateways and wireless connections. It is using in many applications like Transports, Health, Agricultures, Energy control, smart indoor and other systems.

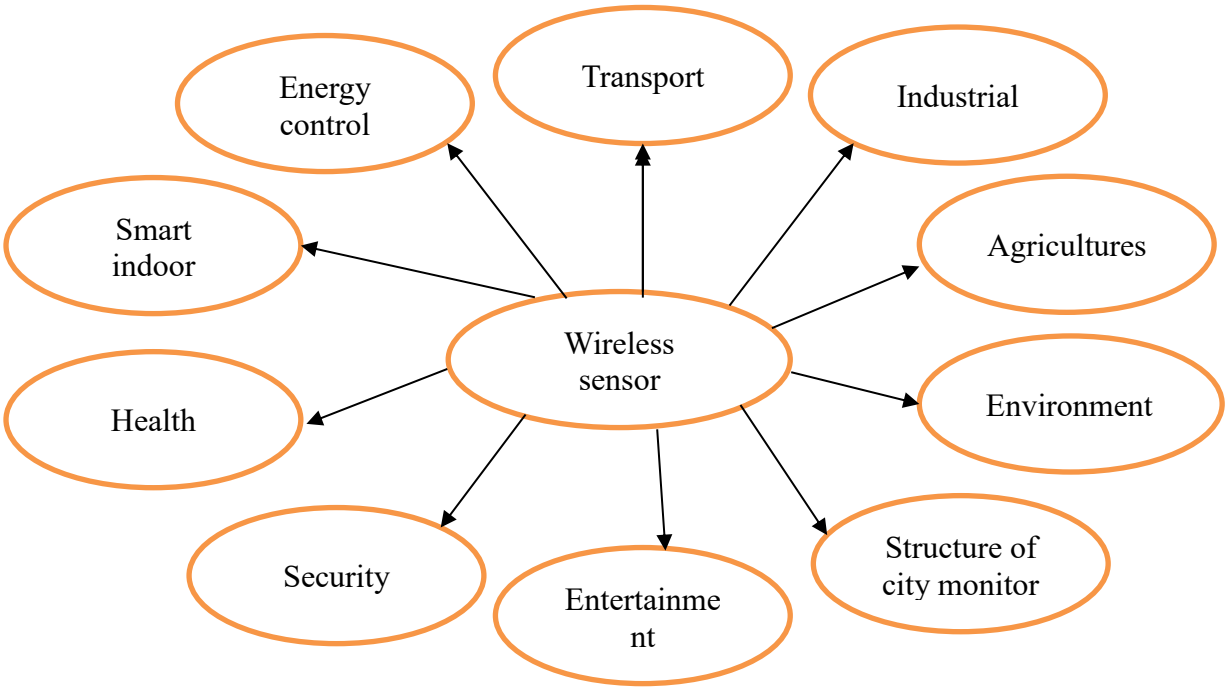


Figure 2.1: Application of Wireless Sensor network (WSN)

2.4 SHOREST PATH ALGORITHM INDISE PARKING SYSTEM

In the large parking system, when there are one entry gate and more than one Exit gate and multi colors LED lights implemented in road side and upper road side. From entry gate to available parking spot or Parking spot to exit gate, there are many ways to reach there, but we need to find shortest path. In our work, we used Breadth-first search (BFS) algorithm to find path. After finding path, this path started to blink with LED light & driver easy to be navigated own car.

2.5 MORPHOLOGIC OPERATION

We use an image processing operation that is called Morphology, which is used for process images based on shapes. It has been applied on a structure element to an input image, responsible for creating an output image of the same size.

In our thesis, we used each pixel in the output image based on a comparison. we constructed a morphological operation using selection of size and shape of nearest node.

There are two operations named as dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries.

Dilation operation is output pixel of maximum value of all the pixels in the input pixel's neighborhood.

Erosion operation is output pixel of minimum value of all the pixels in the input pixel's neighborhood.

CHAPTER 3: PROPOSED WORK

3.1 ENHASMENT OF IOT – DRIVEN AUTOMATIC OBJECT DETECTION

We proposed enhancement of existing algorithm [3] by adding below things after taking image:

Distortion correction:

When we had taken any image from camera then captured image was distorted due to camera's lens. It was an important to correct distortion to detect the vehicle license number plate. If we took black box images of the vehicle, then observed more distortions. The reason behind it was that using of lens to reduce blank spots by black box images [5].

Noise Reduction:

By using Gaussian blur filter- It used effect in graphics software to be reduced image noise and reduce detail. The filtered image I^* can be find out as in equation (3.1),

$$I^* = \frac{1}{Wx} \sum_{y \in S} G\sigma s \left(\left|\left|x - y\right|\right|\right) G\sigma r(||\ Ix - Iy\ ||) Iy, \quad (3.1)$$

where Wx = $\sum_{y \in S} G\sigma s \left(\left|\left|x - y\right|\right|\right) G\sigma r(||\ Ix - Iy\ ||)$ is normalization factor, σ_s and σ_r used for the filtering amount, S: considered window, $G_{\sigma s}$ and $G_{\sigma r:}$ Gaussian function kernels, and I: Intensity value.

Grey scale convergence:

Advanced formula, Z= (2.99R + 5.87G +1.14B)/10

Where Z – Grey Scale value, R- Red color, G- Green color, B-Blue color

Morphologic operation:

Morphologic had Erosion and dilation Operation.

Dilation operation: Broken characters by expanding the objects.

See equations (3.2) and (3.3) for dilation and erosion respectively

$$S = X \oplus Y[k; l] = \max \{X[k - n; l - m] + Y[n; m]; (k - n; l - m) \subset X; [n; m] \subset Y\} \quad (3.2)$$

Erosion operation: Responsible for shrinking the objects

$$S = X \ominus Y[k; l] = \min \{X[k - n; j - m] - Y[n; m]; (k - n; j - m) \subset X; [n; m] Y\} \quad (3.3)$$

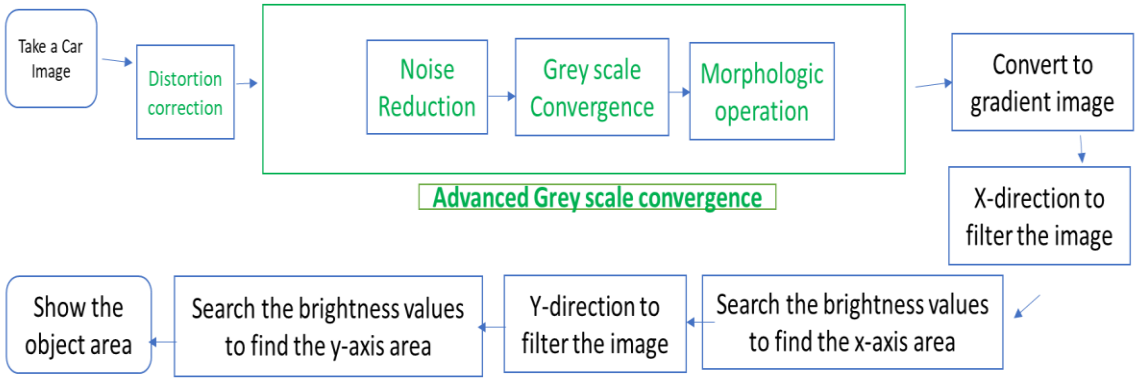


Figure 3.1: Enhanced IoT – driven automated object detection algorithm

3.2 ADVANCED TRAFFIC LOAD DETECTION TECHNIQUE FOR INTERSECTION

We proposed this technique by integration of algorithm [traffic load detection from fuzzy system + IoT based motion sensor]. This algorithm provided best accurate detection of traffic load of one direction and help to smooth and traffic lights control.

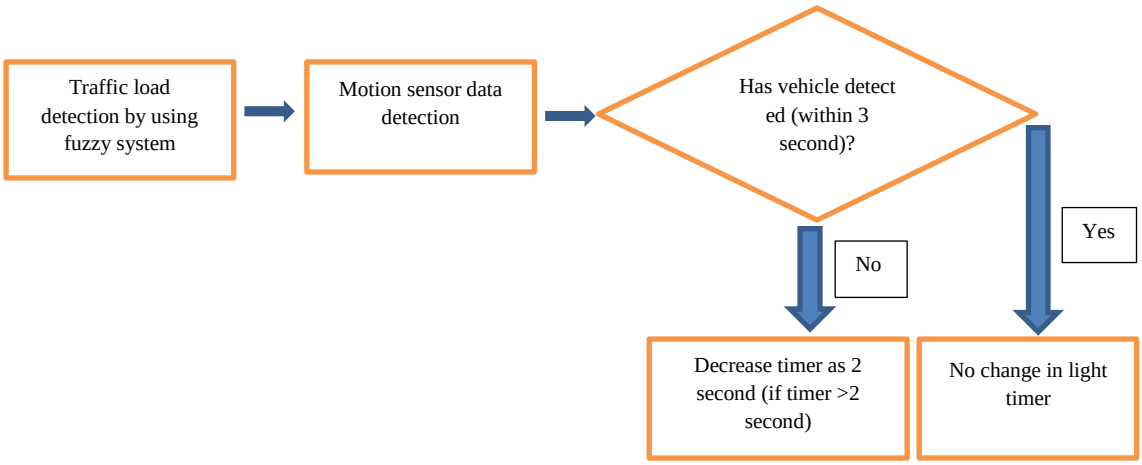


Figure 3.2: Advanced traffic load detection techniques for intersection

3.3 ENHASMENT OF IOT–DRIVEN AUTOMATIC OBJECT DETECTION

We proposed adding below point, that was Reserved -based parking system and navigation inside parking area for easy direction finding with help of shortest path and easy tacking of vehicles by using LNP detection from above method.

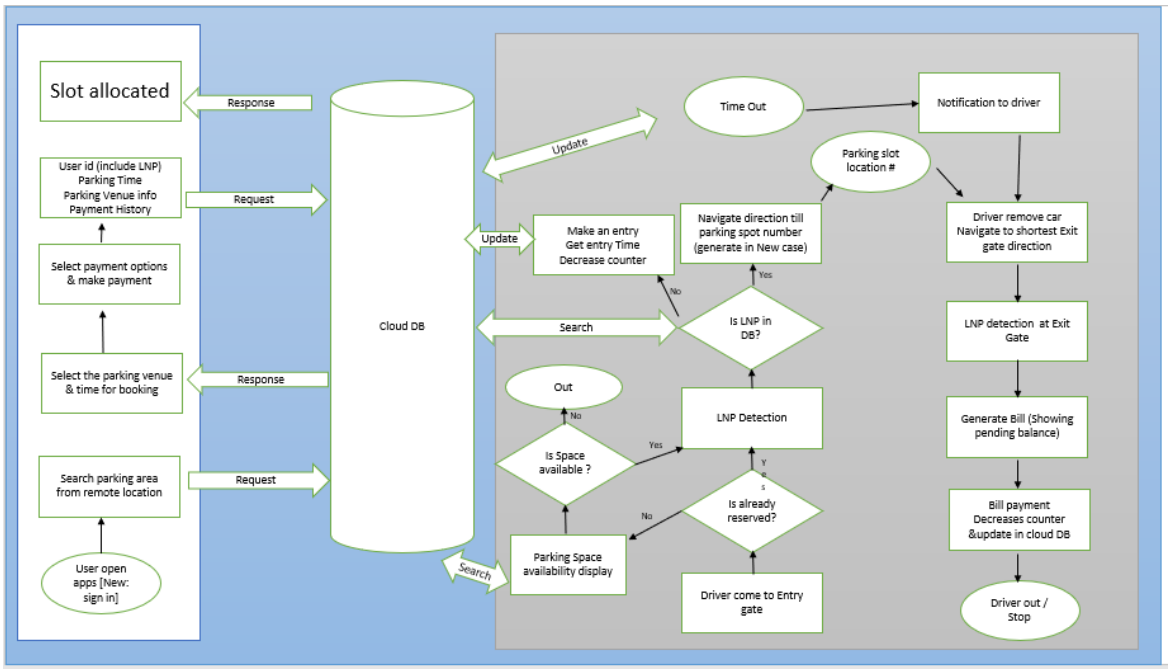


Figure 3.3: IoT based smart parking system flow diagram

CHAPTER 4: METHODOLOGY

4.1 FOR LICENSE NUMBER PLATE DETECTION–ALGORITHM ENHANCEMENT

In this section of thesis, we enhanced automated object detection algorithm as proposed by Hu & Ni [3].

As per existence algorithm working was depended on taking out the highest energy frequency. It helped to take out either vehicle license number plates or the vehicles out from taken images. It was helpful to reduced big data storage in surveillance system.

We used below 4 operations before **Convert to gradient image:**

4.1.1. Distortion correction [5]:

When we had taken any image from camera then captured image was distorted due to camera's lens. It was an important to correct distortion to detect the vehicle license number plate. If we have taken black box images of the vehicle, then observed more distortions. The reason behind it was that using of a wide lens to be reduced spots by black box images [5].

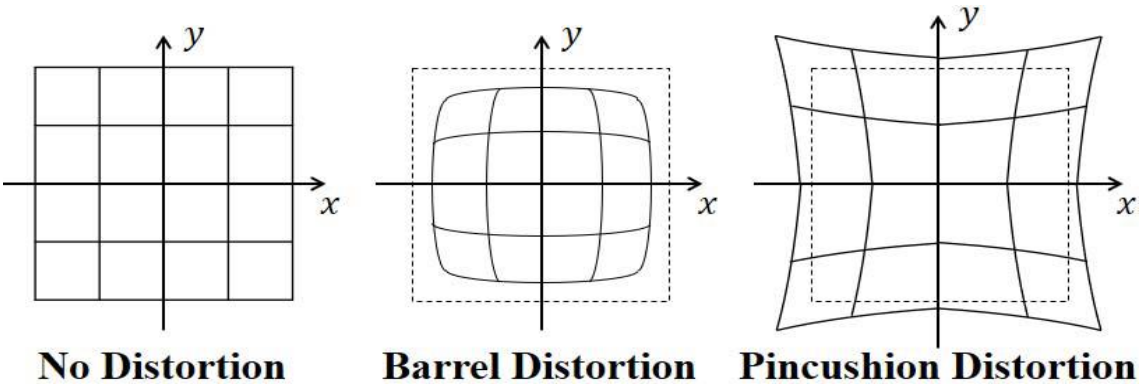


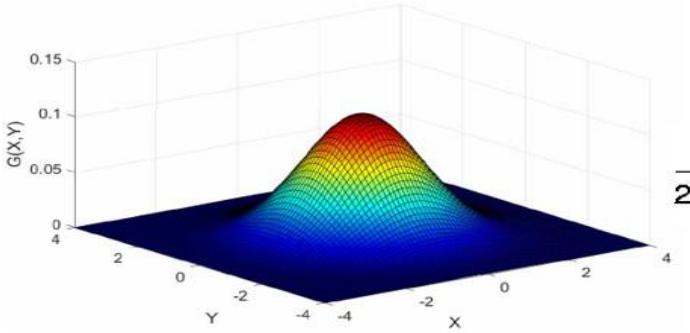
Figure 4.1: Different lens distortion's examples

There are 4 types of distortion: Concave and convex lens were spectrum, decentering and radial distortion. Radial distortion was most outstanding part of lens

distortion correction. We corrected the radial distortion using minimizing intersection ratio between coordinates and straight lines. We resolved the distortions using warping image with a reverse distortion. We a determining that which distorted pixel corresponds to each undistorted pixel.

4.1.2. Noise reduction [4,5]:

Image noise might be occurred during converting light into voltage in an image input device using CCD. Main reason to produced noise was digitals circuits of digital camera, temperature, etc. Reduction of noise was important to remove error in detections of LNP. Here, we used a Gaussian blur filter for reduction of noise.



$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Figure 4.2: Gaussian blue filter for noise reduction

Gaussian function for image blurring is used by Gaussian blur filter. In this process, we applied the transformation of image's pixel.

Equation 4.1 is showing the Gaussian function in two

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{2\sigma^2}} \quad (4.1)$$

where x: distance from origin distance in the horizontal axis and y: distance from the origin distance in the vertical axis r. σ: the standard deviation of the Gaussian distribution.

In Figure 4.2, we had set weighted average value for each neighborhoods pixel. We provided low weights if increased distance between neighboring pixels and original pixel. In the same way, we provided high weights decreased distance between neighboring pixels and original pixel. Thus, the Gaussian blur filter responsible for reducing noise which generate randomly.

4.1.3. Grey scale convergence – Advanced formula:

Here, we replaced Grey scale convergence formula to advanced formula used in paper [4].

Formula proposed by Hu & Ni [3], Z = (3R +6G +B)/10

Used formula in our thesis [4], **Z= (2.99R + 5.87G +1.14B)/10** [More accurate]

Where Z – Pixel's greyscale value

For each pixel : R – Red color, G- Green color, B- Blue color

4.1.4. Morphologic Operation [4]:

It used for separation of individual elements, and merger of disparate elements

Dilation operation: Broken characters by expanding the objects.

See equations (4.1) and (4.2) for dilation and erosion respectively

$$S = X \oplus Y[k; l] = \max \{X[k - n; l - m] + Y [n; m];(k - n; l - m) \subset X; [n; m] \subset Y\} \quad (4.1)$$

Erosion operation: Responsible for shrinking the objects

$$S = X \ominus Y [k; l] = \min \{X [k - n; j - m] – Y [n; m]; (k - n; j - m) \subset X; [n; m] Y\} \quad (4.2)$$

Table 4.1: Compression of all algorithms for object detection

Technology	Background	Environment	Light's Condition	Picture quality	Color's Model	Accuracy
Based on color feature	More depend on proper location of number plate	Illumination should not be skewed	Good in day & night images but bad in fornon-uniform light's images	Binary images	RGB > HIS	80.00%
Based on Geometry	Uniform	Illumination should uniform	More contrast	Both binary and digital images	Greyscale	82.00%
Histogram	Uniform	Better in varied weather	Best for poor illumination & different light condition	Grey scale 640*480 pixels digital and binary image	HIS	82.50%
Based on Edge	For high quality result sharp edges require	NA	More sensetive for noise and light intensity	352*288 Images 30f/sec Video	Greyscale	91.00%
Neural Network	Good for uniform background and bad for non-uniform	Good for uniform illumination	Additional light in day time and refelecting material in night time	Digital image : 320*240 Optimized resolution image : 1600*1200 aspect ratio	RGB to Greyscale	94.00%
Morphology	Better for multiple backgroud	Good work for almost every	Good for low light conditions	Static Image	Greyscale	96.00%

4.2. FOR ADVANCED TRAFFIC LOAD DETECTION ALGORITHM FOR INTERSECTION

Here, we used AI based and IoT Based algorithm for detection the traffic load and help to decide traffic light that there were no more vehicles in that path, that result:

If Path A traffic light was Green (showing time 20 seconds left) and our system provide data that no more vehicle in this path, then traffic light reduced as Yellow lights (2 seconds) and provided green signal to Path B after 2 seconds. That result saving of 18 seconds in a smart way with double accuracy cheeked.

As we know that Fuzzy system able to be synced with human calculation. It was a part of Artificial Intelligence.

If we denoted 0 as absolute false & 1 as absolute true. Then fuzzy system able to be provided data tents to 1 like 0.8, 0.9, or near to 0 like 0.2, 0.3, 0.1 etc.

Fuzzy system was using membership function to determine that Extension or termination of green light. Our main aim was to maximize the vehicles crossing within a time during. For this accurate result we added IoT system in it. To implement it, we added motion sensors in Zebra slot of road (3 Meter).

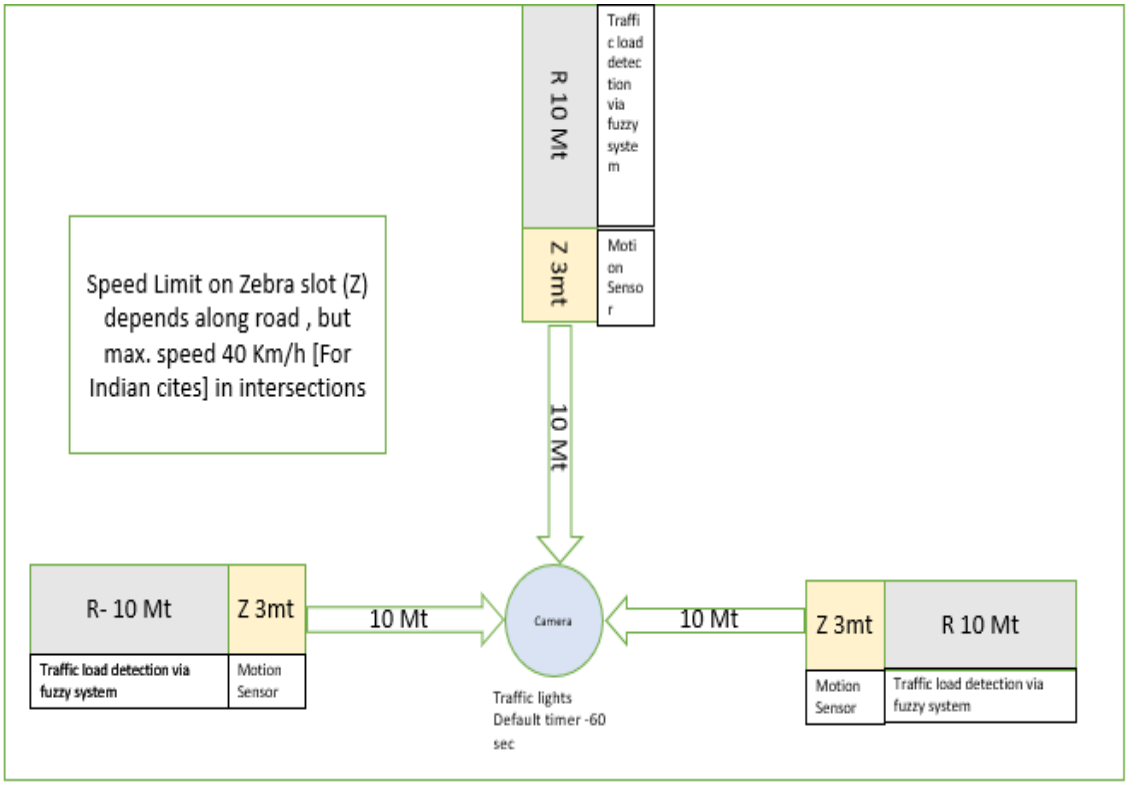


Figure 4.3: Fuzzy system and IoT system add in intersection

We have taken an example: As per figure 4.3: Max. speed limit during intersection is 40 km/h and Zebra crossing length is 3 mt, then time taken to cross zebra crossing is: 0.27 Second. Let an example of very rough case, suppose a person cross this Zebra crossing @ 5 km/ h, then taken time to cross it: 2.16 Seconds.

In our thesis, we had taken a waiting time of 3 second. if motion sensor was not detecting any motion till 3 second then, it terminated Green lights. For extension if more traffic detected by fuzzy system. Then we extended time as membership function Δt. In our case we have taken time for extension is 5 seconds.

This both methodology merged together and proving a better result in simulation of MATLAB.

4.3. FOR IOT BASED SMART PARKING SYSTEM

In this part, we added LNP detection technique in entry and exits gates. Also, enhanced parking system for the whole city and Navigation of direction inside the parking system.

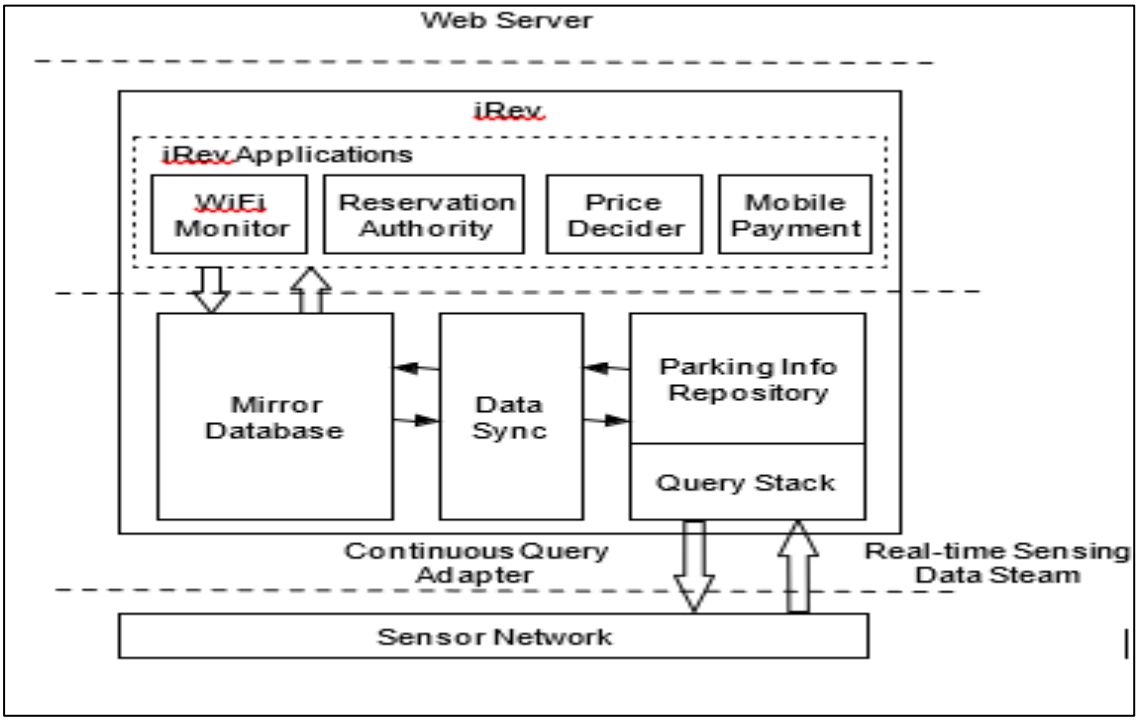


Figure 4.4: Smart parking system architecute

How to reserve a parking space by driver, we have described its methodology here:

LNP detection – This method already discussed in upper section. We have used it.

Navigation system – For providing the direction inside parking area we were using shortest path (from BFS).

We know that in our proposed system, we had implemented LED lights in each direction continuously with multiple lights. As software detected shortest path and provided instruction, then LED lights on to showing direction.

CHAPTER 5: EXPERIMENT AND RESULTS

5.1 TOOL USED

The purpose of simulation is that to importing the smart parking system and traffic's management, we simulated this as per parking behaviors of different user and implemented it to related parking system.

For this, we used Java (JDK1.8) and visual studio to develop the main components, shortest path calculation and for database access, used MySQL. For the sensor data, we have use dummy data in MS excel. For traffic management system, we simulated our work with the help of MATLAB.

5.2 ANALYSIS USING DATA SET

5.2.1 Using data set for traffic management for intersection

In intersection with 4-sided road and vehicle detectors are installed on include Normal road + zebra crossing road.

Detectors estimated approaching vehicles in a given time interval.

Suppose fuzzy variables for vehicles which is incoming denoted as A, for vehicles which is in queue denoted as Q and for extension of time if more vehicles as denoted as EXT.

Table 5.1: Input Membership function data

Input Membership function data					
Membership Function for	**Phase**	**Few**	**Small**	**Medium**	**Many**
A	Green	-4 to 4	0 to 6	4 to 12	8 to 10
Q	Red	-4 to 4	0 to 6	4 to 12	8 to 10

We determined EXT with input fuzzy and time taken of output variable.

Table 5.2: Output Membership function data

Output Membership function data				
Membership Function for	**Zero**	**Short**	**Medium**	**Long**
EXT	0 to 5	5 to 10	10 to 15	15 to 20

We extended green time till 5 seconds in our proposed system using fuzzy system controlling.

A: Fuzzy rule:

After a minimum green (5 s)

If (A== few && Q == few|| small|| medium|| many)
EXT = zero;
elseif (A==small && Q ==few||small)
EXT = short;
elseif (A==small && Q ==medium||many)
EXT = zero;
elseif (A==medium && Q ==few||small)
EXT = medium;
elseif (A==medium && Q ==medium||many)
EXT = short;
elseif (A==many && Q ==few)
EXT = long;
elseif (A==many && Q ==medium||small)
EXT = medium;
elseif (A==few && Q ==many)
EXT = short;

Doing first ext. (increase green light 5 second)

```
If (A== few && Q == few|| small|| medium|| many)
EXT = zero;
elseif (A==small && Q ==few||small)
EXT = zero;
elseif (A==small && Q ==medium||many)
EXT = zero;
elseif (A==medium && Q ==few||small)
EXT = short;
elseif (A==medium && Q ==medium||many)
EXT = zero;
elseif (A==many && Q ==few)
EXT = medium;
elseif (A==many && Q ==medium||small)
EXT = short;
elseif (A==few && Q ==many)
EXT = zero;
```

As we saw that no need to more extension after second extension for green. Here we applied our IoT based motion sensor algorithm:

For example:

Max. speed limit during intersection is 40 km/h and Zebra crossing length is 3 mt, then time taken to cross zebra crossing is: 0.27 Second

Let an example of very rough case, suppose a person cross this Zebra crossing @ 5 km/ h, then taken time to cross it: 2.16 Second. In our thesis, we have taken a waiting time of 3 second. if motion sensor was not detecting any motion till 3 second then, it provided suitable direction for next green signals. Our aim was that maximum numbers of vehicle passing in a given time.

Weight, M (A): generated from the number of queuing vehicles wait for Green

Producing labels – M (A) : zero (Z) = -4 ~ 4;
low (L) = 0 ~ 8;
medium (M) = 4 ~ 12;
high (H) = 8 ~ 16.;

The rules were used to given priority to our idea means high demands during green signals. Let us suppose that 'N' is given the 'right of the way'.

M(E): East,
M(W): West;
M(S): South;
M(N): North;

Rule are following below:

If (M(W) == high)
Next Green phase = W;
elseif (M(W) == medium && M(S)&&M(E) ==low)
Next Green phase = W;
elseif (M(W)&&M(S)&&M(E) == low)
Next Green phase = W;
elseif (M(W)<high && M(S) ==high &&M(E) == low||high||medium)
Next Green phase = S;
elseif (M(W) ==low && M(S) == medium &&M(E) < high)
Next Green phase = S;
elseif (M(W)< high && M(S)< high &&M(E) == high)
Next Green phase = E;
elseif (M(W)&& M(S) == low &&M(E) == medium)
Next Green phase = E;

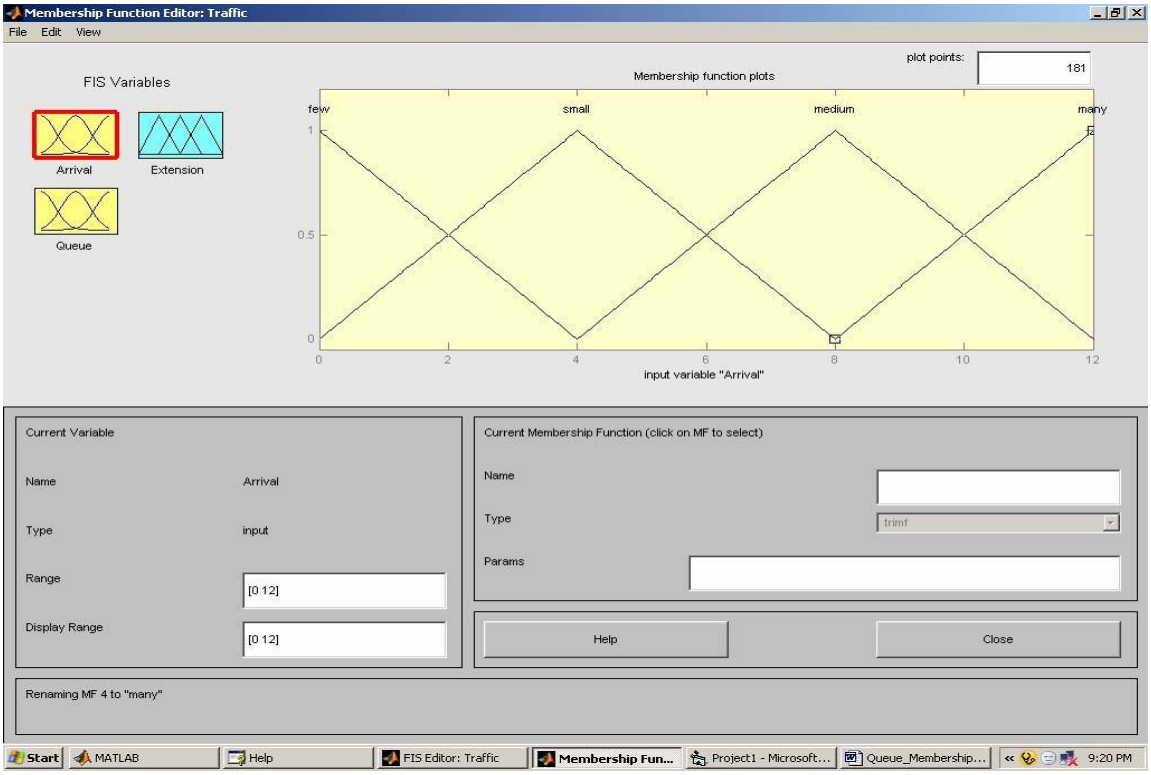


Figure 5.1: Arrival's input fuzzy variable

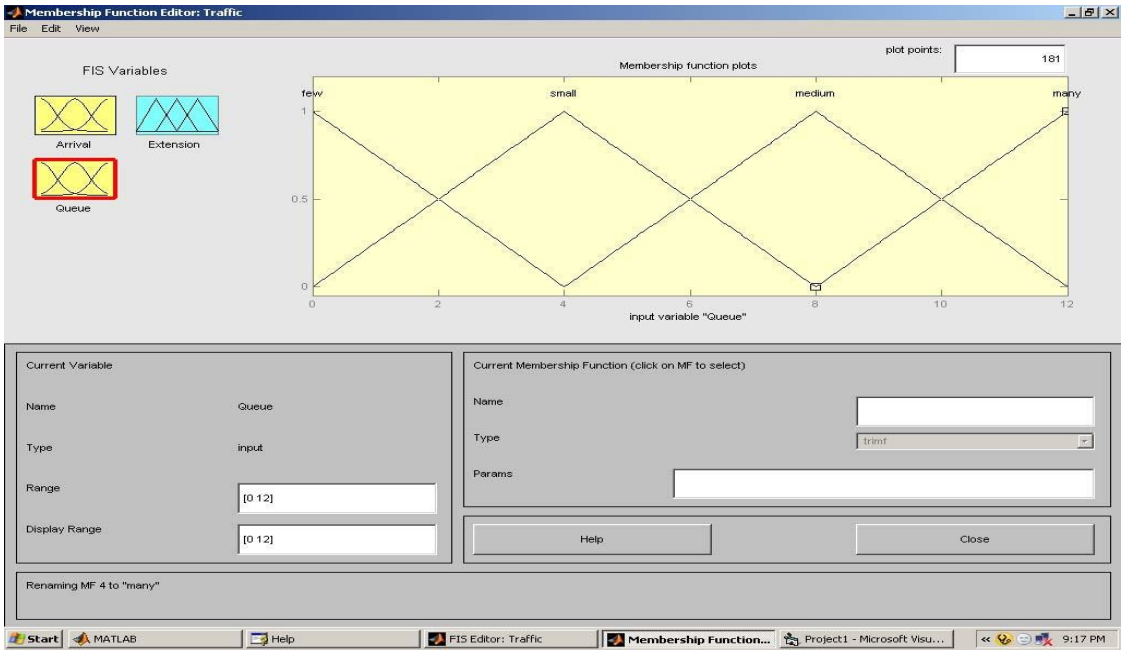


Figure 5.2: Queue's input fuzzy variable

5.2.2 Using data set for IoT based smart parking system

1. For all parking system in a city:

For evaluation our proposed parking system's performance, we created some parameters are below:

Walking Distance: It was a distance which is measure starting from drivers' current location to the destination and determined where to park. In a practical life, driver wanted to park their vehicle nearby destination location. Hence, walking distance shows the happiness of drivers.

Traffic Searching: If we talk that searching of the parking space was wanted increase traffic. Our system helped drivers to pre-reserve booking and saving time. We investigated smartness and performance of proposed system using these parameters. We simulated our proposed system within parking system management plans. These results shown in next section.

Blind Search: Blind searching was used by driver, who want to park vehicle. These drivers started to search different parking area. If a driver found an available space, stop search and park there; otherwise, the driver again started search again this other parking area till not getting parking lot.

Parking Information Shared (PIS): It was showing current state of parking system. As driver shared parking information to other driver, information like parking space available in this parking area. This satiation is very much critical in busy hours, because currently very less parking spaces.

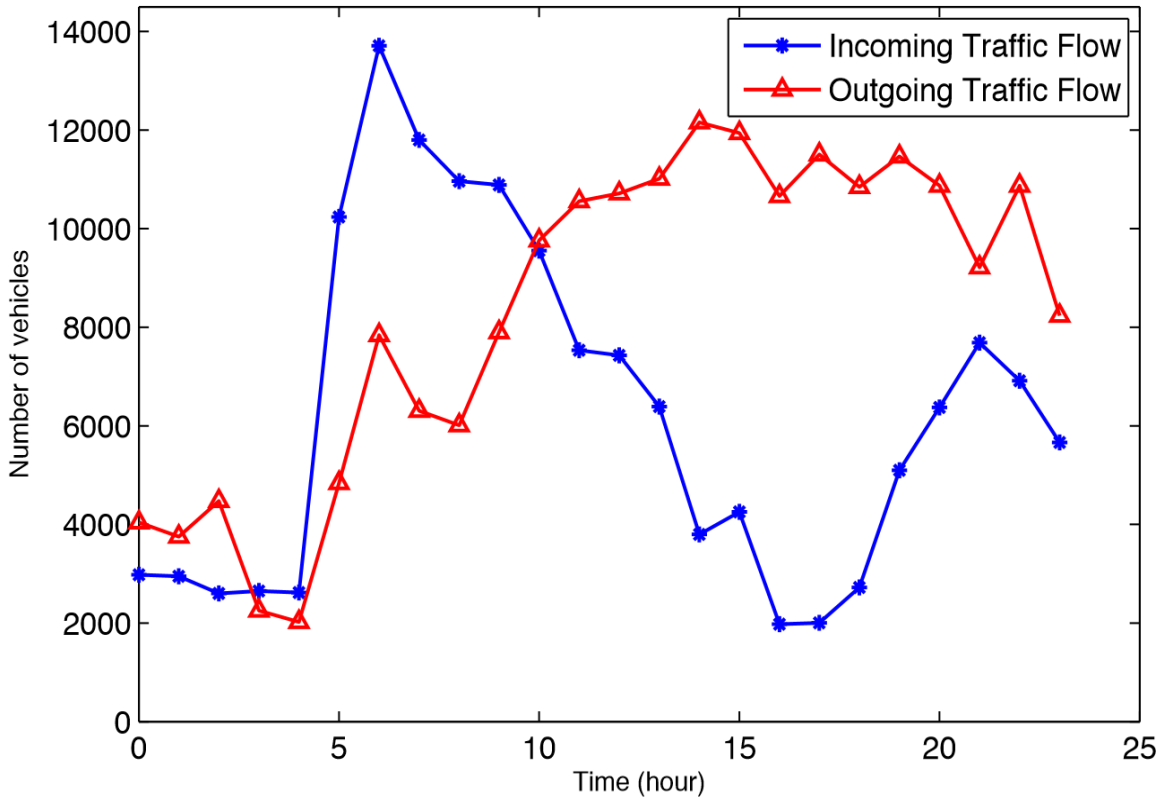
Buffed PIS (BPIS): Some designer has used buffered to find solution of PIS phenomenon. In a parking system, leaved some free spaces as buffer for peak hours solution. In such a way, a parking area had buffered space then it shows that the parking lot is fully occupied. If the buffer was very small, then unable to removed. If it was very big, then used parking spaces as low.

Based on our observation on real google Map data analysis, we found that incoming traffic during peak hours (6 ~10 AM) was high on weekday and the same situation with outcoming traffic 5~9 PM. We have tracked data from google MAP open APIs and used it our simulation work. Therefore, currently parking demands was very high. Means our system helped to user as a reserve parking on prior to save time & money.

Table 5.3: Traffic data table on different roads (In 15 days)

Dated	Road Name	6 ~10 AM	10 ~2 PM	2 ~5 PM	5~9 PM	9~6AM
01-May-18	X	High	Mediam	Low	High	Low
02-May-18	X	High	Mediam	Low	High	Low
03-May-18	X	High	Mediam	Mediam	High	Low
04-May-18	X	High	Mediam	Low	High	Low
05-May-18	Y	Mediam	Low	Low	Low	Low
06-May-18	Z	Low	Low	Low	Low	Low
07-May-18	Y	High	Mediam	Low	High	Low
08-May-18	Y	High	High	Low	High	Low
09-May-18	Y	High	Mediam	Low	High	Low
10-May-18	X	High	Mediam	Low	High	Low
11-May-18	X	High	Mediam	Low	High	Low
12-May-18	X	Mediam	Low	Low	Mediam	Low
13-May-18	X	Low	Low	Low	Low	Low
14-May-18	Y	High	Mediam	Low	High	Low
15-May-18	Y	High	Mediam	Low	High	Low
Average		**High**	Mediam	Low	**High**	Low

Traffic on 01/05/18



Traffic on 15/05/18

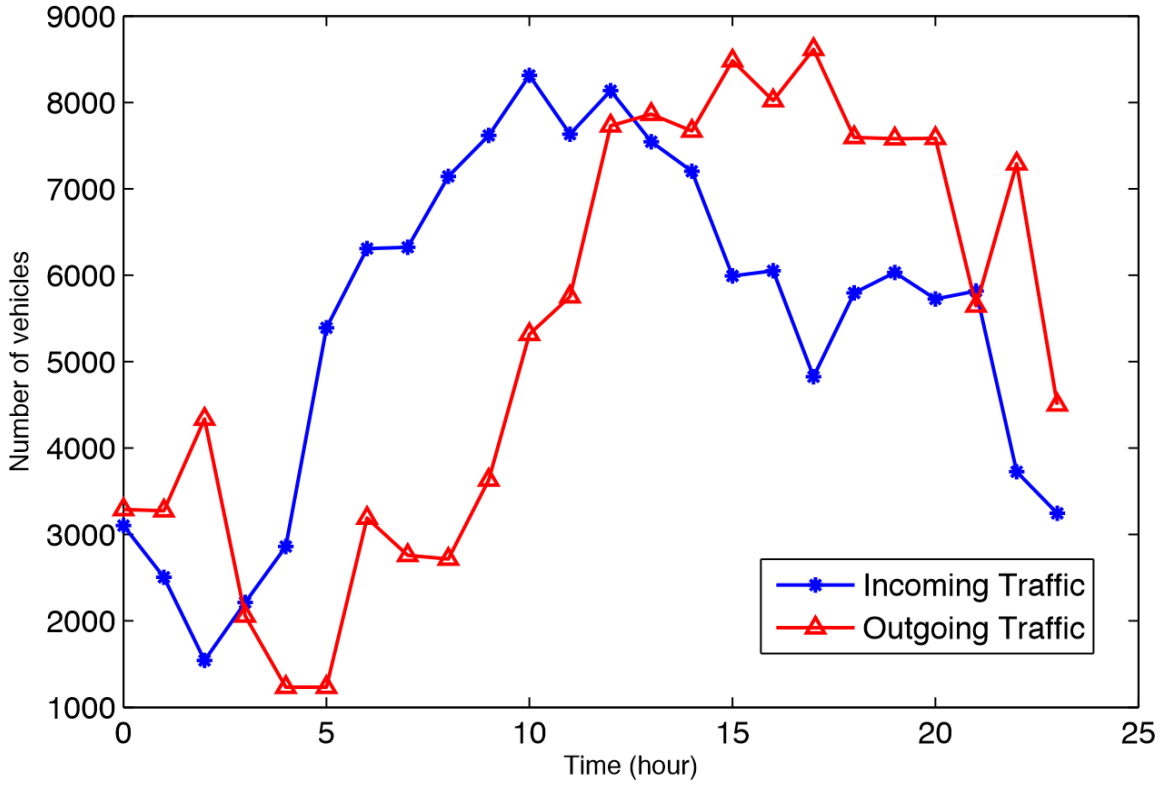


Figure 5.3: Incoming and Outgoing Traffic flow in a city

2. For inside parking system:

We simulated a program in C++. We provided LNP number as input (Vehicles number after detection). After detection of number plate, this program checked in DB that if an already reserve slot X, then it provided direction to reach X. If in case of new entry, provided new available slot Y & direction.

In case of exits, it navigated via shortest path to exits gate and again detect LNP & generated bill according to time.

Table 5.4: Entry and Exit gate database

Entry DB					
LNP Number	Is reservered	Slot number	current time	Booking entry time	Entry time (G)
Exits DB					
LNP Number	Is reserverd	Current exits time	Booking exits time	Exist Time (S)	Billing time = G-S

5.3. RESULT

5.2.1 Result for License number plate: Here we checked till before gradient convergence (As per our algorithm changes). Below are some samples of images.

Image 1: Checked for taken image in day time

Image 2: Checked for taken image in night time

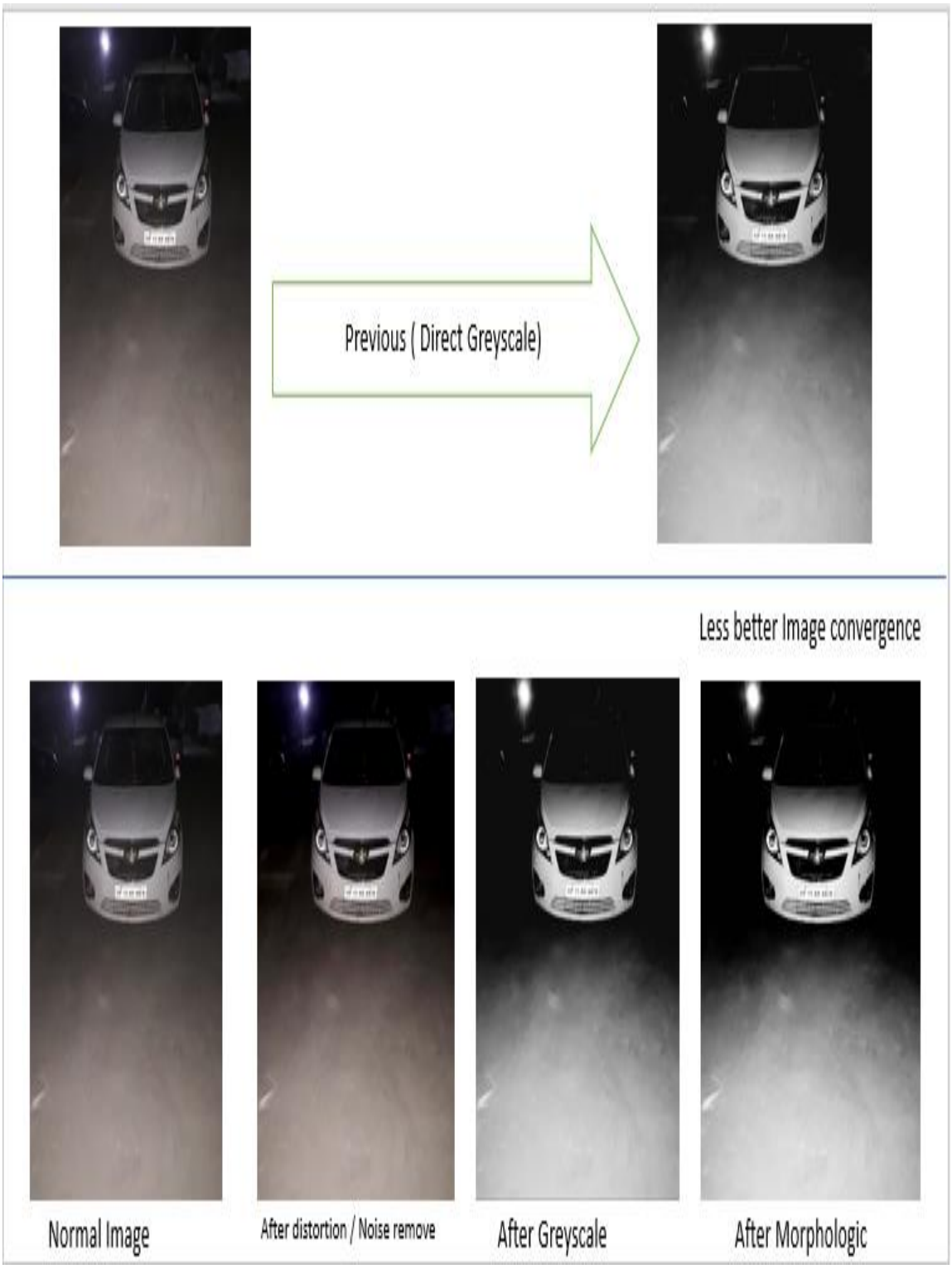


Image 3: Checked for taken image from heights

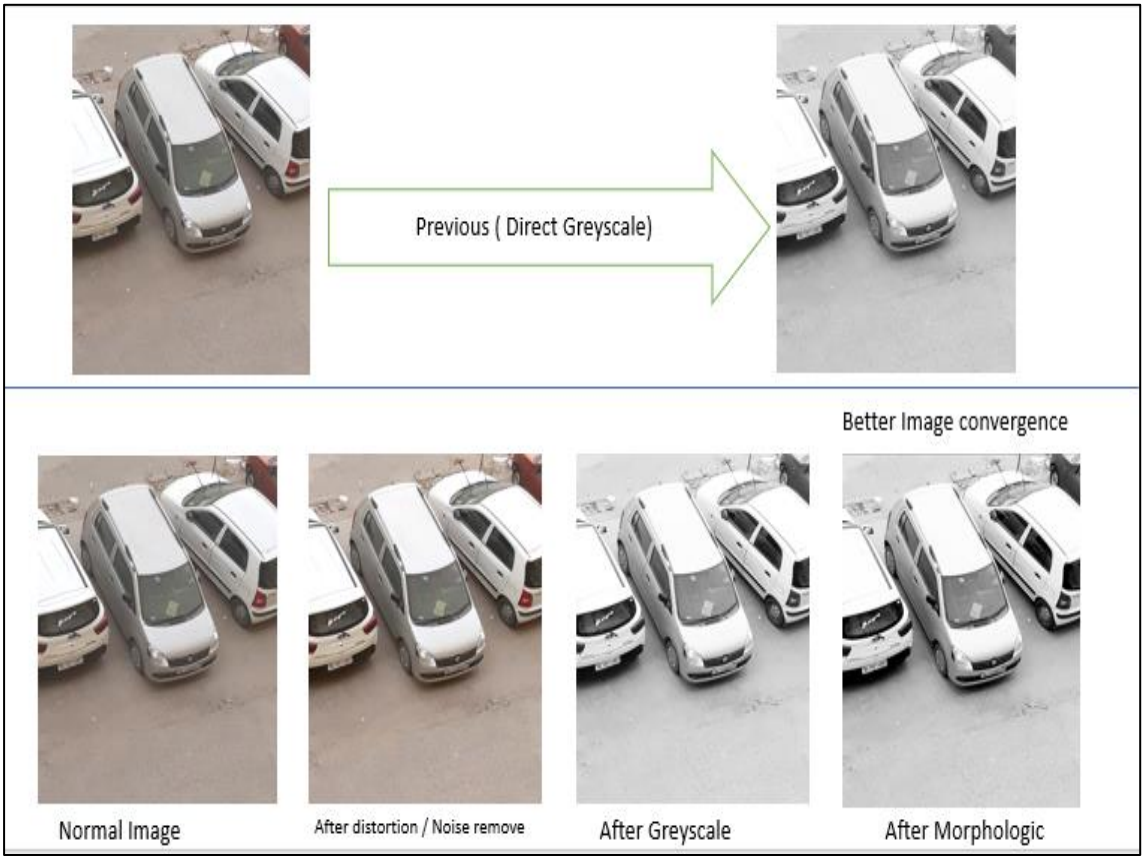


Figure 5.4: Applying proposed algorithm in different images

Result analysis:

Table 5.5: Result compression between proposed and exist detection techniques

Comparision		Image Quality		Image Size	
Image Name	**Type**	**Proposed**	**Already exits**	**Proposed**	**Already exits**
Image1	Daylight	Good	Poor	656 KB	554 KB
Image2	Night Light	Average	Poor	4.46 MB	740 KB
Image3	Height	Good	Average	1711 KB	3320 KB

As per result analysis, our proposed enhancement in algorithm given better Image quality in less size. It provided best result after implementation this change in existence algorithm. Also, this enhancement was already proved in other research paper [4][5][6][7] with separate operation. We have merged all and implement in paper [3] and found that result was good.

5.2.2 Result set for traffic management for intersection

Displaying a traffic intersection simulation in MATLAB environment. As below:

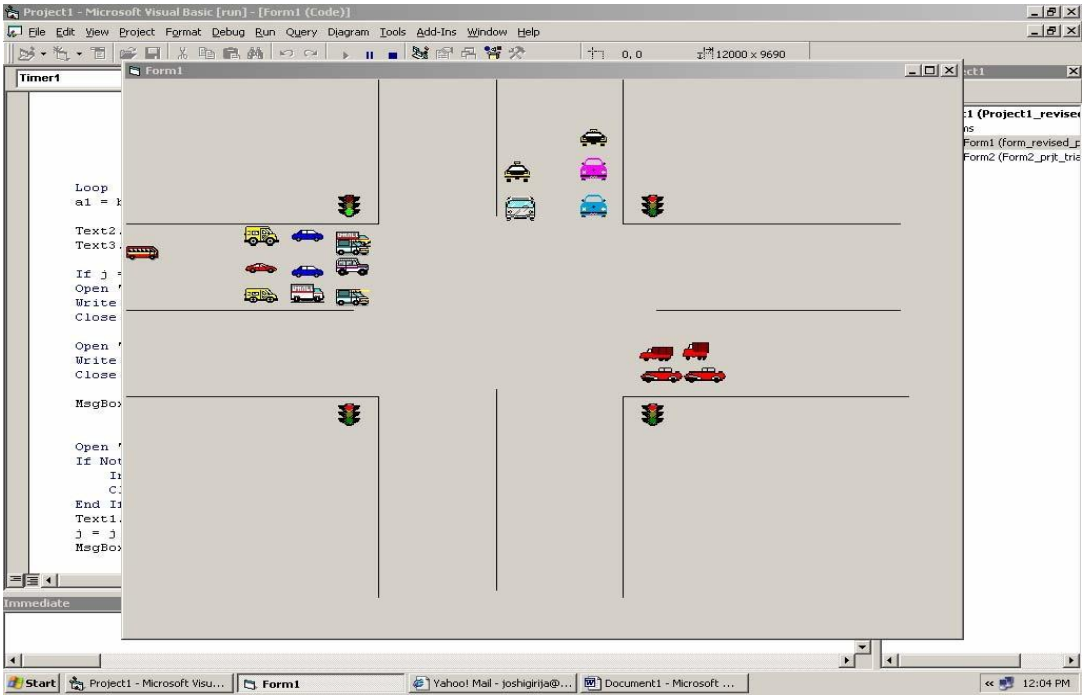


Figure 5.5: Simulation of traffic intersection

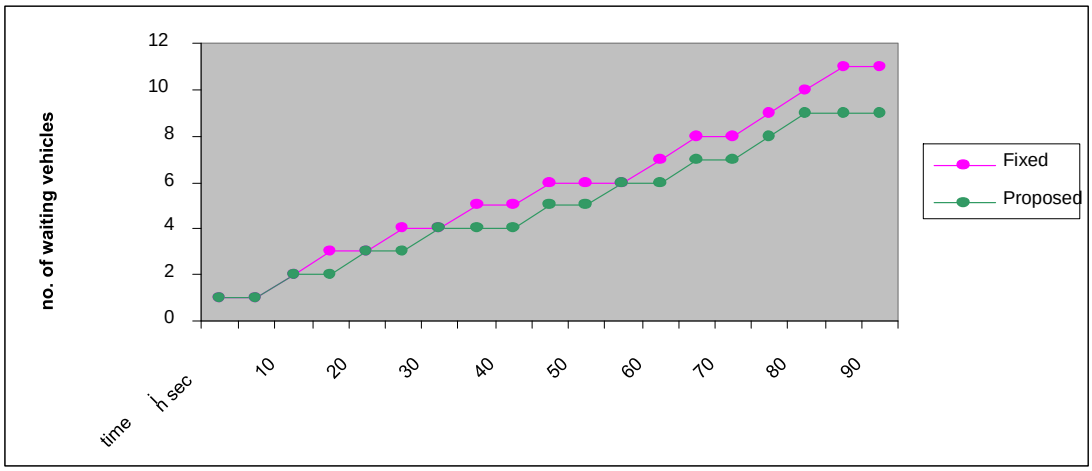


Figure 5.6: Graph showing compression between Fixed and Proposed system

As per graph, as per result the proposed system was more effective than normal fixed system. It provided help to reducing the number of vehicles waiting time. Hence it was a time saving algorithm than fixed normal system.

5.2.3 Result for proposed parking system

1. For parking system in city:

Traffic Searching for parking with used of proposed technique:

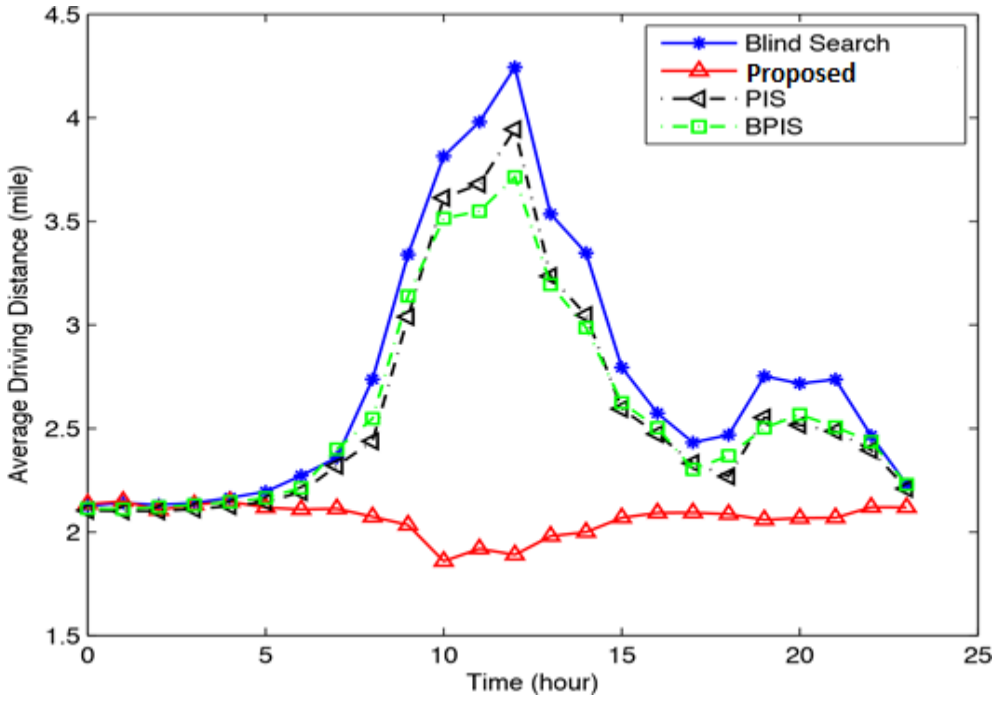


Figure 5.7: Comparison among different traffic searching strategies

Table5.6: Result comparison with Proposed system and other systems

Average Distance (In Mile)	6AM to 10 AM	10 AM to 5 PM	5 PM to 9 PM
Blind Search	Worst	Average	Worst
PIS	Worst	Average	Average
BPIS	Average	Average	Average
Proposed	Best	Best	Best

Here, result was best for proposed system during the peak hours also. This approach is more better than PIS and BPIS also.

2. For inside parking system:

Input: We simulated it in C++ after adding shortest path & navigation. Inputs were below:
Number of TC: 1
Dimension of parking space (K*L): 8 10
Parking area: R- Road & P-Parking, Entry (1, 1) & Exit (K, 1) & (K, L), Entry denoted by -1, Exit denote by -2

R P P P P P P P R
R R R R R R R R R
R P P P P P P P R
R P P P P P P P R
R R R R R R R R R
R P P P P P P P R
R R R R R R R R R
R P P P P P P P R

Number of trails: 5
1 45878 (LNP Number)
1 23564
1 4781
2 23564
1 7894

Output:
After simulation of this system inside parking, we have found Navigation of car inside parking and time difference (in second) for bill generation.

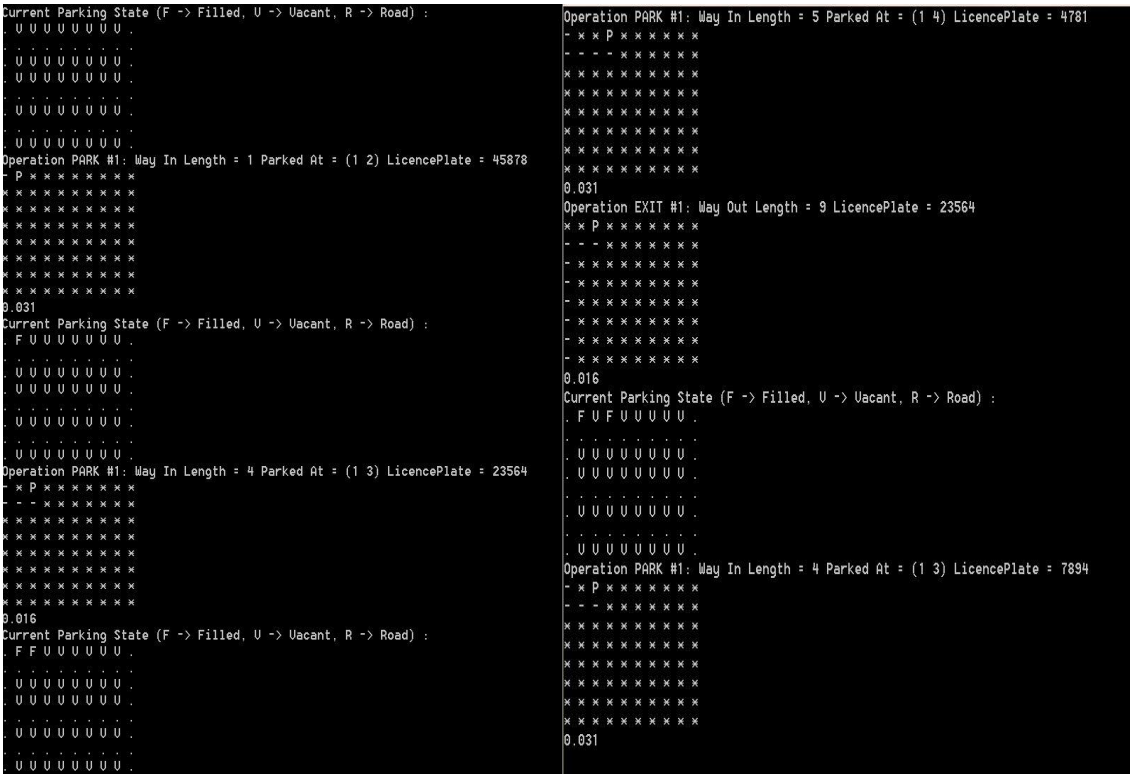


Figure 5.8: Simulation result of inside parking system

Source Code for Parking System (Inside Parking system)

// Begin and Ending date and timing are same because time difference is very low. In actual system this would not be the case.

```
#define MAXH 1005

// 1. 1 Entry (1, 1)
// 2. 2 Exits (K, 1) (K, L)

char parkingLot[MAXH][ MAXH];
bool visited[MAXH][ MAXH];
int pathTraverse[MAXH][ MAXH][2];
char displayPath[MAXH][ MAXH];
char currentParkingState[MAXH][ MAXH];

int K, L;
int ds[] = {0, 1, 0, -1};
```

```
int dt[] = {1, 0, -1, 0};

bool isSafe(int s, int t)
{
	if (s >= 1 && s <= K && t >= 1 && L <= L && parkingLot[x][y] == 'R' && !visited[s][t])
		return true;
	return false;
}

// Distance is same therefore applying Breadth-First-Search otherwise have used dijkstra
int bfs(int s, int t, int exitCnt, int exit_S[], int exit_T[])
{
	queue<pair<int, pair<int, int> > > pq;
	pq.push(make_pair(0, make_pair(s, t)));
	pathTraverse[s][t][0] = s;
	pathTraverse[s][t][1] = t;
	visited[s][t] = true;
	int cnt, exit_s, exit_t;
	bool flag = false;
	while (!pq.empty())
	{
		pair<int, pair<int, int> > presentAt = pq.front();
		pq.pop();
		int p = presentAt.second.first;
		int q = presentAt.second.second;
		flag = false;
		for (int i = 0; i < exitCnt; i++)
		{
			if ((p == exit_S[i] && q == exit_T[i]))
			{
				//cout << "P = " << p << " exit_S[i] = " << exit_T[i] << " q = " << q << " exit_S[i] = " << exit_T[i] << endl;
```

```
                        cnt = presentAt.first;
                        exit_s = p;
                        exit_t = q;
                        flag = true;
                        break;
                }
        }
        if (flag)
                break;
        for (int e = 0; e< 4; e++)
        {
                int _s = p + ds[e];
                int _t = q + dt[e];

                if (isSafe(_s, _t))
                {
                        visited[_s][_t] = true;
                        pathTraverse[_s][_t][0] = p;
                        pathTraverse[_s][_t][1] = q;
                        pq.push(make_pair(presentAt.first + 1, make_pair(_s,
_t)));
                }
        }
    }
    // Empty the queue
    while (!pq.empty())
        pq.pop();

    // Print the path
    while (!(exit_s == pathTraverse[exit_s][exit_t][0] && exit_t ==
pathTraverse[exit_s][exit_t][1]))
    {
        displayPath[exit_s][exit_t] = '-';
```

```
            int tmp_s = exit_s;
            exit_s = pathTraverse[exit_s][exit_t][0];
            exit_t = pathTraverse[tmp_s][exit_t][1];
        }

        return cnt;
}

enum CMD
{
        PARK = 1,
        EXIT
};
#define MAXH 100005

int hshCarToGrid[MAXH][2];

int main()
{
        int TC;
        freopen("input.txt", "r", stdin);
        freopen("output.txt", "w", stdout);
        cin >> TC;
        for (int t = 1; t <= TC; t++)
        {
            cin >> K >> L;
            for (int e = 1; e <= N; e++)
            {
                for (int f= 1; f<= M; f++)
                {
                    cin >> parkingLot[e][f];
                    if (parkingLot[e][f] == 'P')
                        currentParkingState[e][f] = 'V';
```

```
                else
                        currentParkingState[e][f] = '.';
                visited[e][f] = false;
                pathTraverse[e][f][0] = -1;
                pathTraverse[e][f][1] = -1;
                displayPath[e][f] = '*';
        }
}
int operation;
cin >> operation;
for (int za = 1; za<= operation; za++)
{
        int car_s, car_t;
        int wayOutLen, wayInLen;
        int cmd;
        cin >> cmd;
        int exit_X[2], exit_Y[2], exitCnt, licencePlate;
        // licencePlate No should be less than 10^5
        bool flag;
        clock_t begin_time, end_time;
        time_t b_time, e_time;
        char *buffer;
        switch (cmd)
        {
        case EXIT:
                begin_time = clock();
                b_time = time(0);
                buffer = ctime(&b_time);
                //cout << "Begin : " << buffer << endl;
                cin >> licencePlate;
                car_x = hshCarToGrid[licencePlate][0];
                car_y = hshCarToGrid[licencePlate][1];
                //cin >> car_x >> car_y;
```

```
                exit_X[0] = N; exit_Y[0] = 1;
                exit_X[1] = N; exit_Y[1] = M;
                exitCnt = 2;
                wayOutLen = bfs(car_x, car_y, exitCnt, exit_X, exit_Y);
                cout << "Operation EXIT #" << t << ": Way Out Length =
" << wayOutLen << " LicencePlate = " << licencePlate <<endl;
                displayPath[car_x][car_y] = 'P';
                currentParkingState[car_x][car_y] = 'V';
                // Trace the output path
                for (int e= 1; e<= N; e++)
                {
                        for (int f = 1; f <= M; f++)
                        {
                                cout << displayPath[e][f] << " ";
                                visited[e][f] = false;
                                displayPath[e][f] = '*';
                        }
                        cout << endl;
                }
                end_time = clock();
                e_time = time(0);
                buffer = ctime(&e_time);
                //cout << "End : " << buffer << endl;
                //cout      <<      float(clock()      -      begin_time)      /
CLOCKS_PER_SEC << endl;;
                break;
        case PARK:
                begin_time = clock();
                b_time = time(0);
                buffer = ctime(&b_time);
                cin >> licencePlate;
                //cout << "Begin : " << buffer << endl;
```

```
                cout << "Current Parking State (F -> Filled, V -> Vacant, R -> Road) :" << endl;
                flag = true;
                for (int e = 1; e <= N; e++)
                {
                        for (int f = 1; f <= M; f++)
                        {
                                cout << currentParkingState[i][j] << " ";
                                if (flag && currentParkingState[i][j] == 'V')
                                {
                                        car_x = e;
                                        car_y = f;
                                        flag = false;
                                }
                        }
                        cout << endl;
                }
                hshCarToGrid[licencePlate][0] = car_x;
                hshCarToGrid[licencePlate][1] = car_y;
                //cin >> car_x >> car_y;
                exit_X[0] = 1; exit_Y[0] = 1;
                exitCnt = 1;
                wayInLen = bfs(car_x, car_y, exitCnt, exit_X, exit_Y);
                cout << "Operation PARK #" << t << ": Way In Length = " << wayInLen << " Parked At = (" << car_x << " " << car_y << ")" << " LicencePlate = " << licencePlate<<endl;
                currentParkingState[car_x][car_y] = 'F';
                displayPath[car_x][car_y] = 'P';
                // Trace the output path
                for (int e = 1; e <= N; e++)
                {
                        for (int f = 1; f <= M; f++)
                        {
```

```
                                        cout << displayPath[e][f] << " ";
                                        visited[e][f] = false;
                                        displayPath[e][f] = '*';
                                }
                                cout << endl;
                        }

                        end_time = clock();
                        e_time = time(0);
                        buffer = ctime(&e_time);
                        //cout << "End : " << buffer << endl;
                        //cout << "End : " << end_time << endl;
                        //cout << float(clock() - begin_time) /
CLOCKS_PER_SEC << endl;;
                        break;
                default:
                        break;
                }
        }
        // When we have Licence plate we can make a DB entry using it as primary
key.
        //cout << "Case #" << t << " Database for this Scenrio = "<< endl;
    }
    return 0;
}
```

CHAPTER 6: CONCLUSION

The idea of Smart Cities has always helpful for human being. From last few years, many research, development have been growing for smart cities. The enhancement of IoT, Cloud technologies and AI have provided new area in development of smart cities.

As we know that smart parking system and traffic system have been the main key to building smart cities. Here, we have discussed the current problem of parking and provided a solution as IoT based smart parking system with integration of cloud system. Our proposed system provides a unique information about current availability of parking area in real time. It abled to helpful for booking a parking area from remote locations using mobile application or website and select the root in which less traffic. Proposed advanced License plate detection algorithm provided more accurate data of detection in any environment, at any time and any weather.

In the field of traffic management, our proposed technique (AI – Fuzzy and IoT – sensors system), provided more real traffic data and help to save time in intersection in a smart way.

Future Scope

Our proposed system abled to collect information and sync different parking area identification as per driver need. We can enhance our system in future by adding below fields:

Security of communication: By using secure symmetric encryption algorithm, we may add encrypted data for all communication between service providers and users.

Users privacy: During user's authentication in system, we may use virtual identifier. To hide the real identification of any user this will support.

Smart License number plate: We may have enhanced license number plate by adding sensors & technology that will help in over speeding, accidents detection and share information to emergency contacts with help to IoT.

References

[1] A. Khanna and R. Anand, "IoT based Smart Parking System," in *International Conference on Internet of Things and application (IOTA)*, Pune, 2016.

[2] A. Roy and P. Poddar, "Smart Traffic & Parking Management using IoT," in *Information Technology, Electronics and Mobile Communication Conference (IEMCON),IEEE 7th Annual*, Vancouver, 2016.

[3] L. Hu and Q. Ni, "IoT- Driven Automated Object Detection Algorithm for Urban Surveillance System in Smart Cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 747-754, April 2018.

[4] G. Abinaya, R. Banumathi, V. Sesharsri and A. N. Kumar, "Rectify the Blurred License Plate Image From Fast Moving Vehicles Using Morphological Process," in *IEEE International Conference on Electrical, Instrumentation and Communication Engineering (ICEICE)*, Karur, India, April 2017.

[5] N. Farijan and M. Rahimi, "Algorithms for licenseplate detection : A survey," in *International Congress on Technology, Communication and Knowledge*, Mashhad, Iran, Nov 2014.

[6] V. Sikri, "The Smart Number Plate," in *International Conference on Communication and Signal Processing (ICCSP)*, Melmaruvathur, India, April, 2016.

[7] H. N. Do, M. T. Vo, B. Q. Vuong, H. T. Pham, A. H. Nguyen and H. Q. Luong, "Automatic License Plate recognition Using Mobile Device," in *International Conference on Advanced Technologies for Communications (ATC)*, Hanoi, Vietnam, 2016.

[8] The global warning survival guide, Time magazine, 2007.

[9] E. M. Atkins, "Cyber-physical aerospace: Challenges and future directions in transportation and exploration systems," in Technical Report CMU/SEI-2011TR-002, 2011.

[10] E. M. Clark, B. Krogh, A. Platzer and R. Rajkumar, "Analysis and verification challenges for cyberphysical transportation systems," in *Carnegie Mellon University*, Pittsburgh, PA, 2008.

[11] F. P. Kelly, A. K. Maulloo and K. H. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237-252, March 1998.

[12] A. L. Kornhauser, "About prospect eleven and the DARPA project," in *J. Robotic System*, 2006.

[13] T. Litman, "Smart transportation emission reduction strategies," in *Victoria Transport Policy Institute*, 2011.

[14] M. Caliskan, D. Graupner and M. Mauve, "Decentralized discovery of free parking places," in *Proceedings of the 3rd international workshop on Vehicular ad hoc networks (VANET'06)*, Los Angeles, CA, USA, September, 2006.

[15] M. Feldman, K. Lai and L. Zhang, "A price-anticipating resource allocation mechanism for distributed shared clusters," in *Proceedings of the 6th ACM conference on Electronic commerce*, Vancouver, BC, Canada, June, 2005.

[16] J. H. Kim, S. K. Kim, S. H. Lee, T. M. Lee and J. Lim, "License Plate Detection and Recognition Algorithm for Vehicle Black Box," in *International Automatic Control Conference (CACS)*, Pingtung, Taiwan, Nov 2017.

[17] I. Lee and O. Sokolsk, "Medical cyber physical systems," in *Design Automation Conference (DAC)*, Anaheim, CA, USA, June 2010.

[18] R. Lu, X. Lin, H. Zhu and X. Shen, "SPARK: A new VANET-based smart parking scheme for large parking lots," in *Proceedings of IEEE INFOCOM'09*, Rio de Janeiro, Brazil, April 2009.

[19] B. McMillin, C. Gill, M. L. Crow, F. Liu, D. Niehaus, A. Potthast and D. Tauritz, "Cyber-physical systems distributed control: the advanced electric power grid," in *Electrical Energy Storage Applications And Technologies*, 2007.

[20] R. Rajkumar, I. Lee, L. Sha and J. Stankovic, "Cyberphysical systems: The next computing revolution," in *Proceedings of the 47th Design Automation Conference*, Anaheim, California, June, 2010.

[21] A. Arizmendi, J. Reisdorff and J. Armstrong, "Parking Lot Detection Proposal," 2007.

[22] K. Sampigethaya, R. Poovendran and L. Bushnell, "Secure operation, control and maintenance of future e-enabled airplanes," *Proceedings of the IEEE special issue on Aviation Information Systems*, vol. 96, no. 12, pp. 1992 - 2007, 2008.

[23] A. Susan, Shaheen and C. J. Rodier, "Smart parking management field test: A bay area rapid transit (BART) district parking demonstration," in *Institute of Transportation Studies, University of California*, Davis, California, 2005.

[24] X. Wang and H. Schulzrinne, "Pricing network resources for adaptive application," in *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, Anchorage, USA, 2001.

[25] W. Wolf, "The good news and the bad new," *IEEE Computer,10.1109/MC.2007.404*, vol. 40, no. 11, pp. 104-105, 2007.

[26] D. Work, A. Bayen and Q. Jacobso, "Automotive cyber physical systems in the context of human mobility," in *National Workshop on High-Confidence Automotive Cyber-Physical Systems*, Troy, MI, April, 2018.

[27] D. B. Work and A. M. Bayen, "Impacts of the mobile internet on transportation cyberphysical systems: Traffic monitoring using smartphones," in *National*

Workshop for Research on High-Confidence Transportation Cyber-Physical Systems: Automotive, Aviation and Rail, Washington, 2008.

[28] B. Li, B. Tian, Q. Yao and K. Wang, "A vehicle license plate recognition system based on analysis of maximally stable extremal regions," *9th IEEE International Conference on Networking, Sensing and Control (ICNSC),* pp. 399 - 404, 2012.

[29] M. Yu and Y. Kim, "An approach to Korean license plate recognition based on vertical edge matching," *IEEE International Conference on Systems, Man, and Cybernetics,* vol. 4, p. 2975 – 2980, 2000.

[30] A. A. Ghaili, S. Mashohor, A. Ramli and A. Ismail, "Vertical-edge-based car-license-plate detection method," *IEEE Transactions on Vehicular Technology,* vol. 62, no. 1, pp. 26 - 38, A. Al-Ghaili, S. Mashohor, A. Ramli, A. Ismail, 2013.

[31] G. Li, R. Yuan, Z. Yang and X. Huang, "A yellow license plate location method based on RGB model of color image and texture of plate," *Second Workshop on Digital Media and its Application in Museum & Heritages,* pp. 42 - 46, 2007.

[32] G. Sun, C. Zhang, W. Zou and G. Yu, "A new recognition method of vehicle license plate based on genetic neural network," *The 5th IEEE Conference on Industrial Electronics and Applications*, pp. 1662 - 1666, 2010.

[33] J. Hsieh, S. Yu and Y. Chen, "Morphology-based license plate detection from complex scenes," *16th IEEE International Conference on Pattern Recognition,* p. 176 –179, 2002.

[34] R. Kerwin, A. Bandala, A. Sybingco, K. Lim and Elmer, "Vehicle Detection and Tracking using Corner Feature Points and Artificial Neural Networks for a Visionbased Contactless Apprehension System," in *Computing Conference*, London, UK, July 2017.

[35] W. J. Zhong and X. Xiaoqing, "A Real-time Detection of Vehicle's Speed Based on Vision Principle and Differential Detection," *IEEE Conference Publications,* pp. 493 - 496, 2009.

[36] S. Sivaraman and M. M. Trivedi, "A review of recent developments in vision-based vehicle detection," *Intelligent Vehicles Symposium (IV), IEEE,* pp. 310 - 315, 2013.

[37] C. P. Uy, R. A. Bedruz, A. R. Quiros, A. Bandala and E. P. Dadios, "Machine vision for traffic violation detection system through genetic algorithm," *International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM),* pp. 1-7, 2015.

[38] C. M. Mwangi, S. M. Kangethe and G. N. Nyakoe, "DESIGN AND SIMULATION OF A FUZZY LOGIC TRAFFIC SIGNAL CONTROLLER FOR A SIGNALIZED INTERSECTION," in *Jomo Kenyatta University of Agriculture and Technology*, Nairobi, Kenya, 2017.