

AUTOMATED TRAFFIC SIGN-BASED VEHICLE CONTROL WITH DEEP LEARNING

Submitted in partial fulfillment of the requirements for the award of a Bachelor of
Engineering Degree in Electronics and Communication Engineering

by

MANISHA S (Reg.No:39130537)

DIVYA VINOJ KUMAR (Reg.No:39130120)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600119

APRIL 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Final Year Project Report is the Bonafide work of **MANISHA S (Reg.No:39130537)** and **DIVYA VINOJ KUMAR (Reg.No:39130120)**, Who carried out the project entitled “**AUTOMATED TRAFFIC SIGN BASED VEHICLE CONTROL WITH DEEP LEARNING**” under my supervision from January 2023 to April 2023.

Internal Guide & Head of the Department

Dr. T. RAVI, M.E., Ph.D.

Submitted for Viva voce examination held on 25-04-2023

Internal Examiner

External Examiner

DECLARATION

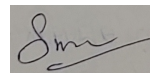
We, **MANISHA S (39130537)** and **DIVYA VINOJ KUMAR (39130120)** hereby declare that the Project Report entitled **AUTOMATED TRAFFIC SIGN BASED VEHICLE CONTROL WITH DEEP LEARNING** by us under the guidance of **Dr. T. RAVI, M.E., Ph.D.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Electronics and Communication Engineering**.

DATE: 25/04/23

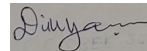
PLACE: Chennai

SIGNATURE OF THE CANDIDATES

1) MANISHA S



2) DIVYA VINOJ KUMAR



ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N.M. NANDHITHA, M.E., Ph.D.** Dean, School of Electrical and Electronics and **Dr. T. RAVI, M.E., Ph.D.** Head of the Department, Department of Electronics and Communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. T. RAVI, M.E., Ph.D.** Head of the Department, Department of Electronics and Communication Engineering for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of project work.

We wish to express our thanks to all teaching and non-teaching staff members of the **Department of Electronics and Communication Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The aim of this project is to develop and implement an algorithm that automatically detects and recognizes INDIAN traffic signs, from the grayscale images captured by a camera mounted on the interior mirror of a vehicle, as a part of designing smarter vehicles. The system operates in real-time within the computational limits of contemporary embedded general-purpose processors. This system will assist the driver by providing the necessary information, regarding the assigned traffic rules, right in front of him and provide additional safety measures by monitoring the vehicle. This method consists of two phases in it: a detection phase, in which all the possible traffic signs in the input image are detected first, and a recognition phase, in which the detected regions are recognized and the information regarding the traffic signs is extracted from them. The detection phase utilizes the region characteristics, such as aspect ratio and size, to hypothesize the traffic sign locations in the input image. We have utilized the idea of the connected component labeling technique and adapted it for the grayscale images, to divide the input image into a set of regions. The recognition phase calculates the invariant features of the inner parts of the detected regions using Hu's moments. It verifies the hypothesis first, before extracting the traffic sign from the detected region using a feed-forward neural network. This method will be experimented on several traffic images and should show that the region characteristics are more immune to different noisy conditions such as partial occlusions.

TABLE OF CONTENT

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF FIGURES	ix
	LIST OF TABLES	xi
1	INTRODUCTION	1
	1.1 DEEP LEARNING	2
	1.1.1 Automatic Traffic Sign Detection System	5
	1.1.2 The Architecture of the System	6
	1.2 ORGANIZATION OF THE REPORT	7
2	LITERATURE SURVEY	8
	2.1 SYSTEMATIC REVIEW	8
	2.2 SUMMARY	14
	2.3 LIMITATIONS IN EXISTING WORK	15
	2.4 PROBLEM DEFINITION	15
	2.5 OBJECTIVES OF THE PROJECT	15

3	EXISTING METHODOLOGY	16
	3.1 EXISTING SYSTEM	16
	3.2 IMAGE PIXEL	19
	3.3 GRAYSCALE IMAGES	20
	3.3.1 Applications	23
	3.4 WORKING PROCESS	23
	3.5 PROPOSED SYSTEM	24
4	PROPOSED METHODOLOGY	25
	4.1 BLOCK DIAGRAM OF PROPOSED SYSTEM	25
	4.2 HARDWARE DESCRIPTION	26
	4.2.1 Arduino Uno	27
	4.2.2 L298n Motor Driver	27
	4.2.3 DC motor	28
	4.2.4 Power Supply	29
	4.3 WORKING METHOD CIRCUIT DIAGRAM	30
	4.4 SOFTWARE ALGORITHM FLOWCHART	31
	4.5 OBJECT RECOGNITION	34
	4.6 STEPS INVOLVED IN PROPOSED ALGORITHM	35
	4.7 PREPROCESSING	37
	4.8 FEATURE EXTRACTION FROM HU's MOMENTS	37
	4.9 REGION CLASSIFICATION AND SIGN EXTRACTION	41
	4.10 CLASSIFICATION BY MATCHING FEATURES	41
	4.11 ROAD SIGN EXTRACTION USING NEURAL NETWORK	42

5	RESULTS AND DISCUSSION	47
	5.1 DETECTION PHASE	48
	5.1.1 Condition 1: Region size	49
	5.1.2 Condition 2: Aspect ratio	50
	5.2 RECOGNITION PHASE	51
	5.2.1 Pre-Processing	51
	5.2.2 Feature extraction	51
	5.2.3 Classification	52
	5.2.4 Traffic Sign Information Extraction	53
	5.2.5 Experiment 1	54
	5.2.6 Experiment 2	56
	5.2.7 Experiment 3	59
	5.2.8 Experiment 4	62
6	CONCLUSION AND FUTURE SCOPE	66
	6.1 SYNOPSIS	66
	REFERENCES	68

LIST OF FIGURES

FIGURE.NO	DESCRIPTION	PAGE NO
1.1	Block diagram of automatic traffic sign detection system	6
3.1	Road sign and corresponding actions	22
3.2	Sample traffic scene	23
4.1	Block Diagram of automatic traffic sign-based vehicle control system	25
4.2	Figure of Arduino	27
4.3	Figure of L298N motor driver	28
4.4	Figure of DC motor	28
4.5	Figure of power supply	29
4.6	Circuit diagram of automatic traffic sign-based vehicle control system	30
4.7	Flow chart of automatic traffic sign-based vehicle control system	31
4.8	Flow chart of image processing in CNN	32
4.9	Implementation diagram of the automatic traffic sign-based vehicle control system	33
4.10	Architecture of the multilayer neural network	44
5.1	Original image	48
5.2	Grayscale image labeling	48
5.3	Histogram of the labeled image	49
5.4	Generated hypothesis	50
5.5	Preprocessing	51
5.6	Analyzing the upper half of the detected region	52

FIGURE.NO	DESCRIPTION	PAGE NO
5.7	Analyzing the lower half of the region	53
5.8	Recognized speed limit sign	53
5.9	Untouched image	54
5.10	Labeled image	54
5.11	Diagram of the labeled image	55
5.12	Generated deduction	55
5.13	Determined traffic sign	56
5.14	Initial image	56
5.15	Classified image	57
5.16	Histogram of the termed image	57
5.17	Generated interpretation	58
5.18	Perceived traffic sign	59
5.19	Primary image	59
5.20	Specified image	60
5.21	Illustration of labeled image	60
5.22	Generated inference	61
5.23	Developed Presumption	62
5.24	Aboriginal image	62
5.25	Designated image	62
5.26	Histogram of the entitled image	63
5.27	Generated system	63
5.28	Recognized sign board	64
5.29	Automatic traffic sign-based vehicle control system	65

LIST OF TABLES

TABLE NO	DESCRIPTION	PAGE NO
5.1	Examples of few regions' characteristics	50
5.2	Classifying the region by recognizing its components	52
5.3	The region's parameters from the labeled image	55
5.4	Classifying the hypothesized regions	56
5.5	Region criterion from the labeled image	58
5.6	Analyzing the interpreted region	58
5.7	Region specification from the image 5.21	61
5.8	Segregating the hypothesized region in the figure	61
5.9	Region framework from the labeled image	63
5.10	Distinguishing the inferred regions in the figure	64

CHAPTER 1

INTRODUCTION

Our universe is filled with different kinds of rules or laws such as physical laws, environmental laws etc. Similarly, a set of traffic rules has been created, to guide the people travelling and to regulate the traffic flow. Some of these traffic rules are indicated using the sign boards positioned by the sides of the roads. There are different kinds of sign boards used in America, such as yield signs, stop signs speed-limit signs, warning signs and guide signs. These sign boards are designed in such a way that by looking at them the driver could extract the information from it. The driver requires all kinds of information while travelling. A driver does multiple tasks while he is driving such as, monitoring the road path ahead, observing the vehicles around him, constantly checking the blind spots, indicating the lane changes and keeping in track with the traffic rules. Thus, the driver is overburdened while driving. Nowadays the new cars are being equipped with different kinds of new technologies which help the driver in his driving duties. The topic that comprises all such kinds of technologies is known as driver assistance system. As the name suggests it assists the driver by providing the required information for him, warning about the traffic violations and avoiding the accidents.

It includes subtopics like sign board detection, monitoring a lane change, vehicle detection, auto parking and speed detection. A sign board detection system detects the sign boards that are present along the sides of the roads, recognizes them and displays them on the vehicle's dashboard. A lane change monitoring system assists the driver by warning him if the blind spots are not clear while he is changing the lanes. A vehicle detection system automatically detects the vehicles in the road path ahead. An auto parking system parks the vehicle for the driver.

A traffic sign detection system detects the traffic sign board located by the side of the road, recognizes it and displays it on the dashboard in front of the driver. Thus, these tools can help the driver by reducing his efforts while driving and providing additional safety measures.

- Respond quickly to changing traffic needs.
- Experience higher tilt possibility compared to mechanical tilt.
- Reduces human error with accurate mechanical control.
- No specialized manpower or equipment needed.
- Ideal for isolated or difficult-to-access sites

1.1 DEEP LEARNING

Deep learning (also known as deep structural learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi supervised or unsupervised. Deep-learning architectures such as deep neural networks, deep belief networks, deep reinforcement learning, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, speech recognition, natural language processing, machine translation, bioinformatics, drug design, medical image analysis, climate science, material inspection and board game programs, where they have produced results comparable to and in some cases surpassing human expert performance. The adjective "deep" in deep learning refers to the use of multiple layers in the network.

Early work showed that a linear perceptron cannot be a universal classifier, but that a network with a non-polynomial activation function with one hidden layer of unbounded width can. Deep learning is a modern variation which is concerned with an unbounded number of layers of bounded size, which permits practical application and optimized implementation, while retaining theoretical conditions.

In deep learning the layers are also permitted to be heterogeneous and to deviate widely from biologically informed connectionist models, for the sake of efficiency, trainability and understandability, whence the "structured" part. Most modern deep learning models are based on artificial neural networks, specifically convolutional neural networks (CNN)s, although they can also include propositional formulas or latent variables organized layer-wise in deep generative models such as the nodes in deep belief networks and deep Boltzmann machines.

The word "deep" in "deep learning" refers to the number of layers through which the data is transformed. More precisely, deep learning systems have a substantial credit assignment path (CAP) depth. The CAP is the chain of transformations from input to output. CAPs describe potentially causal connections between input and output. For a feedforward neural network, the depth of the CAPs is that of the network and is the number of hidden layers plus one (as the output layer is also parameterized).

For recurrent neural networks, in which a signal may propagate through a layer more than once, the CAP depth is potentially unlimited. No universally agreed-upon threshold of depth divides shallow learning from deep learning, but most researchers agree that deep learning involves CAP depth higher than 2.

CAP of depth 2 has been shown to be a universal approximation in the sense that it can emulate any function. Beyond that, more layers do not add to the function approximate ability of the network. Deep models ($CAP > 2$) can extract better features than shallow models and hence, extra layers help in learning the features effectively.

Deep learning architectures can be constructed with a greedy layer-by-layer method. Deep learning helps to disentangle these abstractions and pick out which features improve performance.

For supervised learning tasks, deep learning methods eliminate feature engineering, by translating the data into compact intermediate representations akin to principal components and derive layered structures that remove redundancy in representation.

Deep learning algorithms can be applied to unsupervised learning tasks. This is an important benefit because unlabeled data is more abundant than the labelled data. Examples of deep structures that can be trained in an unsupervised manner are deep belief networks. Deep neural networks are generally interpreted in terms of the universal approximation theorem or probabilistic inference.

The universal approximation theorem for deep neural networks concerns the Capacity of networks with bounded width but the depth is allowed to grow. It has been proved by previous research that if the width of a deep neural network with ReLU activation is strictly Larger than the input dimension, then the network can approximate any Lebesgue integrable function; If the width is smaller or equal to the input dimension, then a deep neural network is not a universal approximator. The probabilistic interpretation derives from the field of machine learning.

Deep Learning Features inference as well as the optimization concepts of training and testing, related to fitting and generalization, respectively. More specifically, the probabilistic interpretation considers the activation of nonlinearity as a cumulative distribution function. The probabilistic interpretation led to the introduction of dropout as regularized in neural networks. The probabilistic interpretation was introduced by researchers including Hopfield, Windrow and Narendra and popularized in surveys such as the one by Bishop.

1.1.1 Automatic Traffic Sign Detection System

The primary causes of the vast majority of traffic accidents that occur in the world today are hasty driving, ignorance, and disregard for posted traffic signs. It has also been exemplified in a report that almost 14 people lose their lives every 60 minutes. Drivers and pedestrians are both put in danger when they disregard traffic signs. Congestion and accidents are decreased with the use of traffic signals and signs indicating the authorized rules and regulations. In the United States traffic signs vary depending on the locations and road conditions. The traffic signs on the freeways and other city roads are different. The work zones have different rules of traffic. So, in order to follow the traffic rules, the driver must look for the signs located by the sides of the roads.

If the vehicle is equipped with an automatic traffic sign detection system it will detect the traffic signs, recognize them, and displays the information right in front of the driver. This system would also be helpful to warn the driver if he is not paying attention to the traffic signs due to weather conditions or during the night time when it's hard to see the traffic signs. In addition to helping the driver, this technology will increase pedestrian and passenger safety on the road. By interconnecting the traffic sign detection system and the cruise control system, the cruising speed could be automatically chosen.

Nowadays new vehicles are being equipped with navigation systems which also provide sign information. But, according to the automaker BMW, this information is only 70 percent accurate. The work zone signs and other temporary signs are not provided by these navigation systems. BMW also claims that by using the traffic sign detection system along with the navigation system the accuracy of the information could be increased to 95 percent.

1.1.2 ARCHITECTURE OF THE SYSTEM

Figure 1.1 shows the global view of the automatic traffic sign detection system. A camera mounted on the rear-view mirror of the vehicle is used to capture the traffic scenes that appear in front of the vehicle. Each traffic scene is processed in real-time using the required algorithm to detect the sign board present in it and to extract the information from it. The extracted information is displayed on the dashboard of the vehicle.

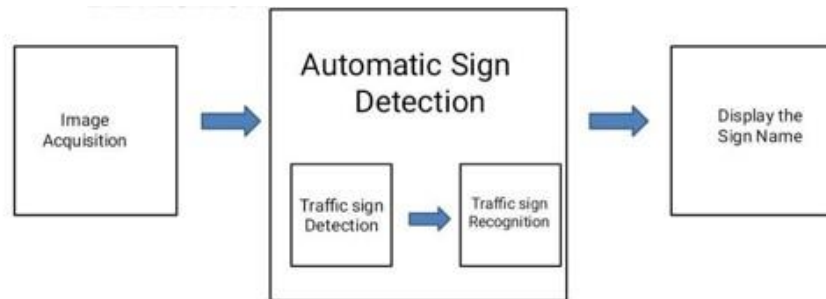


Fig 1.1 Block diagram of automatic traffic sign detection system

The traffic sign detection system consists of two major parts, hypothesizing the sign board locations in the input image and recognizing the hypothesized regions and extracting the useful information from them. Each part has multiple steps involved in it. The Block diagram of the automatic traffic sign detection system is shown in the above figure. The method operates on grayscale images, as it detects the traffic signs first and then recognizes them to verify it, and then extracts the information from the detected regions. This method for the traffic sign detection system consists of two major parts, hypothesizing the sign board locations in the input image and recognizing the hypothesized regions and extracting the useful information.

Each part has multiple steps involved in it. This method operates on grayscale images, detects the traffic signs first and then recognizes them to verify the hypothesis and to extract the traffic sign from the detected regions. These steps are discussed in detail in the following chapters.

1.2 ORGANIZATION OF THE REPORT

The project report is organized into six chapters.

Chapter 1: This chapter focuses on an introduction about Automated traffic sign Detection Systems. The recent trends and origin of our idea are discussed.

Chapter 2: This chapter provides a detailed literature survey related to the project, Literature Summary, Problem Definition, and Objectives of the project.

Chapter 3: This chapter describes the existing system and its working process.

Chapter 4: This chapter explains the proposed system and elucidates the materials and methods used in this project.

Chapter 5: This chapter provides the hardware results demonstrated with the proposed methods and techniques followed by a discussion about the project.

Chapter 6: This final chapter concludes the project. It also gives the scheme of the future work that is how the project can be enhanced further. Finally, the references referred to precede the project are given in brief.

CHAPTER 2

LITERATURE SURVEY

This chapter provides detailed literature survey related to the project, Literature summary, Problem definition, Objectives of the project. The intelligent driver assistance systems are an interesting area of research, because it comprises of topics which are real and very common. For example, the auto parking system has been successfully implemented in new vehicles. Similarly, the other topics like sign board detection, lane change detection and speed detection can be dealt using existing technologies to provide extra comfort and safety to the driver. A lot of research work is going on in these topics. The traffic sign detection system employs two most important image processing methods namely object detection and object recognition. The existing algorithms in these areas, used for the application, are discussed separately in the following sections.

2.1 SYSTEMATIC REVIEW

Cabrera-Cósetl et al. (2019) explained the human knowledge based fuzzy models to make decision in the problem of automatic car parking on scale model of an automatic car with satisfactory results. The author-generated models and training interface in MATLAB and its outcomes are a text file containing the fuzzy models, which sends to the car successfully to execute the parking action. The author also demonstrated the utility and efficiency of the linguistics models used to implement real solution of complex systems and the facility in the technological implementation in this article.

Faheem et al. (2022) analyzed different Intelligent Parking Services for parking guidance, parking facility management and gives an insight into the economic analysis project. It may reduce the problems arise due to non-availability of a reliable, efficient and modern parking system. It also helps in preserving the environment by reducing fuel and time.

Yuwei Chen et al. (2021) analyzed the suspension, Steering and braking system combinedly over integrated vehicle dynamics and his investigation has been focused on coordinating the interactions and function conflicts between the steering system and the suspension system by using a multivariable control approach (stochastic sub-optimal control strategy). The author also demonstrates Simulation results which shows that the integrated control system is effective in improving the overall vehicle performance including handling, lateral stability, and ride comfort, compared to either the EPS-only system or the ASS-only system, and the passive system.

Jarrold M Snider (2019) derived and compared selected path tracking methods. The importance of modeling vehicle and path dynamics is highlighted as vehicle speed is increased and paths become more varying. The author describes the varying levels of precision, complexity, implementation ease, path requirements and robustness associated with these path tracking methods. The author in this also pronounces algorithms over kinematic controllers, linear quadratic regulators, with optimal preview methods as part of this article.

Oentaryo and Pasquier (2020) presented the general structure, working principles, and training cycle of the Gen SoYager FNN. The author summarizes self-organization (clustering), rule formulation and parameter learning (using back propagation) phases and takes place in a single pass of the training data and this allows the Gen so Yager

network to perform on-line learning and is suitable for even (some) real-time applications. The Yager inference scheme employed by the, Gen So Yager FNN adopts the implication- based model of fuzzy relation. It computes the final output based on the level of disagreement instead of agreement.

Fei-Xiang Xu et al. (2019) suggested the following to enhance the handling stability of the 4WS vehicle, a mixed H_2/H_∞ robust controller. The author analyzed various mathematical models like linear 2DOF vehicle model, the nonlinear 8DOF vehicle model incorporating the Dug off tire model, the driver model, and the rear wheel electrohydraulic system model and laid the model foundation for the controller design. The author concluded as, combining the simulation results with the experimental results, the 4WS system with the H_2/H_∞ controller outperformed those of the H_∞ controller and the proportional controller in terms of vehicle handling stability and robustness.

Masiha Sabnam et al. (2016) discussed Radio frequency technology (RFID), and Infrared (IR) sensors. Microcontroller that exhibits a very useful model in automation of vehicle parking system in mall/building. The author discusses slot availability details by using an IR sensor, RFID system and an updated periodically database. Entry point and exit point have been controlled by IR sensors and RFID readers, labels and barriers respectively. The author also narrated the pros and cons of using various methodologies and components in this article. The last decade witnessed increasingly rapid progress in self- driving vehicle technology, mainly backed up by advances in the area of deep learning and artificial intelligence. The objective of this project is to survey the current state-of-the-art deep learning technologies used in autonomous driving. It has started by presenting AI-based self-driving architectures, convolutional and recurrent neural networks, as well as the deep reinforcement.

learning paradigm gain insight into the strengths and limitations of deep learning and AI approaches for autonomous driving and assist with design choices.

Yatin Jog et al. (2015) suggested smart parking views over Automated parking. The author described the smart parking solution is to enable both drivers and parking managers in optimizing parking capacity with car sensors, wireless communications data analytics. The author discusses Wireless Sensor Networks, RFID Technology, using GSM and RFID, QR code-based Vehicle Parking System, and Multi-level Car Parking System using Image Processing.

Janhvi Nimble et al. (2016) proposed a Smart Parking system designed with a mechanical model with an image processing facility. The car would be parked with the use of a lift at multiple levels. Also, image processing is used to capture the number plate and store it in database for comparison to avoid illegal car entry. As per the Author, this also ensures quick and automated parking and easy retrieval of vehicles with low maintenance. On a future enhancement note, this project can be extended for an online booking also few additional improvements which guides the drivers to reach the parking slots easier.

Shidian et al. (2017) discussed recognizing vertical parking spaces, and the detection of vehicle orientation. In this project, large number of samples are used to train the AdaBoost vehicle detector, and transplant test is conducted in his project and then, red taillights are identified using the color model algorithm. The parking orientation of the vehicle is obtained, by analyzing the working principle through automatic parking system, a parking scene model is established.

To achieve vertical parking in practice, the parking movement model and parking movement constraints are fully analyzed, and a reasonable and feasible vertical parking path planning program is utilized.

This program provides a basis for the following parking steering wheel control and path adjustment and to correctly and accurately reject the vehicle movement, an automatic parking path tracking controller based on the vehicle dynamics model is designed to control the vehicle tracking planning route, thus improving parking accuracy.

Abdullah Waseem et al. (2013) analyzed finite element analysis of pallets with three different materials viz. PVC, PE, PC, HDPE, Wood and found the stress and deflection analysis using ANSYS software. Highest deflection is analyzed in PE and lowest in wood by ANSYS Software. Stresses generated are almost the same in all plastic materials except wood. From these results, PVC is considered based on minimum deflection out of other plastic material. This research helps in identifying a method of selecting materials based on their behavior shown by different materials when subjected to different inputs.

Shaoping Lu et al. (2017) utilized pallet-throughout as pallet circulating mode, the project constructs a pallet- throughout closed-loop system based on RFID between industry enterprise and commercial enterprise. In this system RFID technology successfully resolves the problem that exists with pallet- unit information collecting. The author advises upgrading solution with some additional RFID and WMS and design middleware for system for good feasibility and flexibility for industrial enterprises or commercial enterprises. The author also suggests Pallet-unit system based on RFID offers an effective method to improve logistics efficiency.

Nursalbiah Nasir et al. (2021) described about the CAD analysis with acceptable design for rotary car park system. The author considered the design process based on the size and weight of the car commonly being found which is sedan type. The author has used Solid works software for 3D modeling the whole structure of rotary system and analyzed hangar bar and frame bar that supports the pallet structure to predict stress using the software.

Maziar Mashaei et al. (2020) presented an optimization algorithm of the most challenging design of Handling and Locating pallet system (HLPS) with decoupled multiple loops. The author developed a novel design constraint for pallets and buffers and a new optimization algorithm under linear constraint models with design variables of number of pallets, conveyor velocity, scheduling of parts, and buffer capacities. The author also specified this algorithm is applicable for wide range of flexible manufacturing systems involving low number of part types, this can also be upgraded to large number of part types or modules in the optimization model leads to the huge amount of computation time.

DaríoMaravall et al. (2019) proposed a model-based neuro controller using reinforced learning algorithm for making decisions in a moving car like robot. The respective control actions were sensor-based and consist of a dynamic, neural based process in which the neuro controller optimizes the adhoc performance that builds interactions with the environment. The resulting model was eventually utilized to generate the control actions. Simulated experimentation has also shown that there was an improvement in robot behavior when a model is used, at the cost of higher complexity and computational load.

Peizhi Zhang et al. (2019) adopted reinforcement learning to perpendicular parking so that the vehicle can continuously learn and accumulate experience from considerable parking attempts. The author ensures that the parking slot can be obtained continuously during learning, a parking slot tracking algorithm is proposed based on fusion of vision and vehicle chassis information. This approach is based on the images of parking slots which may require image acquisition.

Ruizheng et al. (2013) proposed an RL decision-making method for autonomous vehicles. These decision-making policies to be taken by the autonomous vehicle to follow the vehicle was focused. Though the simulation model is good, the parking decisions were not considered.

Xin Xu et al. (2018) proposed a reinforced learning approach to autonomous decision making of intelligent vehicles on highways. However, the author has improved the decision-making algorithm for vehicle following and not for parking decisions.

2.2 SUMMARY

From the literature survey mentioned above, there are preexisting systems that helps in the horizontal down tilt of the antenna that only can have 180 degrees tilt and can only have network losses, but it won't include the antenna tilt with any kind of devices. There are many journals and articles regarding Remote Electric Tilt. With this RET device, the disadvantages of the existing system and updates with 360 degrees of tilting of the antenna can be rectified.

2.3 LIMITATIONS IN THE EXISTING WORK

From the detailed literature survey, it is observed that several numbers of research have been made in Automatic car parking systems and there also exists few high-end cars with Automatic car parking feature. Various types of solutions have been provided for autonomous / intelligent / smart car parking systems using different electronics and computer science technologies. However, it is observed that only very few attempts have been made through Reinforce learning algorithm for car parking for making decisions to find the shortest route. And the parking solutions for the existing cars are still found complex and costlier for a new environment. A pallet- based parking system could be simple and cost-effective system of parking and hence attempt has been made to design and develop a Pallet based parking system using Reinforced learning for multi criterial decision-making.

2.4 PROBLEM DEFINITION

- To develop an automated traffic sign-based vehicle control system using deep learning algorithm.

2.5 OBJECTIVE OF THE PROJECT

- To identify the methodology by a detailed literature survey
- To design the detailed block diagram of the automatic traffic sign-based vehicle control system.
- To identify the accuracy of traffic sign recognition based on shape and color using image processing.

CHAPTER 3

EXISTING METHODOLOGY

This chapter describes the existing systems and its function. To implement the multi-objective decision model, a threshold lexicographic ordering method is introduced and integrated with the learning-based algorithm.

3.1 EXISTING SYSTEM

Agent-based approaches have gained popularity in engineering applications, but its potential for advanced traffic controls has not been sufficiently explored. This existing system presented a multi-agent framework that models traffic control instruments and their interactions with road traffic. A constrained Markov decision process (CMDP) model issued to represent agent decision making in the context of multi objective policy goals, where the policy goal with the highest priority becomes the single optimization objective and the other goals are transformed as constraints. A reinforcement learning-based computational framework is developed for control applications.

Algorithms like lane detection has two approaches namely feature based technique where in it localizes the lanes in the road images by combining the low- level features, such as painted lines or lane edges and second is the model-based technique wherein a few parameters to represent the lanes. The tremendous growth in transport systems and the increase in the number of vehicles over the last decades have created a significant problem in urban areas, namely traffic congestion.

This model comprises traffic identification, prediction and congestion avoidance phases. The system checks for the fitness function to determine the traffic intensity and further use predictive analytics to determine the traffic level in future. This also integrates fuel consumption model to save time and energy. Followed by this framework, predicts short-term traffic flow using structure pattern and regression methods. Short term traffic prediction is one of the required fields of study in the transportation domain. It is beneficial to develop a more advanced transportation system, to control traffic signals and avoid congestion. It will improve the traffic system, also enhance green priority, rerouting, fuel consumption and timesaving. The traffic flow structure pattern can be constructed from freeway toll data. Based on the pattern, a prediction method was utilized, which is based on LWL and regression.

Then this framework develops an efficient, intelligent traffic light control and deviation system for reducing the traffic congestion to achieve free flow of vehicles. Traffic light control systems are widely used to monitor and control the flow of automobiles through the junction of many roads. Intelligent traffic light control is critical for an efficient transportation system. This system is composed of Traffic Light Controller (TLC) and Traffic Light Deviation (TLD). The TLC system uses three agents to supervise and control the traffic parameters system deviates the vehicles before entering the congested road.

The Efficient Intelligent Traffic Light Control and Deviation System (EITLCD) contains five elements (i) Sensors: Magnetic sensor is chosen as a traffic sensor. (ii) Data Collector Agent: it examines the data composed through the sensors; they detect, count and categories the vehicles. (iii) Data Processor agent: it operates the vehicle length, velocity and relative quantity of traffic. The data processor will manipulate the traffic data, and the input will be given to an intelligent Traffic Light Controller (TLC) system. (iv) Intelligent TLC: Supervisor agent controls the traffic signal.

This system has a threshold value depending on the traffic area and if the value exists then an alert signal will be sent to the deviation system. (v) Intelligent TDS: If a congestion state is predicted, the deviation system provides a signal to deviate the vehicles arriving in the congested path. Furthermore, it utilizes Internet of Things (IoT) based intelligent transportation system for providing smart traffic solutions for the smart city. IoT involves connecting physical objects to the Internet to build smart systems. The growing population in metropolitan areas in this modern age requires more smart services of transportation. Achieving smart and intelligent transportation involves the use of millions of devices equipped with the Internet of things. IoT based Intelligent transportation system (IoT-ITS) helps in automating railways, roadways, airways and marine, which enhances customer experience about the way goods are transported, tracked and delivered. For an efficient smart Intelligent Transportation System (s-ITS), components such as Central Server, RFID device, Sensors, Lighting control unit and EBOX II are used.

Central Server plays a significant role in providing resilience during any malfunctioning of the system. RFID helps in communicating data flow information between the cars to EBOX II. This RFID device has tags, antennas to transmit information and readers to decode the data. It involves the smart building of an intelligent transport system with the ability to tackle real-time issues. The Vehicular location Tracking reduces the time period of the selection of paths for the vehicles as well as easy location estimation. The Intelligent vehicle parking system. The vehicle parking decision is made by the outcome factors like occupied or free factor. Communication within a VANET: the sensors track the location of the vehicle and its status in the current traffic scenario, and Vehicular Big-data Mining: the signals be communicated to the vehicles through the mining of current traffic status.

Traffic signals operate intelligently to enable emergency vehicles (Ambulance) to move on without any hassles to the commuters. The rapidly increasing vehicle population in India causes many issues in transport management for emergency vehicles such as Ambulance, Fire service, and other emergency vehicles, etc., in the cities and towns of the country. Many patients are brought dead to the hospital due to unpredicted traffic and inadequate response from the people in roadways compared with other countries. To minimize the traffic problems typically faced by ambulances, an intelligent transport system is used. It is integrated with sensor information and communication technologies to achieve traffic efficiency improve environmental quality, save energy, converse time and enhance safety and comfort, especially for ambulance drivers.

This system has two modules which are communicated and controlled by sophisticated hardware components: Intelligent Traffic Light Control System and Ambulance Control System. The Intelligent Traffic Light Control System can ensure the current duration time and give priority to some lanes according to the real-time traffic conditions. The system can ease traffic jams by raising the management level and improving the utilization rate of road effectively. The ambulance control system significantly solves ambulance problems like the fastest routing of an ambulance. This system also analyzes ways interrupted by the congestion and other activities during peak hours and calculates the fastest route.

3.2 IMAGE PIXEL

Connected component labeling (CCL) is used to label the connected regions in the input image. It labels each pixel by studying its neighborhood pixels' labels and deciding which pixel is more appropriate for the center pixel. We have quite a few algorithms existing in the literature to perform connected component labeling technique. Different criteria can be used to classify these algorithms. Some algorithms note the label equivalence and resolve them while some other algorithms label the image without the necessity of resolving the label equivalence.

The algorithms that consider label equivalence can be divided based on the scan method they use. In multi-scan algorithms the image is scanned multiple number of times in forward and backward raster directions, interspersed with label equivalence resolution. In two pass algorithms the labels assigned in the first scan are resolved and updated in the second scan. In single scan algorithms the label equivalence is resolved and updated in parallel with the assigning labels. The algorithms that do not consider resolving the label equivalence have their own kinds of methods to avoid the analysis of redundant labels. Under this category, multiscan algorithms go through the image in forward and backward raster directions alternatively, until there will be no changes in the assigned label values. In contour tracing algorithms it is achieved by tracing the contours of the objects.

The multi scan algorithms require more computations compared to the two pass algorithms. It is observed that a two-pass algorithm is more appropriate for our application. We have chosen the basic two pass algorithm which employs label resolution and modified it to adapt it for the grayscale images. The original two pass connected component labeling algorithm is explained in the next section.

3.3 GRAY SCALE IMAGES

Unlike binary images grayscale images have intensity values spread over a range. The pixels belonging to a single object need not possess equal intensity values. The pixels of different objects or regions are distinguished by their connectivity and intensity values. The pixels belonging to the white region of the traffic sign are connected and have intensity values approximately equal.

These features of the sign board make us use CCL algorithm for detecting this region in the input image. So, in order to adapt the CCL technique for grayscale images few changes are made in its original algorithm. The steps involved in the modified algorithm are discussed below. In a binary image two adjacent pixels are considered as connected if their values are equal otherwise, they are considered as disconnected. In a grayscale image two adjacent pixels can be considered as connected if their intensity values are closer to each other. The difference between their intensity values is calculated and if it falls below a threshold value then the two pixels are considered as connected.

The modified CCL algorithm also consists of three phases in it, scanning phase, label resolution phase and label updating phase. The same forward raster scan mask is used to consider the 4- neighborhood in this algorithm. At each pixel the mask is applied and the connectivity condition, explained before, is checked between the center pixel and each of the 4- neighbor pixels. If none of the 4- neighbor pixels satisfy the connectivity condition, then a new label is assigned to the center pixel. If only one of the 4- neighbors satisfy the connectivity condition, then its label is assigned to the center pixel. If two or more of the 4-neighbor pixels satisfy the condition, then minimum label among them is assigned to the center pixel and all these neighbor labels are considered as equivalent and stored in a one-dimensional array. The following equations explain these steps.

$$\left\{ \begin{array}{ll} g(x, y) = \{m, (m = m + 1) & \text{if } \{i, j \in N\} \text{ abs}(b(x, y) - b(i, j)) > T \quad (3.1) \\ gmin(x, y) & \text{otherwise;} \end{array} \right. \left\{ \begin{array}{l} gmin(x, y) = \min [\{g(i, j) [i, j \in N \& \text{abs}(b(x, y) - b(i, j)) \leq T\}] \quad (3.2) \end{array} \right.$$

	As a no left turn limit indication. Prevents the driver from taking left turns
	As a no-right turn indication. Prevents the driver from taking right turns
	As a school zone indication. Cautions the driver to either slow down or stop the vehicle.
	As a signal to stop. It stops the system and executes a stop operation

Fig. 3.1 Road Sign and Corresponding Actions

3.3.1 APPLICATIONS

It is used to minimize accidents in accident-prone zones such as school, college and hospital zones by controlling the vehicle and making it adhere to the appropriate rules on the roads using image processing and giving a warning to the driver about such zones.

3.4 WORKING PROCESS

The INDIAN traffic signs come in round, rectangular, triangular and octangular shapes. It also comes in different colors such as red, yellow, green, and blue. Red is typically used on road signs to denote laws that must be followed and potential dangers. Regulatory and cautionary signage typically employs the color red.

The color yellow advises motorists to drive more slowly or cautiously. Signs that provide instructions or guides to a certain location are typically highlighted in green. Blue with a rectangular shape is commonly used to represent a nearby destination. Regulatory signs frequently combine blue and circular shape combination. Initially, the input image is converted into a grayscale image. A sample grayscale traffic scene is shown in Figure 3.3.



Fig.3.2 Sample Traffic Scene

To extract the traffic sign information from the sign board present in the input image, first the signboard must be in the input image. The first phase of our algorithm tries to detect the traffic signs present in the input image. Here the detection phase does not require high accuracy, because the recognition phase is able to recognize the true and false sign board regions. The first phase of the algorithm hypothesizes the traffic sign locations in the input image using the appearance-based features of the sign board. First the input image is divided into a set of regions, using a connected component labeling technique adapted for grayscale images, and then each region parameters are calculated to choose the possible regions.

3.5 PROPOSED SYSTEM

A feed forward Neural Network is a class of deep learning, artificial neural networks, and most commonly useful for several analyses. They visualize metaphors and Numerical data. It can use a difference of multilayer perceptron designed to need minimal pre-processing. It is very similar to normal Neural Networks. They are made up of neurons that have learned weights and biases. Each neuron receives several inputs, performs a dot product and optionally follows it with a non-linearity.

The complete network at rest articulates to achieve the function from the raw input data on one end to achieve the class at the other end. It can make a clear hypothesis that the inputs allow us to encode certain possessions into the FFNN process and then, make the forward function more efficient to implement. They greatly reduce the number of parameters in the network Neural Networks consider an input and transform it through a series of hidden layers. Each hidden layer is made up of a set of neurons, where each neuron is fully connected to all neurons in the previous layer.

CHAPTER 4

PROPOSED METHODOLOGY

This chapter discusses the recognition phase of the algorithm. The first phase of the algorithm detects the regions that have similar features as a traffic sign board. The size and the aspect ratio of the region are the two parameters used to hypothesize the traffic sign regions in the input image. To detect the traffic sign board accurately its informative part, which is characters, should also be identified. Initially, we don't know the exact location of the sign board in the input image so, instead of performing the recognition for the entire image, by hypothesizing the sign board locations in the image, the recognition is performed only for these regions, thus reducing the computation time required. This chapter provides a few introductory points about object recognition and then discusses the recognition method employed in our algorithm.

4.1 BLOCK DIAGRAM OF PROPOSED SYSTEM

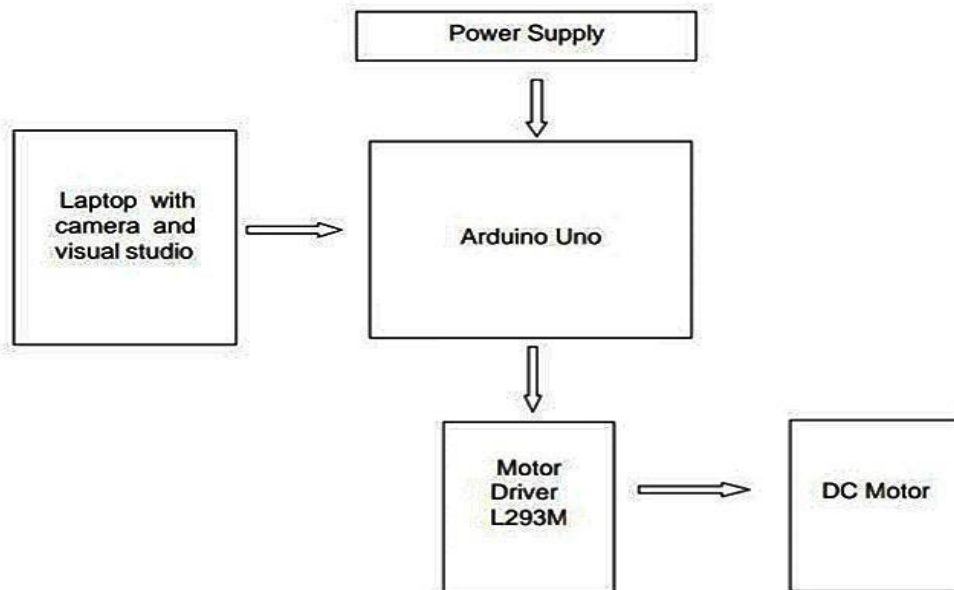


Fig.4.1 Block Diagram of automatic traffic sign-based vehicle control system

Neurons in a single layer function in a separate manner and do not share any relations. The very last fully connected layer is called the output layer and it represents the class achieved. There are three main Layers that build FFNN. They are Convolution Layer, Pooling Layer, and Fully Connected Layer. The input [HxWxD] initially will hold the input values and in this case an input data of width (rows), height (columns), and depth 1 are considered. The convolutional layer will compute the output of neurons that are connected to local regions in the input. It has an independent set of filters that work with the input and a small region is connected to the input volume.

The output volume will be as [32x32x12] and 12 is the number of filters that were used. RELU layer applies an element wise activation function, such as $\max(0, x)$ where zero is threshold value. Here the volume size remains unchanged ([32x32x12]). POOL layer performs down sampling operation throughout the spatial dimensions (width, height). Its resultant volume is [16x16x12]. Its function is to reduce the spatial size of the representation progressively.

Therefore, it reduces the number of parameters and computation in the network. A fully connected layer will compute the class scores. It results in volume of size [1x1x10], where each of the 10 numbers corresponds to a class achieve. As with ordinary Neural Networks and as the name implies, each neuron in this layer will be connected to all the numbers in the earlier volume. In this fashion, FFNN transforms the input raw data, layer by layer from the input raw values to the final class scores. Particularly, the Convolution and Fully Connected layers perform transformations that are a function of not only the activations in the input volume, but also of the parameters (the weights and biases of the neurons).

4.2 HARDWARE DESCRIPTION

The proposed system architecture consists of four primary hardware components, including an Arduino Uno, L298n motor driver, a DC motor, and a DC power supply.

4.2.1 Arduino Uno

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc and initially released in 2010.



Fig.4.2 Figure of Arduino

Specifications:

- It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button.
- The operating voltage is 5V.
- The recommended input voltage will range from 7v to 12V.
- The input voltage ranges from 6v to 20V.
- Digital input/output pins are 14.
- Analog i/p pins are 6.
- DC Current for each input/output pin is 40 mA.
- DC Current for 3.3V Pin is 50 mA.

4.2.2 L298N Motor Driver

The L298N is a dual H-Bridge motor driver which allows speed and direction control of two DC motors at the same time. The module can drive DC motors that have voltages between 5 and 35V, with a peak current up to 2A.

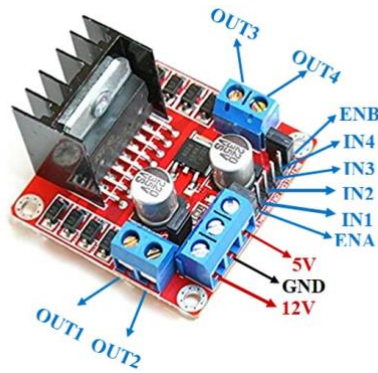


Fig.4.3 Figure of L298N motor driver

Specifications:

- Motor Supply Voltage (Maximum): 46V
- Motor Supply Current (Maximum): 2A
- Logic Voltage: 5V
- Driver Voltage: 5-35V
- Driver Current: 2A
- Logical Current: 0-36mA
- Maximum Power (W): 25W
- Current Sense for each motor
- Heatsink for better performance
- Power-On LED indicator

4.2.3 DC Motor

DC motor is an electrical machine which converts electrical energy into mechanical energy. DC motors take electrical power through direct current and convert this energy into mechanical rotation.

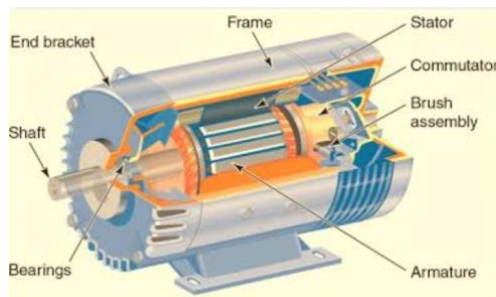


Fig.4.4 Figure of DC motor

Specifications:

- Frame sizes from 8 to 35 mm.
- Speeds from 5,000 to 14,000 rpm.
- Continuous motor torque – 0.36 to 160 mNm.
- Coreless rotor design.
- Low rotor inertia.
- REE coil.
- High power to weight ratio.
- Neodymium magnet available in some brush DC motor models

4.2.4 DC Power Supply

This DC-DC 12V to 3.3V 5V 12V Power Module Multi Output Voltage Conversion is also known as Buck Converter or also as Step-Down Voltage Converter. The module is capable of altering the output of the power source/supply before supplying it to the load so as to deliver the specified power to your



Fig.4.5 Figure of power supply

Specifications:

- Output power on/off switch
- Two-row multiple pin outputs, easy to use and connect.
- Red LED illuminates when input DC power is supplied, and output is turned ON.

- Outputs
 - 3.3V DC at 800mA maximum current
 - 5.0V DC at 800mA maximum current
 - 12V DC (the 12V output is directly connected to the input, i.e., with 12VDC input, this output will be 12VDC; with 9VDC input, this output will be 9VDC)
- Small and Handy Module.
- Dual-side plate design.

4.3 WORKING METHOD CIRCUIT DIAGRAM

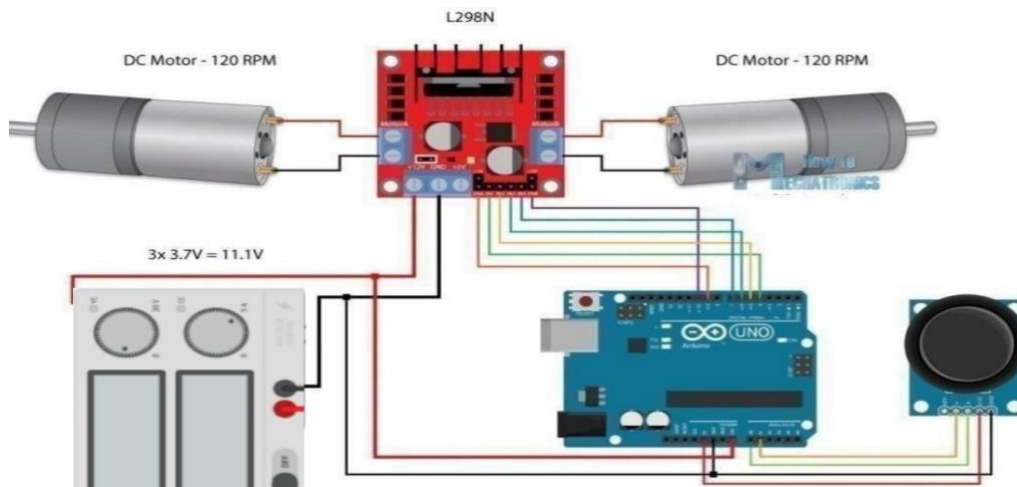


Fig.4.6 Circuit diagram of automatic traffic sign-based vehicle control system

The primary focus of this project is of cutting-edge approaches for automatic vehicle control that utilize the detection of traffic signboards. This system uses image processing technology, which is employed in a variety of industries including geoscience, healthcare, and space research. To prevent accidents and bottlenecks in traffic, traffic signs are crucial. Drivers frequently fail to heed traffic signs in front of their vehicles, which can have disastrous consequences. Annually, More than 1 million individuals die in automobile accidents as a result of drivers speeding, driving recklessly, breaking the law, due to fatigue, alcohol consumption, and even negligence, etc. These

accidents typically occur close to hospitals and schools. As a part of creating smarter automobiles, it is important to build and deploy an algorithm that automatically recognizes Indian traffic signs from the grayscale images obtained from a camera placed on a vehicle's inside.

On the side of lanes, if a signboard is present, the camera will detect it and send that input data for image processing and simultaneously classification process occurs and it gives command to Arduino uno to send input values to L298N motor to control the speed of the motor for driving in uniform speed with the DC integrated Circuit Supply power. Then the useful information from them is extracted by using deep learning algorithm. MATLAB, a high-level programming language, is employed for this process. Hence the output data will be displayed on the system dashboard or display screen. The motor driver will command and control the vehicle speed by using DC motors based on the output data values. This technology will help the driver by directly informing him of the allocated traffic restrictions.

4.4 SOFTWARE ALGORITHM FLOWCHART

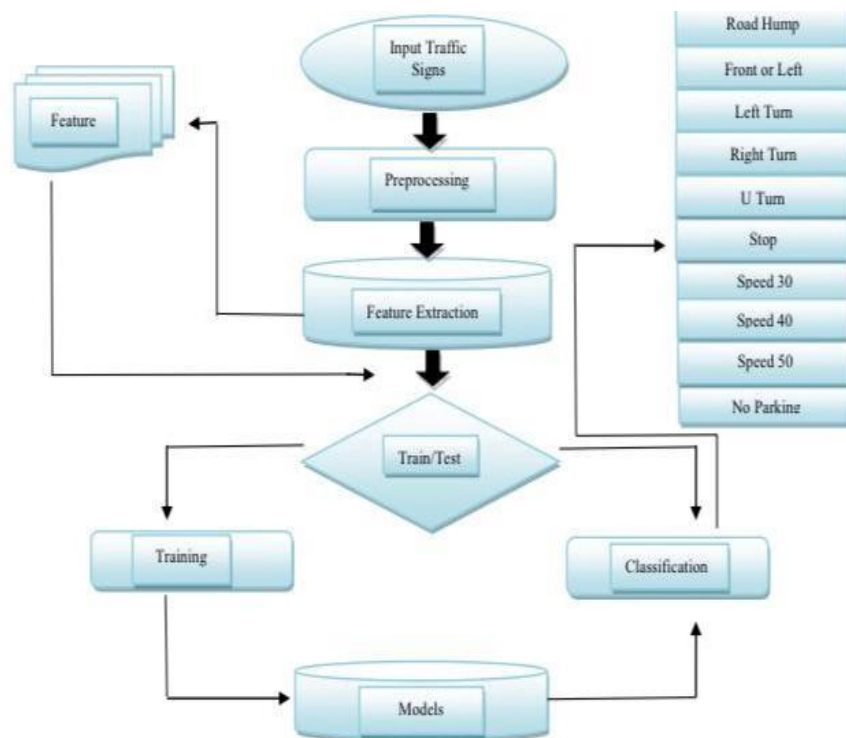


Fig.4.6 Flow chart of automatic traffic sign-based vehicle control system

If the proposed system captures an traffic sign images like No left turn, School Sign Or Hospital Sign by using IP enabled application for controlling the webcam in any model phones for high dimensional pixel images to be stored, then it will go through two phases, known as Sign detection and Recognition phase, Simultaneously it will go through the stored available data's for feature extraction into smaller grid of pixels by using VGG 16/19 Sub algorithm comes under Deep learning Algorithms which is inspired by Animal visual cortex.

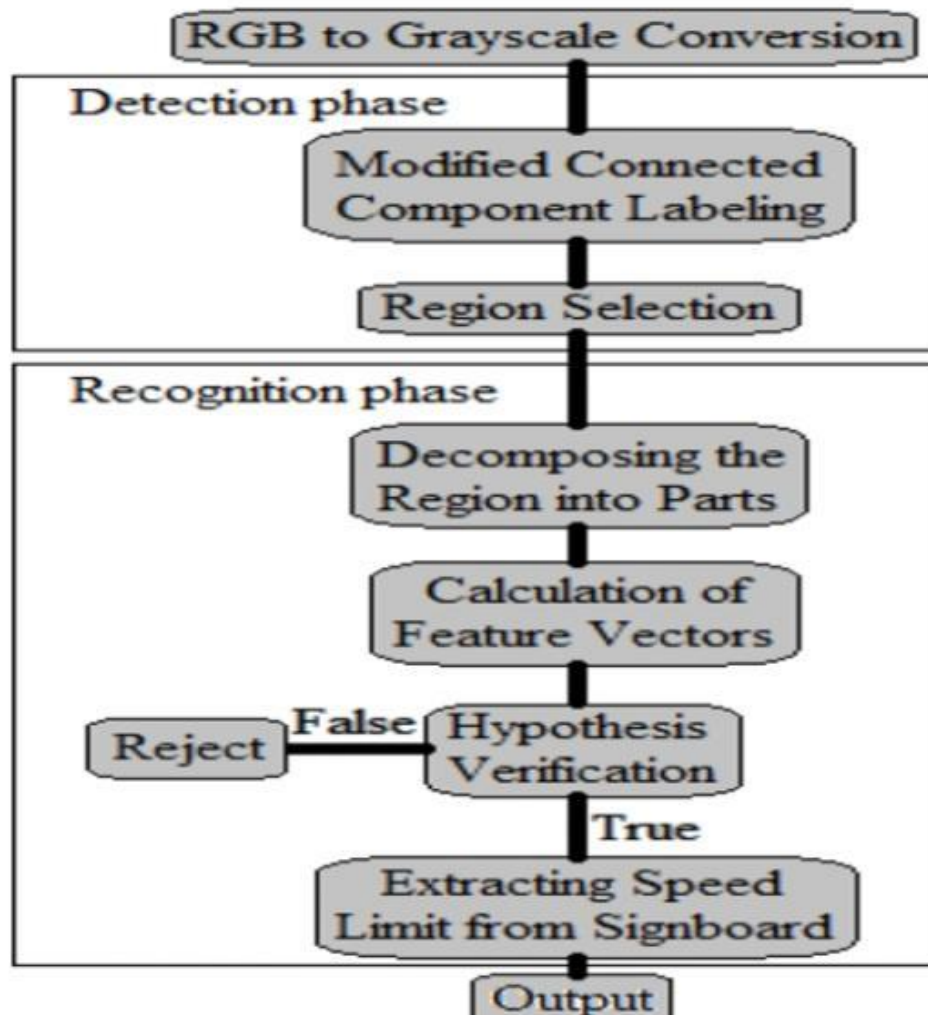


Fig.4.7 Flow chart of image processing in CNN

here RGB refers to Red,Green,blue images which will be converted to gray scale images which are commonly known as black and white images. In this, the first phase known as detection phase where the Connected Component labeling technology is infused, which defines it as Connected Component Labeling (CCL) is a basic algorithm in image processing and an essential step in nearly every application dealing with object detection. It groups together pixels belonging to the same connected component (e.g., object). Then particular regions will be selected from it reducing the spatial volume of images by process of pooling layer which is a sub layer of CNN Algorithm.

The second phase which is known as Recognition phase where the selected regions will be divided into small grids and each part will be calculated using vectors method for evaluating the magnitude of each part , the calculated parts will be compared with stored data's then if it's not a traffic sign which is stored , then it leads to reject it and if it represents the traffic sign then the traffic sign image will be extracted and displayed on the dashboard of vehicle and necessary action will be taken. The implementation diagram is shown in fig.4.7 where the IC-L298N enables parallel threading for the supply and direction of pulses from the Arduino Uno. A convolutional neural network (CNN) algorithm is utilized, along with a CNN sub-algorithm known as VGG16.

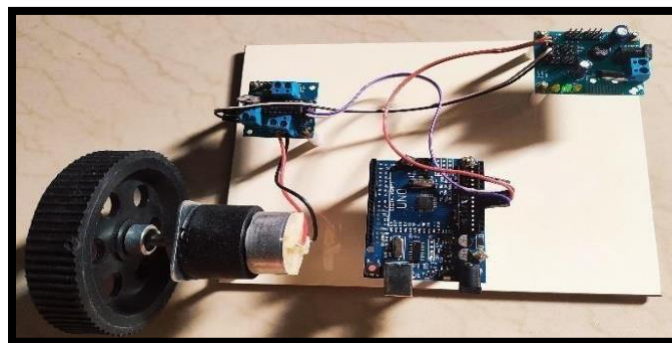


Fig.4.8 Implementation diagram of the automatic traffic sign-based vehicle control system

4.5 INTRODUCTION TO OBJECT RECOGNITION

Object recognition is one of the most important topics of digital image processing. To recognize an object is to identify it or name it or specify its class or type. The applications of object recognition are spread in many fields like medical imaging, industrial applications, remote sensing, sorting and categorizing, image retrieval and object detection.

Depending on the requirements, constraints and nature of the application, different algorithms are proposed for object recognition. These algorithms can be categorized based on different criteria such as the type of features used to describe the object, or the approach of the recognition scheme used. Different algorithms utilize different features of the objects to perform object recognition. Based on object representation these algorithms can be broadly divided into two types.

- i. Shape-based methods.
- ii. Appearance-based methods.

Shape-based algorithms utilize the geometrical features of the objects like corners, edges, blobs, and ridges, and appearance-based algorithms used the features calculated from the regions covered by the objects. Based on the recognition schemes used, object recognition algorithms can be categorized into three types.

- i. Invariant properties methods.
- ii. Parts decomposition methods.
- iii. Alignment methods.

The invariant properties of the objects under different object transformations are recognized in the first kind of algorithms. These methods employ techniques like feature spaces, clustering and separation. The second kind of method relies on decomposing the object into its constituent parts. Each part is recognized individually to identify the object overall. This approach leads to algorithms like symbolic structural descriptions, syntactic pattern recognition and feature hierarchies.

In the alignment methods an object is recognized by measuring the transformations the object has gone under and comparing the object with the available object models in measured transformations. It is not necessary that a given recognition algorithm should belong to only one these kinds, it could combine the features from different methods. Our proposed algorithm is one such kind; it aims at dividing the traffic sign into its constituent parts, which are characters printed on the sign board, and identifying them using their invariant properties.

4.6 STEPS INVOLVED IN THE PROPOSED ALGORITHM

The first phase of the algorithm extracted the possible traffic sign regions by analyzing the region characteristics in the input image. The second phase of the algorithm known as the recognition phase aims at classifying each extracted region as a traffic sign board region or not and extracting the assigned traffic information from it if it is classified as a traffic sign. For this the inner parts of the extracted regions need to be analyzed. Hence by analyzing the structural components of the extracted regions from the first phase of the algorithm, the sign board can be identified. The steps involved in the recognition phase are given as,

- i. Pre-processing
- ii. Feature extraction and
- iii. Classification

4.7 PRE-PROCESSING

This is the first step in the recognition phase. The extracted regions in the first phase possess a similar aspect ratio as the traffic sign. Each hypothesized region is processed one after the other. First, the region is resized to a common size to utilize the spatial information in the classification stage. Then it is normalized, converted into a black and white image and then it is inverted to convert the white background of the traffic sign into black and the black characters of it into white. Then each individual component of the resulting image is labeled using the following mat lab function,

$$L = bwlabel(bw, n)$$

Where 'bw' represents the black and white image, 'L' is the matrix containing labels and 'n' is the variable specifying the type of connectivity of the objects in the input image. The value of 'n' can be 4 or 8 and when it is omitted the function assumes an 8- connectivity. The function assigns unique label values to each of the connected objects in the input image.

4.8 FEATURE EXTRACTION FROM HU MOMENTS

The next step in the recognition phase is extracting useful and robust features of the inner parts of the hypothesized region. The prominent or distinct properties, qualities or characteristics of an object are called its features. For example, corners, edges, texture, size, length, orientation and blobs are some of the features used in the image processing applications. Feature extraction refers to calculating these characteristics of the object that can be used to define it and represent it with less data than the original data.

The noticeable or distinguishable attributes, properties, or characteristics of an object are known as its features. The success of an image processing algorithm depends on the quality of the features used in it. If the nature of the features calculated under different normal and noisy conditions is consistent then the algorithm would be successful, otherwise the algorithm would not be successful.

Different applications encounter different kinds of situations like, occlusions, illumination variations, transformed images, background clutter and poor contrast etc. To design a successful algorithm, one needs to tackle these kinds of situations by focusing on extracting application-dependent features. For example, traffic signs tend to tilt, rotate, are viewed at different angles and partially occluded sometimes.

So, the features extracted under these kinds of noisy conditions should be able to capture the characteristics of the object, which would in turn help for a successful algorithm. Each hypothesized region in the detection phase of our algorithm is broken into its constituent parts in the previous step. In this (feature extraction) step, each part of the region is chosen to calculate its features. If the extracted region is a traffic sign, then the characters on it are its constituent parts.

The recognition of optical characters is one of the popular applications of image processing. In the case of a traffic sign image the recognition of only the specific characters written on it is required, but not all the characters. The general obstacles faced in this process are the rotated, rescaled, and tilted characters. Hence the extracted features should be robust enough to tackle these conditions.

In our algorithm features are extracted from Hu's moments. Hu's moments, also called geometric moment invariants, are introduced by Hu [18]. They were used in the identification of aircraft, radar images, matching of optical images and texture classification. They produce a set of invariant features and can capture the global and geometrical features of the image, which are considered as one of their major advantages.

Hu's moments are derived from the normalized central moments, which in turn are derived from the raw moments of the image. The raw moments of a digital image 'f(x, y)' of the order (p+q) are represented by the symbol, 'mpq' and are given as follows,

$$m_{pq} = \sum_x \sum_y f(x, y) x^p y^q \quad (4.2)$$

These raw moments are used to derive simple image properties like area and center of mass. The (p+q)th order central moments of the image 'f(x, y)' are given by the following equation.

$$\mu_{pq} = \sum_x \sum_y f(x, y) (x - \bar{x})^p (y - \bar{y})^q \quad (4.3)$$

Where,

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad \text{and} \quad \bar{y} = \frac{m_{01}}{m_{00}}$$

These central moments are made invariant to translation and variations in scale by multiplying them with a factor derived from the (00) th central moment. Image orientation information can be calculated from the central moments.

$$5_{pq} = \frac{\mu_{pg}}{\mu_{00}^{\gamma}} \quad (4.4)$$

Where,

$$\gamma = \frac{(p+q)}{2} + 1, \text{ for } (p+q) \geq 2.$$

Hu combined different orders of these normalized central moments to produce a set of seven moments that are invariant to translation, scale, and even rotation. These seven moments are given below,

$$\begin{aligned} \varphi_1 &= 5_{20} + 5_{02} \\ \varphi_2 &= (5_{20} - 5_{02})^2 + 4 5_{11}^2 \\ \varphi_3 &= (5_{30} - 3 5_{12})^2 + (3 5_{21} - 5_{03})^2 \\ \varphi_4 &= (5_{30} + 5_{12})^2 + (5_{21} + 5_{03})^2 \\ \varphi_5 &= (5_{30} - 3 5_{12})(5_{30} + 5_{12})[(5_{30} + 5_{12})^2 - 3(5_{21} + 5_{03})^2] + (3 5_{21} - 5_{03})(5_{21} + 5_{03})[3(5_{30} + 5_{12})^2 - (5_{21} + 5_{03})^2] \\ \varphi_6 &= (5_{20} - 5_{02})[(5_{30} + 5_{12})^2 - (5_{21} + 5_{03})^2] + 4 5_{11}(5_{30} + 5_{12})(5_{21} + 5_{03}) \\ \varphi_7 &= (3 5_{21} - 5_{03})(5_{30} + 5_{12})[(5_{30} + 5_{12})^2 - 3(5_{21} + 5_{03})^2] + (3 5_{12} - 5_{30})(5_{21} + 5_{03})[3(5_{30} + 5_{12})^2 - (5_{21} - 5_{03})^2] \end{aligned} \quad (4.5)$$

Out of the seven Hu's moments, when calculated them for different characters, it is observed that the last three moments are very minute and exhibit very little difference among different patterns of the characters. Keeping these observations in mind we have formed a feature vector that comprises of the first four moments, as shown below.

$$[M] = [\log(\varphi_1 * 10) \log(\varphi_2 * 10^2) \log(\varphi_3 * 10^4) \log(\varphi_4 * 10^4)] \quad (4.6)$$

The moments are multiplied by appropriate factors to bring equilibrium into the feature vector. By taking the logarithmic values the

dynamic range of the feature vector is reduced. These four features are used in the next step for classifying the hypothesized regions and extracting the traffic information from it if it is classified as a traffic sign.

4.9 REGION CLASSIFICATION AND SIGN EXTRACTION

The aim of this step is twofold. First the hypothesis generated in the first phase of the algorithm is verified by classifying the region as traffic sign or not. Second, the information regarding the assigned traffic sign available from the region is extracted. Both these parts require the recognition of the characters that are present on a traffic sign board. The characters on the upper half of the sign board will confirm its identity, while the lower half components will specify the assigned rule.

The features of these characters are matched with the features of the components of the detected regions for recognition. The components of the detected region are termed as objects and the characters present on a traffic sign board are termed as templates. The classification of the region is performed as follows.

4.10 CLASSIFICATION BY MATCHING FEATURES AND LOCATIONS

The classification of the regions is performed by matching the locations and features of the objects with that of templates. By observing the locations of the alphabet and characteristics in the traffic sign a location range is assigned for each alphabet for its beginning row and column. As mentioned already the hypothesized region is resized to the size of the database traffic sign image.

If the beginning row and column of an object in the region are in the range of a template alphabet's assigned location range, then the object and the template are considered as location matched.

Next the features of the location matched object and template are compared by measuring the correlation coefficient between their feature vectors. The feature vectors for the characters of the traffic sign are already calculated and stored in the database. These templates feature vectors are represented by the variable 'Mt' and the feature vectors of the objects are represented by 'Mo'. The correlation coefficient 'C' between the two feature vectors of the object and the template is calculated.

The features of these characters are matched with the features of the components of the detected regions for recognition. The components of the detected region are termed as objects and the characters present on a traffic sign board are termed as templates. The classification of the region is performed as follows.

If the value of the correlation coefficient between the object and the template is above a chosen threshold, then they are considered as similar otherwise it is not considered as a match. This process is repeated for all the objects of the detected region and the number of matches found is noted. If the number of matches found are sufficient, then the region is classified as a traffic sign otherwise it is rejected. An INDIAN traffic sign contains 10 alphabets in its upper half. If the number of matches found is greater than or equal to 6 then the region is classified as a traffic sign. The regions classified as traffic signs are further processed to extract the information regarding the assigned traffic rule.

4.11 TRAFFIC SIGN EXTRACTION USING NEURAL NETWORKS

The assigned traffic sign is printed on the lower half of the sign board. A multilayer feed forward neural network (FFNN) with a back propagation learning algorithm is used for this purpose.

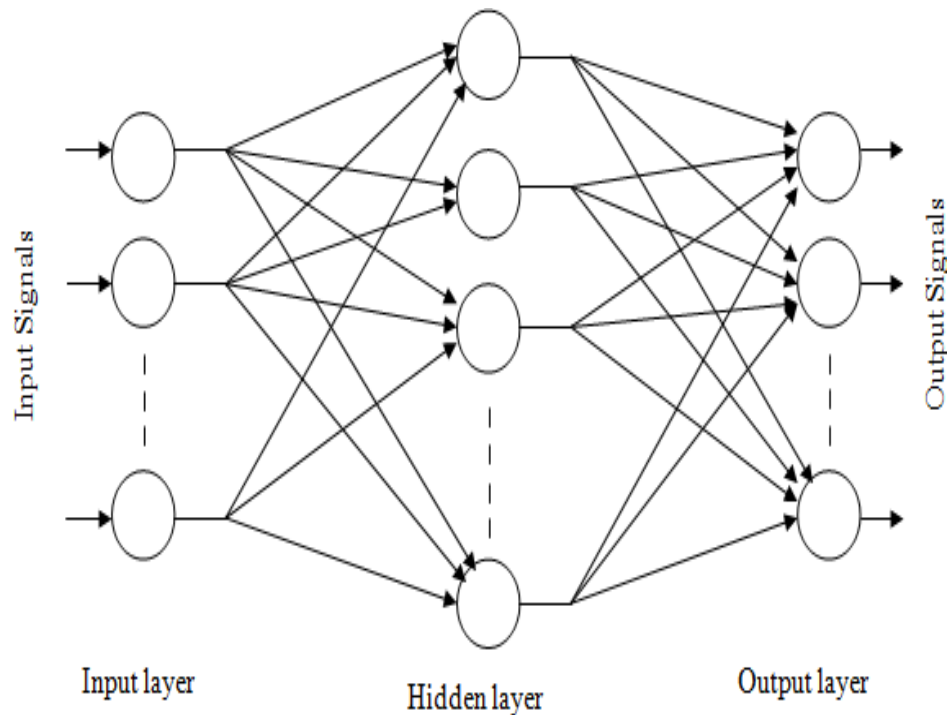


Fig. 4.10 Architecture of the multilayer neural network

A multilayer FFNN consists of one input layer, one output layer and at least one hidden layer in between them. Each layer consists of several neurons, which are analogous to the neurons in the brain. The neurons in the input layer are connected to the neurons in the hidden layer and the neurons in the hidden layer are connected to the neurons in the output layer as shown in the figure 4- 2.

A weight is associated with each neuron-neuron connection and the network learns by repeatedly adjusting these weights. Each neuron receives one or more input signals but always produces a single output signal. The input to the network is given to the neurons in the input layer and the output is obtained from the output layer neurons. The network is trained using the back propagation learning algorithm. The steps involved in it are explained as follows:

Step 1: Initialization

The weights and threshold levels of the network are chosen randomly from a small range.

Step 2: Activation

The inputs $M_1, M_2 \dots M_n$ from the training set and the corresponding outputs $yd,1, yd,2, \dots, yd,n$ are applied to the neurons in the input and output layers respectively.

$$y_j(p) = \text{sigmoid} [\sum_{i=1}^n M_i(p) \times w_{ij}(p) - \theta_j] \quad (4.7)$$

Where 'p' is the iteration value, ' θ_j ' is the threshold level at neuron j, ' w_{ij} ' is the weight of the link connecting neurons i and j and the sigmoid function is given by the following equation.

$$Y^{\text{sigmoid}} = \frac{1}{1 + e^{-X}} \quad (4.8)$$

Step 3: Adjusting the weights.

After feeding the input signals in the forward direction in the network to calculate the outputs at neurons in hidden layer and output layer, now the error values are calculated at output layer and are propagated in the backward direction.

a) The error and error gradient values for the neurons in the output layer are calculated:

$$e_k(p) = y_{d,k}(p) - y_k(p) \quad (4.9)$$

$$\delta_k(p) = y_k(p) \times [1 - y_k(p)] \times e_k(p) \quad (4.10)$$

Where e_k is the error at neuron k and δ_k is the error gradient value.

The corrections for the weights are calculated as,

$$\Delta w_{jk}(p) = \alpha \times y_j(p) \times \delta_k(p) \quad (4.11)$$

Where 'α' is the learning rate. The weight corrections are added to the weights in the current iteration to calculate the new weights.

$$w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \quad (4.12)$$

a) The error gradient value for hidden layer neurons is calculated,

$$\delta_j(p) = y_j(p) \times [1-y_j(p)] \times \sum_{k=1}^n \delta_k(p) \times w_{jk}(p) \quad (4.13)$$

And the weight corrections are calculated as,

$$\Delta w_{ij}(p) = \alpha \times M_i(p) \times \delta_j(p) \quad (4.14)$$

The weights of the connections are updated as,

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p) \quad (4.15)$$

Step 4: Iteration

The value of iteration p is increased by one and the steps 2, 3, and 4 are repeated until the chosen error criterion is satisfied. Some methods fix the number of iterations while others calculate the mean square error value to terminate the process. After training the network and adjusting the weights, the feature vectors of the inner parts of the detected region are fed as inputs to the network to recognize them. The output is collected from the output neurons, which give the assigned traffic sign.

CHAPTER 5

RESULT AND DISCUSSION

The proposed algorithm is experimented on different traffic scenes and the results obtained are presented in this chapter. The traffic scene, which appears in front of a vehicle, is captured by the camera mounted on its rear-view mirror and processed using the proposed algorithm in MATLAB.

There are two phases in the algorithm: sign board detection phase, which detects the possible traffic signs in the input image and sign board recognition phase, which verifies the hypotheses generated in the first phase and extracts the information from it by recognizing the inner components of the regions. Each phase has several steps involved in it. The results obtained after each important step are presented in this chapter in the order they are produced.

The detection phase employs the proposed modified connected component labeling technique, which adapts to the grayscale images. A threshold level of 35 is used to check the connectivity condition, performed in the CCL algorithm, which is observed to be the efficient value in labeling the traffic sign distinctly. The neural network used in the recognition phase consists of 4 neurons in input layer, 7 neurons in hidden layer and 9 neurons in output layer. This network is trained using Hu's moments of the characters written on a traffic sign board.

5.1 DETECTION PHASE

The image captured by the camera is converted into a grayscale image, which is shown in the figure below.



Fig. 5.1 original image

The modified connected component labeling technique is used to label this grayscale image. The algorithm labels the connected regions distinctly.



Fig. 5.2 Grayscale image labeling

After labeling the input grayscale image, the regions in the image are analyzed using the region characteristics. We have used the size and the aspect ratio of the regions to hypothesize the locations of the traffic sign in the input image.

5.1.1 Condition 1: Region Size

The input image contains a lot of small objects like the characters on signboards, numberplates and a few very big regions. We have chosen the regions with sizes in the range of 1000 to 10,000. This condition would be helpful to ignore the very small and very big regions. The size of the regions is calculated by counting the number of pixels in them. The variation of the region sizes in the labeled image (figure 5.2) is shown in figure 5-3, using the histogram of the labeled image. The horizontal axis represents the label value of the region, and the vertical axis represents the size of the region.

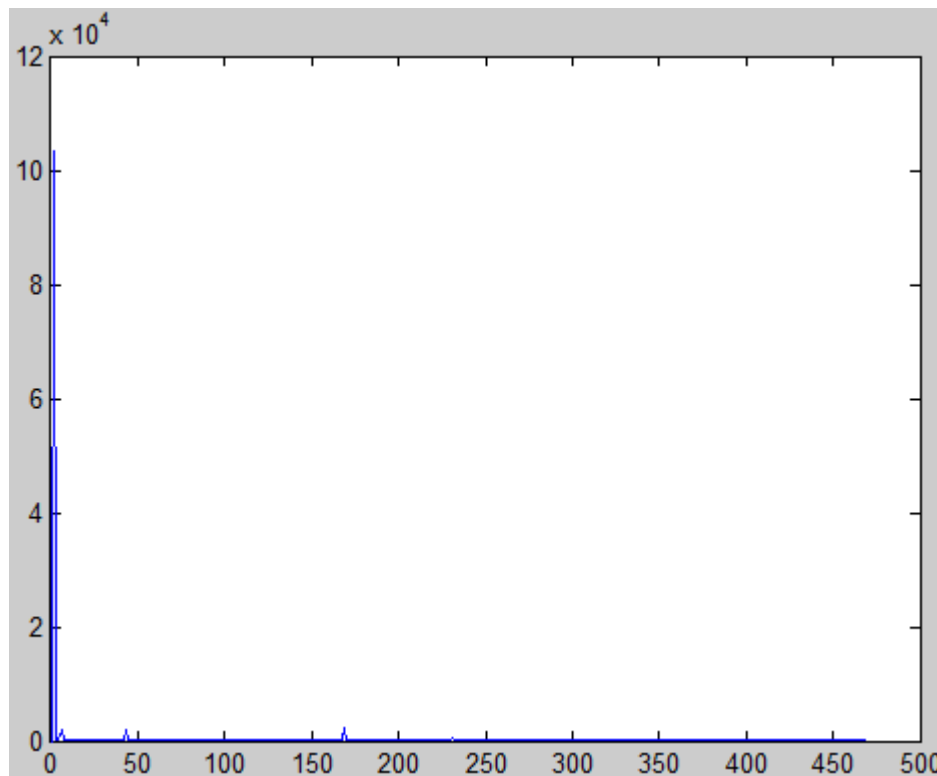


Fig. 5.3 Histogram of the labeled image

5.1.2 Condition 2: Aspect ratio

As mentioned in chapter 3, we have chosen the regions whose aspect ratio falls in the range of 0.58 to 1. The table below lists a few regions and their sizes and aspect ratios.

Region Label Value	Size of the Region	Aspect Ratio of the Region
5	1908	2.4
42	1830	0.807
168	2175	3.1

Table 5.1 Examples of few regions' characteristics from the figure 5.2

From the above table we can see that the region with label value 42 satisfies both our conditions. This region is hypothesized as the traffic sign region and the resulting image is shown in the figure below.



Fig. 5.4 Generated Hypotheses

5.2 RECOGNITION PHASE

The beginning and ending coordinates of the hypothesized region are noted and this part of the original image is stored in a separate matrix. This image is then resized to a common size, converted into a black and white image, inverted the colors and labeled the individual components in it.

5.2.1 Pre-Processing

Figure 5-5 (a) Detected region (b) Inverted black and white image (c) Labeled image The labeled image is used to select the individual components of the region and the black and white image is used to calculate their features.

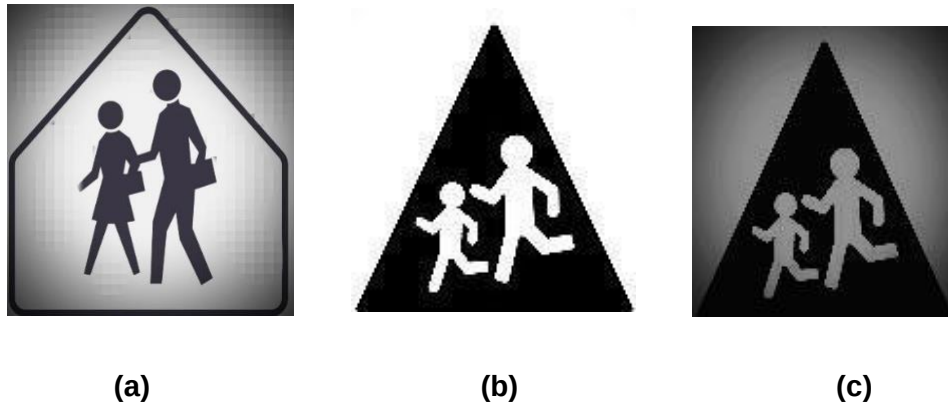


Fig. 5.5 preprocessing

5.2.2 Feature Extraction

In this step, each component of the hypothesized region is chosen separately, and its feature vector is formed by calculating Hu moments.

$$[M] = [\log(\emptyset_1 * 10) \log(\emptyset_2 * 10^2) \log(\emptyset_3 * 10^4) \log(\emptyset_4 * 10^4)]$$

5.2.3 Classification

In this step, the hypothesis generated in the detection phase is verified by recognizing the upper half components of the region. The classification method uses the geometrical features derived from Hu's moments and the spatial information of the individual components of the region. The region is classified as a traffic sign if there are at least 6 recognized characters at specified locations in the region. The results of the classification step are shown in the table 5.2.



Fig. 5.6 Analyzing the upper half of the detected region

No of components in the region	No. of recognized characters	Classification
10	10	School Sign

Table 5.2 Classifying the region by recognizing its components.

5.2.4 Traffic Sign Information Extraction

The region is further processed only if the region is classified as a traffic sign. The assigned speed limit is printed on the lower half of the signboard.



Fig. 5.7 Analyzing the lower half of the region.

We have trained a neural network with 4 input layer neurons, 7 hidden layer neurons, and 9 output layer neurons for recognizing the numbers from 0 to 8. The network is trained with a back propagation learning algorithm. The feature vectors of the traffic sign characters are fed as inputs to the network, and it is trained for 10,000 iterations. The image for each traffic sign is already available in the data base and the recognized traffic sign board is displayed as output.



Fig. 5.8 Recognized traffic sign

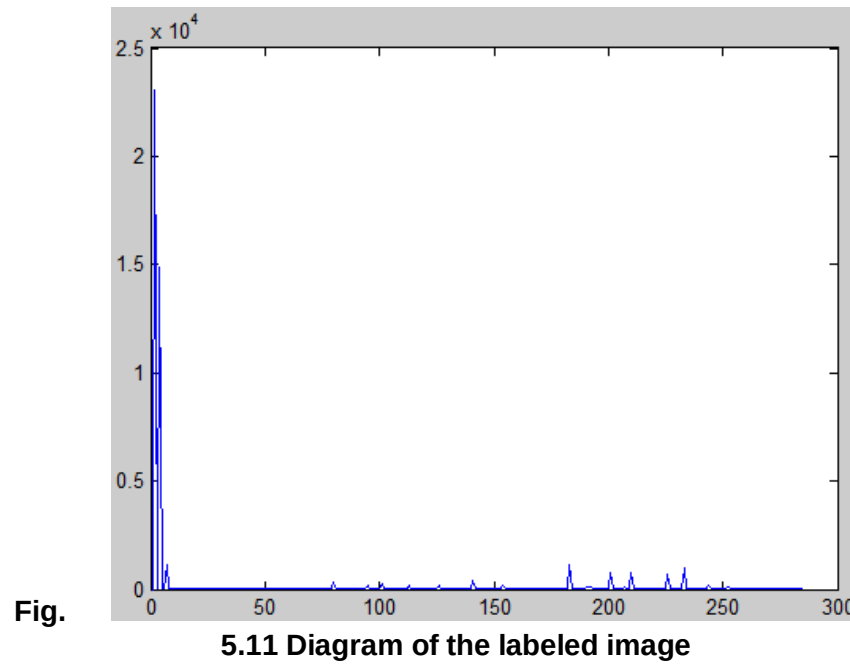
5.2.5 Experiment 1



Fig. 5.9 Untouched image



Fig. 5.10 Labeled image



Region Label Value	Size of the Region	Aspect Ratio of the Region
6	1123	0.729
182	1162	0.961

Table 5.3 The region's parameters from the labeled image



Fig. 5.12 Generated Deduction

Hypothesis	No. of upper- half components	No. of Characters	Classification
1	1	0	Not a Traffic Sign
2	10	10	Traffic Sign

Table 5.4 Classifying the Hypothesized Regions

Hypothesis 1 is rejected in the classification step and hypothesis 2 is classified as a speed-limit sign and its lower half components are recognized using the trained neural networks. The recognized traffic signboard is displayed in the figure below.



Fig. 5.13 Determined traffic sign

5.2.6 Experiment 2



Fig. 5.14 initial image



Fig. 5.15 Classified image

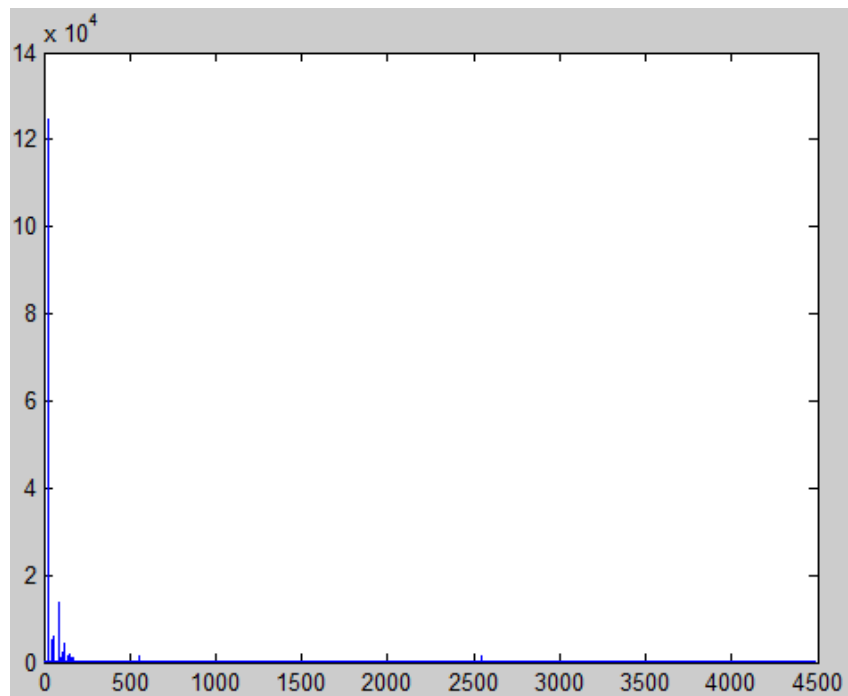


Fig. 5.16 Histogram of the termed image

Region label	Size of the Region	Aspect Ratio of the Region
1711	1272	0.771

Table 5.5 Region Criterion from the labeled image

By analyzing the region parameters from the labeled image, we obtained only one region that satisfies the two conditions.



Fig. 5.17 Generated Interpretation

No. of upper-half components	No. of recognized Characters	Classification
10	10	Traffic sign

Table 5.6 Analyzing the interpreted region

In the next step the lower half components of the region are recognized using the trained neural network and the recognized traffic sign is displayed as the sign board as shown in the figure below.

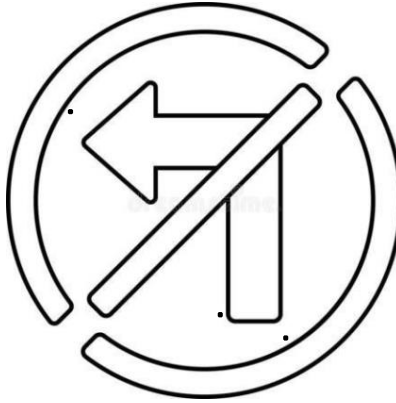


Fig. 5.18 Perceived traffic sign

5.2.7 Experiment 3



Fig. 5.19 Primary image



Fig. 5.20 Specified image

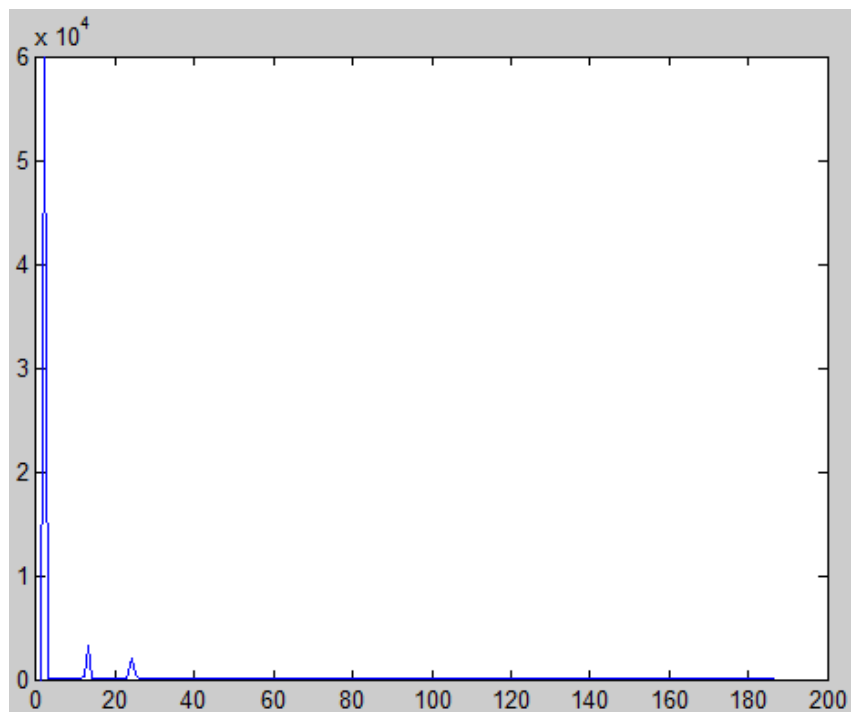


Fig. 5.21 Illustration of the labeled image

Region label	Size of the Region	Aspect Ratio of the Region
12	3302	0.887
23	1981	0.456

Table 5.7 Region Specification from the image 5.21

Using the region parameters calculated from the labeled image the possible traffic sign locations are shown in the figure below.



Fig. 5.22 Generated Inference

No. of upper-half components	No. of Recognized Characters	Classification
10	9	No Left Turn

Table 5.8 Segregating the hypothesized region in figure 5.22



Fig. 5.23 Developed Presumption

5.2.8 Experiment 4



Fig. 5.24 Aboriginal image



Fig. 5.25 Designated image

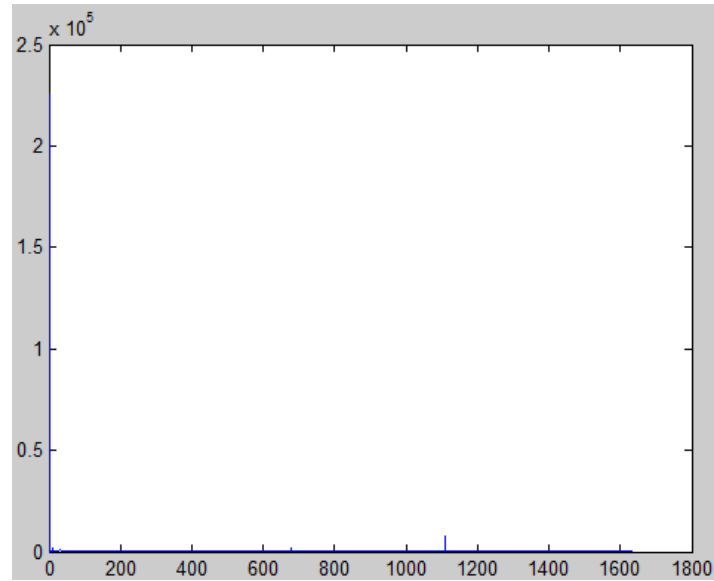


Fig. 5.26 Histogram of the entitled image

Region label	Size of the Region	Aspect Ratio of the Region
679	1594	0.8363
1108	7934	0.6012

Table 5.9 Region Framework from the labeled image



Fig. 5.27 Generated System

Hypothesis	No.of Components in the Region	No. of Characters in the Region	Classification
1	1	0	Not a Traffic Sign
2	1	7	Traffic sign

Table 5.10 Distinguishing the inferred regions in the figure.

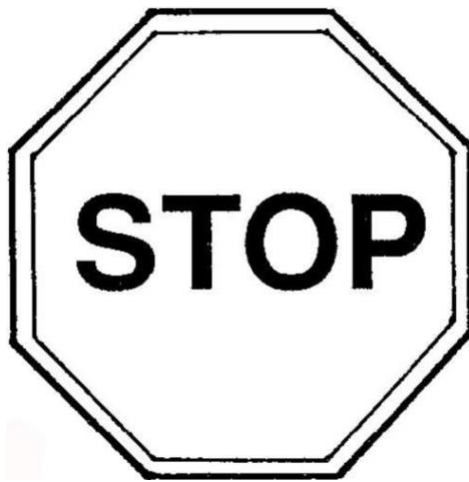


Fig. 5.28 Recognized sign board

IC-L298N allows parallel threading for the supply and direction of pulses from the Arduino Uno in conjunction with the driver. Deep learning is used for this process. A convolutional neural network (CNN) algorithm is employed, and a CNN sub-algorithm called VGG16 is utilized. The application table clearly outlines the various features of the motor driver IC when compared to other driver ICs. Code is uploaded into the Arduino and Arduino Motion driver is opened which is in close relation to the output of Arduino Uno.

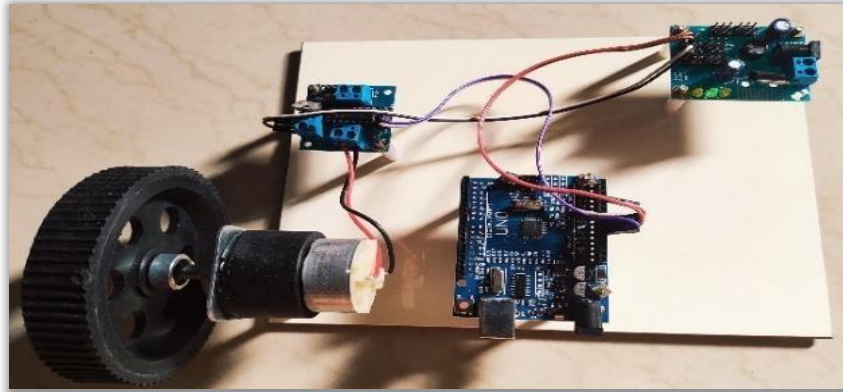


Fig. 5.29 Automatic traffic sign-based vehicle control system

From this point on, the motor driver IC controls the vibration motor, and in response, the camera records the vibration and analyzes it using pre-set techniques before processing it with an Arduino Uno. Arduino IR sensor reads the status of the plate and processes it.

When a traffic sign is given as input, due to the algorithm's simplicity and decreased processing demands, it converts the RGB input image to a Grayscale image. Gray scale also reduces an image to the fewest possible pixels, facilitating simple visualization. Then it will undergo detection and recognition phase.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

This final chapter concludes the project. Additionally, it provides a plan for how the project will be improved going forward. The references that were used to support the project are briefly listed at the end.

6.1 SYNOPSIS

This project presents a new method for automatically detecting the traffic signs positioned on either side of a road. We have modified the original connected component labeling technique to adapt to the grayscale traffic images. This forms the detection phase of our algorithm. In the next phase, called as recognition phase, the individual components of the regions are labeled, and their invariant features are calculated from Hu's moments. The components in the upper half of the region are first analyzed to identify the region. If the region contains the specific alphabets that are present on a traffic sign, then the region is classified as a traffic sign and processed further, otherwise, it is rejected. If the region is classified as a traffic sign, then its lower half components which specify the assigned traffic limit are identified using a feed forward neural network, trained with a backpropagation learning algorithm.

Traffic sign boards are often rotated, tilted, partially occluded, and are viewed at different angles. Under these noisy conditions, appearance-based features are more robust than shape-based features. Hence, we have utilized the region characteristics instead of the geometrical features. The proposed recognition method includes features from two different kinds of recognition methods; they decompose the region into its structural parts and use the invariant features for the recognition scheme. This allows for achieving more accurate results.

6.2 FUTURE SCOPE

In the future, a decrease in dimension of the feature vectors will be used to accelerate the method of identification. The connected component labeling algorithm requires more computations when applied on grayscale images, as it must label every pixel unlike in binary images. The number of computations can be probably reduced by defining the domain of operation for the algorithm. In the recognition phase, we have employed two different methods for classifying the region and extracting the information, although both do the same job. It is because the neural network is not able to reject unknown patterns. If the neural network is equipped with such a mechanism, then it would further simplify the method. The device suggested in this method would take the place of an automobile's manual activities. The main goal is to recognize and categorize the traffic signs from an input image while operating the car at the designated traffic sign.

REFERENCES

- [1]. Angela Tam, Hua Shen, Liu and Xiaou Tang (2003). Quadrilateral sign board detection and text extraction. CISST, pp. 708- 713.
- [2]. A.Broumandnia, M.Fathy (Dec. 2005). Application of pattern recognition for Farsi license plate recognition. ICGST International Conference on Graphics, Vision and Image Processing (GVIP-05).
- [3]. C. Anagnostopoulos, T. Alexandropoulos, S. Boutas, V. Loumos, E. Kayafas(2005). A template- guided approach to vehicle surveillance and access control. In proc.IEEE Conf. on Advanced Video and Signal Based Surveillance, pp. 534 – 539.
- [4]. C. Anagnostopoulos, E. Kayafas, V. Loumos (2000). Digital image processing and neural networks for vehicle license plate identification. Journal of Electrical Engineering, volume 1, No.2, pp. 2-7.
- [5]. David Llorens, Andres Marzal, Vicente Palazon, and Juan M. Vilar (2005). Car License Plates Extraction and Recognition Based on Connected Components Analysis and HMM Decoding. Lecture Notes on Computer Science 3522, J.S. Marques et al. , Ed. Springer-Verlag, pp. 571–578.
- [6]. F. Martín, M. García, J. L. Alba, (Jun. 2002). New Methods for Automatic Reading of VLP's (Vehicle License Plates). IASTED International Conference Signal Processing, Pattern Recognition, and Applications.
- [7]. F. Moutarde, A. Bargeton, A. Herbin and L. Chanussot (2007). Modular traffic signs recognition applied to on-vehicle-time visual detection of American and European speed limit signs. In IEEE Intelligent Vehicles Symposium Proc., pp.1122- 1126, Istanbul (Turkey).
- [8]. F. Zernike, Physica, Vol. 1. P. 689, 1934.
- [9]. Hsien-Chu WU, Chwei-Shyong TSAI, and Ching-Hao LAI (2004). A License Plate Recognition System in E-Government. Information and Security. An International Journal, Vol. 15, No. 2, pp. 199-210.

- [10]. Hua Shen and Xiaoou Tang (2003). Generic sign board detection in images. In MIR'03 Proceedings of the 5th ACM SIGMM International workshop on Multimedia information retrieval, pp. 144-149, New York, NY, USA.
- [11]. Juan J. Rodriguez, Cesar Garcia-osorio and Jesus Maudes (2008). License plate number recognition – New Heuristics and a Comparative Study of Classifiers. International Conference on Informatics in Control, Automation and Robotics – ICINCO, pp. 268-273.
- [12]. J. R. Parker and Pavol Federl (1995). An approach to license plate recognition. Laboratory for Computer Vision, Computer Graphics Laboratory, University of Calgary.
- [13]. Rosenfeld A., Pfaltz J.L. (1966). Sequential Operations in Digital Processing, Journal of ACM, Volume 13, pp. 471-494.
- [14]. Ullman, Shimon (2000). High-Level Vision: Object Recognition and Visual Cognition (1st ed.). A Bradford Book.
- [15]. J. Torresen, J. W. Bakke, and L. Sekanina (Oct. 2004). Efficient recognition of speed limit signs. In Intelligent Transportation Systems, Proceedings. The 7th International IEEE Conference on, pp. 652-656.
- [16]. Kehtarnavaz, N., and A. Ahmad (Sept. 1995). Traffic sign recognition in noisy outdoor scenes. In IEEE Intelligent Vehicles Symposium (IV), pp. 460-465, Detroit, MI, USA
- [17]. K.K.Kim K.I.Kim J.B.Kim H. J.Kim (2000). Learning-Based Approach, for License Plate Recognition. In proc. IEEE Signal Processing Society Workshop, Neural Networks for Signal Processing, Volume 2, pp. 614 - 623.
- [18]. L. Dreschler and H. Nagel (1981). Volumetric model and 3-D trajectory of a moving car derived from monocular TV- frame sequence of a street scene. In *Proc. IJCAI*, pp. 692–697.

- [19]. Lorsakul, A., and Suthakorn, J. (Nov. 2007). Traffic Sign Recognition for Intelligent Vehicle/Driver Assistance System Using Neural Network on OpenCV. Proceedings of the 4th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), POSTECH, PIRO, KOREA, pp. 279 – 284.
- [20]. L. Kitchen and A. Rosenfeld (Dec. 1982). Gray-level corner detection. Pattern Recognition Letters, volume 1, issue 2, pp. 95–102, New York, NY, USA.
- [21]. L. Priese, V. Rehrmann, R. Schian, and R. Lakmann (1993). Traffic sign recognition based on color image evaluation. In IEEE Intelligent Vehicles Symposium (IV), pp. 95- 100.
- [22]. M. Betke and N. Makris (1994). Fast object recognition in noisy images using simulated annealing. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA.
- [23]. M. K. Hu (1962). Visual pattern recognition by moment invariants. IRE Trans. Information Theory, volume 8, issue 2, pp. 179-187.
- [24]. M. Shah and R. Jain (Dec 1984). Detecting time-varying corners. Computer Vision, Graphics and Image Processing, vol. 28, no. 3, pp. 345–355.
- [25]. Nick Barnes (2005). Improved signal to noise ratio and computational speed for gradient-based detection algorithms. Robotics and Automation. ICRA, Proceedings of the 2005 IEEE International Conference on, pp. 4661-4666, Australia.
- [26]. N. Barnes and A. Zelinsky (2004). Real-time radial symmetry for speed sign. Detection. In *IEEE Intelligent Vehicles Symposium (IV)*, pp. 566–571, Parma, Italy.
- [27]. N. Barnes and G. Loy (2004). Fast shape-based road sign detection for a driver assistance system. Intelligent Robots and systems, pp. 70-75, vol.1, Stockholm, Sweden.
- [28]. O. A. Zuniga and R. Haralik (1983). Corner detection using the facet model. In *Proc. IEEE CVPR Conference*, pp. 30–37.

- [29]. P. Comelli, P. Ferragina, M.N. Granieri, F. Stabile (1995). Optical recognition of motor vehicle license plates. *IEEE Trans. On Vehicular Technology*, volume 44, No. 4, pp. 790-799.
- [30]. Shyang-Lih Chang, Li-Shien Chen, Yun-Chung Chung, and Sei-Wan Chen (2004). Automatic License Plate Recognition. *IEEE Trans. on Intelligent Transportation Systems*, Volume 5, No. 1, pp. 42-53.
- [31]. Tran Duc Duan, Tran Le Hong Du, Tran Vinh Phuoc, Nguyen Viet Hoang(2005). Building an Automatic Vehicle License-Plate Recognition System. In *proc. Intl.Conf. in Computer Science (RIVF)*, pp. 59-63.
- [32]. X. Gao, L. Podladchikova, D. Shaposhnikov, K. Hong, and N. Shevtsova (August 2006). Recognition of traffic signs based on their color and shape features extracted using human vision models. *Journal of Visual Communication and Image Representation*, volume 17, issue 4, pp. 675-685, Rostov-on-Don, Russia.
- [33]. Yafeng Hu, Feng Zhu, and Xianda Zhang (2005). A Novel Approach for License Plate Recognition Using Subspace Projection and Probabilistic Neural Network. *Lecture Notes on Computer Science 3497*, J. Wang, X. Liao, and Z. Yi, Ed. Springer- Verlag, LNCS 3497, pp. 216–221.
- [34]. Yo-Ping Huang, Shi- Yong Lai and Wei-Po Chuang (2004). A Template-Based Model for License Plate Recognition. In *proc. IEEE Int. Conf. on Networking, Sensing &Control*, pp. 737-742.

