TRAFFIC-SIGN DETECTION AND RECOGNITION USING DEEP LEARNING

Submitted in partial fulfillment of the requirements for the award of Bachelorof Engineering degree in Computer Science and Engineering

by

G.sai jithin, (Reg. No. 38110164)

Yanamala Umesh (Reg. No.38110662)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY - 2022

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

*DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**BONAFIDE***

CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KORAPATI MAHESH (Reg. No: 38110288), SILUVERU HAKHILESH RAM (Reg. No: 38110189)** who

carried out the project entitled "**TRAFFIC-SIGN DETECTION AND RECOGNITION USING DEEP LEARNING**

" under my supervision from November 2021 to April 2022.

Internal Guide **External Guide (if Applicable)**

Dr. G. Nagarajan M.E., Ph.D.,

ad of the Department

Dr. L. LAKSHMANAN M.E., Ph.D.,

Submitted for Viva voce Examination held on ______________________

Internal Examiner **External Examiner**

DECLARATION

I G.sai jithin, Yanamala Umesh hereby declare that the Project Report entitled **TRAFFIC-SIGN DETECTION AND RECOGNITION USING DEEP LEARNING** done by me under the guidance of **Dr. G. Nagarajan M.E., Ph.D., (Internal)** is submitted in partial fulfillment of the requirements forthe award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE: 25-03-2022

K. Mahesh.

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. T. Sasikala M.E., Ph. D, Dean, School of Computing Dr. L. Lakshmanan M.E., Ph.D., and Dr. S. Vigneshwari M.E., Ph.D. Head of the Department, Dept. of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr. G. Nagarajan ME., Ph. D for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Computer Science and Engineering who were helpful in many ways for the completion of the project.

TABLE OF CONTENTS

LIST OF FIGURES

TRAFFIC-SIGN DETECTION AND RECOGNITION USING DEEP LEARNING

Abstract

Automatic detection and recognition of traffic signs plays a crucial role in management of the traffic-sign inventory. It provides an accurate and timely way to manage traffic-sign inventory with a minimal human effort. In the computer vision community, the recognition and detection of traffic signs are a well-researched problem. A vast majority of existing approaches perform well on traffic signs needed for advanced driver-assistance and autonomous systems. However, this represents a relatively small number of all traffic signs (around 50 categories out of several hundred) and performance on the remaining set of traffic signs, which are required to eliminate the manual labor in traffic-sign inventory management, remains an open question. In this paper, we address the issue of detecting and recognizing a large number of traffic-sign categories suitable for automating traffic-sign inventory management. We adopt a convolutional neural network (CNN) approach, the mask R-CNN, to address the full pipeline of detection and recognition with automatic end-to-end learning. We propose several improvements that are evaluated on the detection of traffic signs and result in an improved overall performance. This approach is applied to detection of 200 traffic-sign categories represented in our novel dataset. The results are reported on highly challenging traffic-sign categories that have not yet been considered in previous works. We provide comprehensive analysis of the deep learning method for the detection of traffic signs with a large

intra-category appearance variation and show below 3% error rates with the proposed approach, which is sufficient for deployment in practical applications of the traffic-sign inventory management.

CHAPTER 1

INTRODUCTION

Proper management of traffic-sign inventory is an important task in ensuring safety and efficiency of the traffic flow. Most often this task is performed manually. Traffic signs are captured using a vehicle-mounted camera and manual localization and recognition is performed off-line by a human operator to check for consistency with the existing database. However, such manual work can be extremely time consuming when applied to thousands of kilometers of roads. Automating this task would significantly reduce the amount of manual work and improve safety through quicker detection of damaged or missing traffic signs. A crucial step towards the automation of this task is replacing manual localization and recognition of traffic signs with an automatic detection. In the computer-vision community the problem of traffic-sign recognition has already received a considerable attention, and excellent detection and recognition algorithms have already been proposed. But these solutions have been designed only for a small number of categories, mostly for traffic signs associated with advanced driver-assistance systems (ADAS) and autonomous vehicles. Detection and recognition of a large number of traffic-sign categories remains an open question. Various previous benchmarks have addressed the traffic-sign recognition and detection task. However, several of them focused only on traffic-sign recognition (TSR) and ignored the much more complex problem of traffic-sign detection (TSD) where finding accurate location of traffic sign is needed. Other benchmarks that do address TSD mostly cover only a subset of traffic-sign categories, most often ones important for ADAS and autonomous vehicles applications. Most categories appearing in such benchmarks have a distinct appearance with low inter-category variance and can be detected using handcrafted detectors and classifiers. Such examples include round mandatory signs or triangular prohibitory signs.

1.1 PROBLEM DEFINITION

Automatic detection and recognition of traffic signs plays a crucial role in management of the traffic-sign inventory. The issue of detecting and recognizing a large number of traffic-sign categories suitable for automating traffic-sign inventory management. We adopt a convolutional neural network (CNN) approach, the mask R-CNN, to address the full pipeline of detection and recognition with automatic end-to-end learning. We propose several improvements that are evaluated on the detection of traffic signs and result in an improved overall performance.

CHAPTER 2

LITERATURE SURVEY

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, then the next step is to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system. The major part of the project development sector considers and fully survey all the required needs for developing the project. For every project Literature survey is the most important sector in software development process. Before developing the tools and the associated designing it is necessary to determine and survey the time factor, resource requirement, man power, economy, and company strength. Once these things are satisfied and fully surveyed, then the next step is to determine about the software specifications in the respective system such as what type of operating system the project would require, and what are all the necessary software are needed to proceed with the next step such as developing the tools, and the associated operations.

[2.1] A novel lightweight CNN architecture for traffic sign recognition without GPU requirements

For a safe and automated vehicle driving application, it is a prerequisite to have a robust and highly accurate traffic sign detection system. In this paper, we proposed a novel energy-efficient Thin yet Deep convolutional neural network architecture for traffic sign recognition. Within the proposed architecture, each convolutional layer contains less than 50 features enabling our convolutional neural network to be trained quickly even without the aid of a graphics processing unit. The performance of the proposed architecture is measured using two publicly available traffic sign datasets, namely the German Traffic Sign Recognition Benchmark and the Belgian Traffic Sign Classification dataset. First, we train and test the performance of the proposed architecture using the large German Traffic Sign Recognition Benchmark dataset. Then, we retrain the network models using transfer learning on the more challenging Belgian Traffic Sign Classification dataset to evaluate test performance. The proposed architecture outperforms the performance of the state-of-the-art traffic sign methods with at least five times less parameter in the individual end-to-end network for training.

[2.2]An efficient convolutional neural network for small traffic sign detection

Deep learning has become a ubiquitous method in object detection among multiple domains recently. However, in the era of edge computing, deploying deep neural networks on mobile edge platforms are challenging due to long latency and huge computational cost. As previous research efforts were usually focused on accuracy, achieving the balance between computational consumption and accuracy is a more significant problem to be tackled in mobile edge computing domain. To this end, we proposed an efficient convolutional neural network (CNN), which can remarkably minimize the redundancy, reduce the parameters and speed up the networks. The effectiveness of the network is further proved with experiments on a Tsinghua-Tencent 100K traffic sign dataset. Results show that under the same-level model size, our network outperforms the state-of-the-art Fast R-CNN and Faster R-CNN with 10% improvement in accuracy. Compared to similar work, the computational consumption on running time and memory of our network has been also reduced in the premise of little loss in accuracy.

[2.3]Traffic Sign Detection and Recognition using a CNN Ensemble

In today's world, almost everything we do has been simplified by automated tasks. In an attempt to focus on the road while driving, drivers often miss out on signs on the side of the road, which could be dangerous for them and for the people around them. This problem can be avoided if there was an efficient way to notify the driver without having them to shift their focus. Traffic Sign Detection and Recognition (TSDR) plays an important role here by detecting and recognizing a sign, thus notifying the driver of any upcoming signs. This not only ensures road safety, but also allows the driver to be at little more ease while driving on tricky or new roads. Another commonly faced problem is not being able to understand the meaning of the sign. With the help of this Advanced Driver Assistance Systems (ADAS) application, drivers will no longer face the problem of understanding what the sign says. In this paper, we propose a method for Traffic Sign Detection and Recognition using image processing for the detection of a sign and an ensemble of Convolutional Neural Networks (CNN) for the recognition of the sign. CNNs have a high recognition rate, thus making it desirable to use for implementing various computer vision tasks. TensorFlow is used for the implementation of the CNN. We have achieved higher than 99% recognition accuracies for circular signs on the Belgium and German data sets.

[2.4]The Speed Limit Road Signs Recognition Using Hough Transformation and Multi-Class Svm

In this paper, a method for the speed limit traffic sign recognition is proposed. The method is based on Support Vector Machines, which is one of the most efficient algorithms used for traffic sign recognition. It comprises three phases. In the preprocessing phase, RGB images are converted into HSL images in order to increase the contrast. In the detection phase, Hough Transformation is used for detecting the speed limit signs along with Gauss and Median filters for removing the noise from the detected images. The detection phase achieves accuracy of 95.3%. In the classification phase, a Histogram of Oriented Gradients descriptor for feature extraction is used together with Support Vector Machines for image classification and speed limit sign recognition. The proposed method was used on the two databases - GTSRB, German Traffic Sign Recognition Benchmark and rMASTIF, Croatian traffic sign database. The recognition accuracy of 93.75% is achieved. The presented method proves to be applicable in advancing driving assistance systems due to its detection and recognition accuracy as well as its performance, thus making it appropriate for real-time applications.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 HARDWARE REQUIREMENTS

System	: Pentium i3 Processor
Hard Disk	: 500 GB.
Monitor	: 15’’ LED
Input Devices	: Keyboard, Mouse
Ram	: 2 GB

3.2 SOFTWARE REQUIREMENTS

Operating system	: Windows 10
Coding Language	: Python

CHAPTER 4

SYSTEM ANALYSIS

4.1 PURPOSE

The purpose of this document is detection of traffic sign using machine learning algorithms. In detail, this document will provide a general description of our project, including user requirements, product perspective, and overview of requirements, general constraints. In addition, it will also provide the specific requirements and functionality needed for this project - such as interface, functional requirements and performance requirements.

4.2 SCOPE

The scope of this SRSdocument persists for the entire life cycle of the project. This document defines the final state of the software requirements agreed upon by the customers and designers. Finally at the end of the project execution all the functionalities may be traceable from the SRSto the product. The document describes the functionality, performance, constraints, interface and reliability for the entire life cycle of the project.

4.3 EXISTING SYSTEM

Yuan et al. in which though the accuracy rate was high, the processing time was also very high. Another method used fusion network formation to obtain features of the signs and background statistics around the observed image, but the complexity was high.

4.4 Disadvantages of exsisting system

- Increased algorithms complexity

- Heavy system hardware requirement
- Different pre-processing for training data are necessary

4.5 PROPOSED SYSTEM

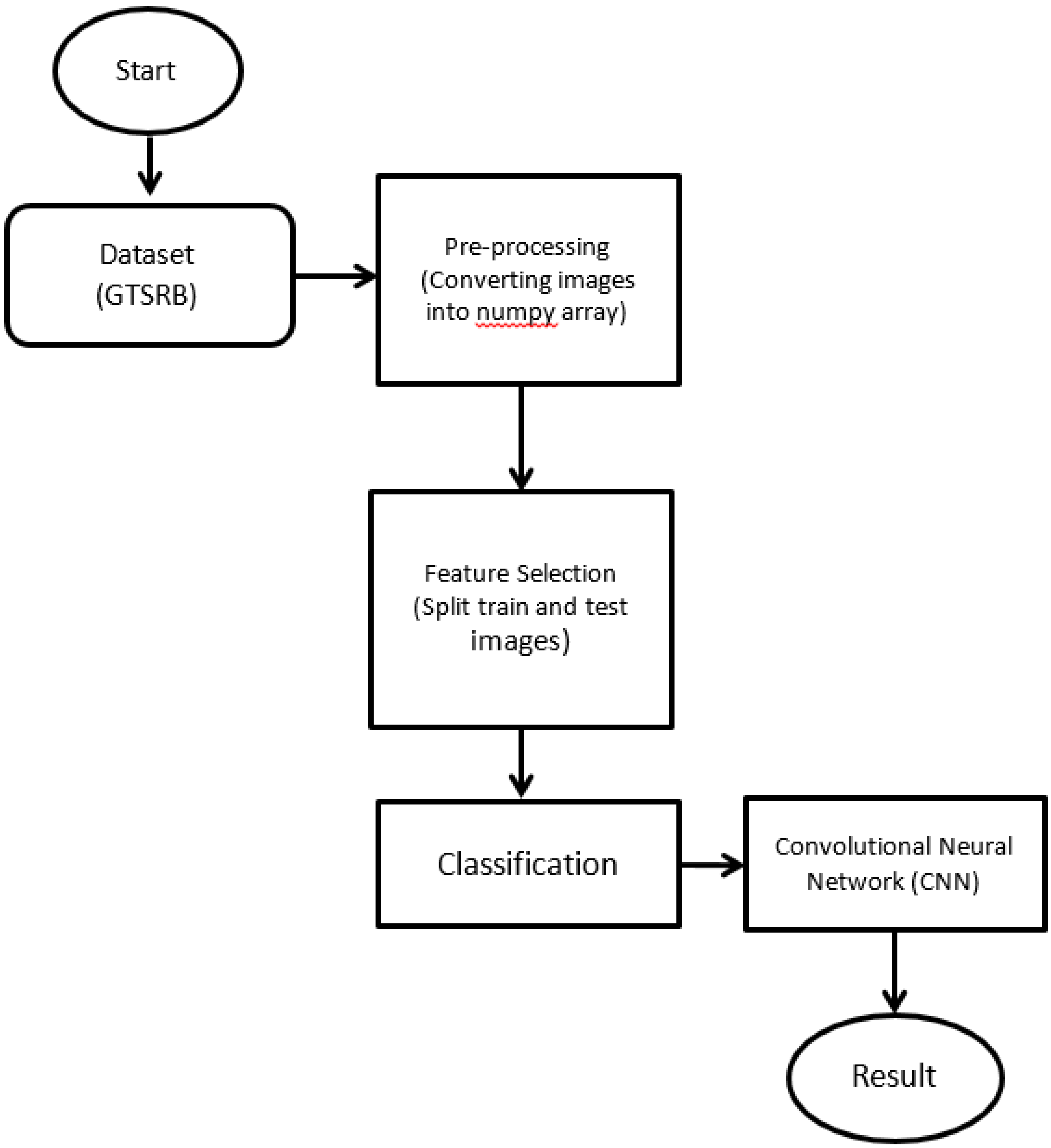
Traffic sign recognition and detection is an important part of any autonomous vehicle. However, the real challenge lies in the detection and recognition of these traffic sign from the natural image in real time and with accuracy. This paper gives an overview of the traffic road sign detection and recognition system, we developed and implemented using an artificial neural network which is trained using real-life datasets. This paper presents the usage of convolution neural network along with dataset as an implementation of our project to attain real-time result with accuracy. The system developed based on this methodology can be implemented in public transports, personal cars, and other vehicles in order to keep drivers alert and reduce human errors that lead to accidents. The project has a wide implementation of self-driving vehicles.

4.6 ADVANTAGES OF PROPOSED SYSTEM

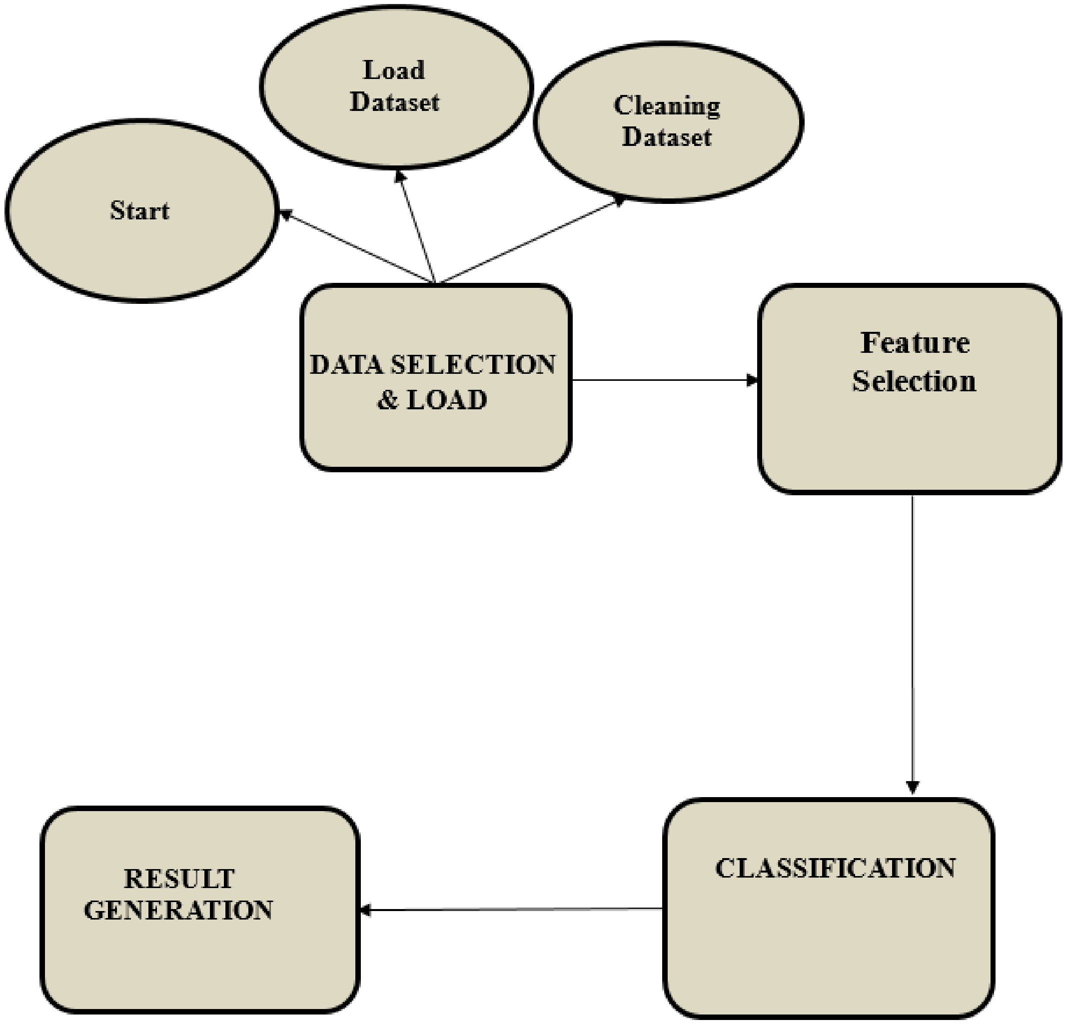
- Reducing the number of accidents caused by driver distraction and to reduce the seriousness of such accidents.
- Improve the driver's safety on the road.

DATA FLOW DIAGRAM

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.


Start
Dataset
(GTSRB)
Pre-processing
(Converting images
into numpy array)
Feature Selection
(Split train and test
images)
Classification
Convolutional Neural
Network (CNN)
Result

ER DIAGRAM



UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

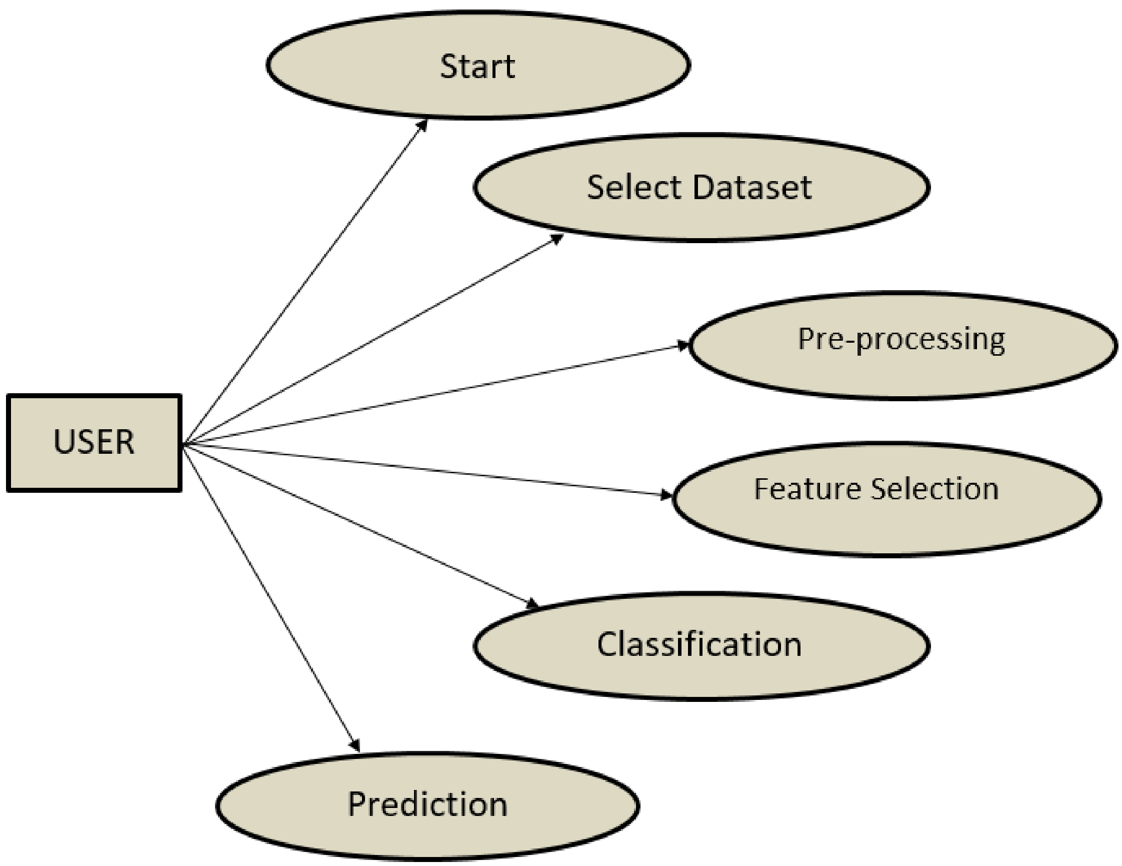
GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

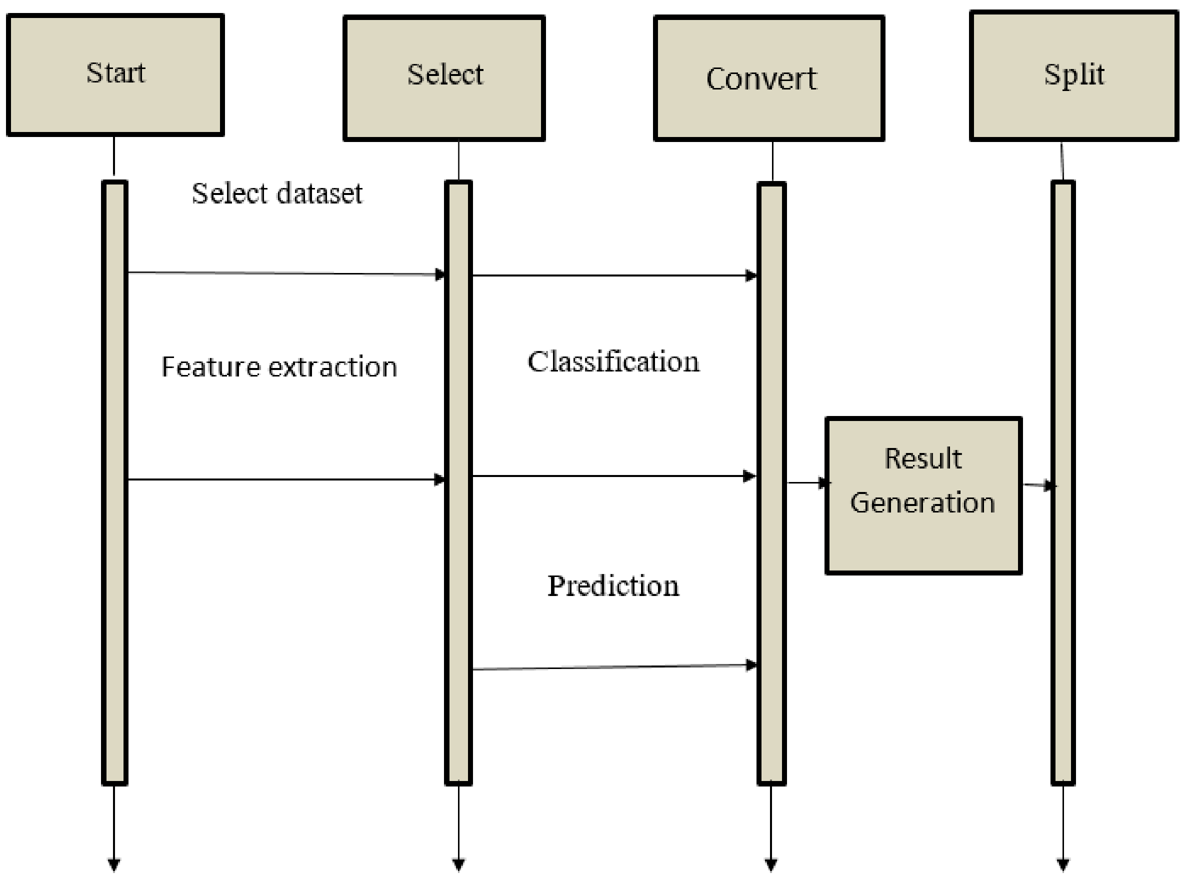
USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



SEQUENCE DIAGRAM:

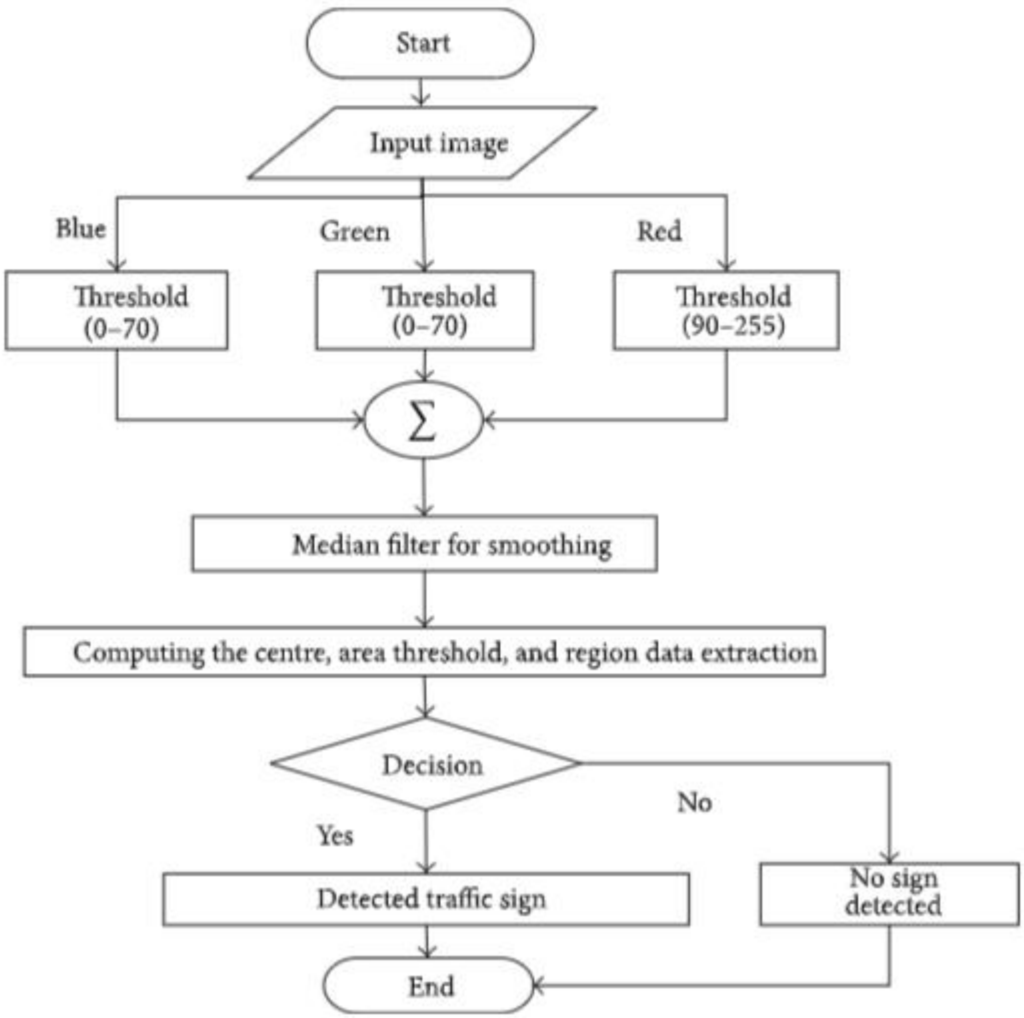
A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified

Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



CHAPTER 6

MODULES DESCRIPTION

6.1 MODULES

- Segmentation Module
- Data Pre-processing
- Traffic Sign-Recognition
- Traffic Sign-Detection

MODULE DESCRIPTION

6.1.1 Segmentation Module

Segmentation is the method of partitioning a visual image into different subgroups (of pixels) called Image Objects, which reduces the image's complexity and makes image analysis easier. Thresholding is the method of using an optimal threshold to transform a grayscale input image to a bi-level image.

6.1.2 Data Pre-Processing Module

To save space or reduce computing complexity, we can find it helpful to remove redundant details from images in some situations. .Converting colorful images to grayscale images, for example. This is because color isn't always used to identify and perceive an image in several objects. Grayscale may be sufficient for identifying such artefacts. Color images can add needless complexity and take up more memory space because they hold more detail than black and white images color images are represented in three channels, which means that converting it to grayscale reduces the number of pixels that need to be processed. For traffic signs gray values are sufficient for recognition.

6.1.3 Traffic Sign-recognition Module

Deep Learning is a subdomain of Machine Learning that includes Convolutional Neural Networks. Deep Learning algorithms store information in the same manner as the human brain does, but on a much smaller scale .Image classification entails extracting features from an image in order to identify trends in a dataset. We are using CNN for traffic sign recognition as it is very good at feature extraction. In CNN, we use filters. Filters come in a variety of shapes and sizes, depending on their intended use. Filters allow us to take advantage of a specific image's spatial localization by imposing a local communication pattern between neurons. Convolution is the process of multiplying two variables pointwise to create a new feature. Our image pixels matrix is one function and our filter is another. The dot product of the two matrices is obtained by sliding the filter over the image. Matrix called "Activation Map" or "Feature Map". The output layer is made up of several convolutional layers that extract features from the image. CNN can be optimized with the help of hyper parameter optimization. It finds hyper parameters of a given machine learning algorithm that deliver the best performance as measured on a validation set. Hyper parameters must be set before the learning process can begin. The learning rate and the number of units in a dense layer are provided by it. In our system will consider dropout rate, learning rate, kernel size and optimizer hyper parameter.

6.1.4 Traffic Sign Detection Module

In this Module, we have addressed the problem of detecting and recognizing a large number of traffic-sign categories for the main purpose of automating traffic-sign inventory management. Due to a large number of categories with small interclass but high intra-class variability, we proposed detection and recognition utilizing an approach based on the Mask RCNN detector. The system provides an efficient deep network for learning a large number of categories with an efficient and fast detection.

CHAPTER 7

SYSTEM IMPLEMENTATION

7.1 SYSTEM ARCHITECTURE

Describing the overall features of the software is concerned with defining the requirements and establishing the high level of the system. During architectural design, the various web pages and their interconnections are identified and designed. The major software components are identified and decomposed into processing modules and conceptual data structures and the interconnections among the modules are identified. The following modules are identified in the proposed system.

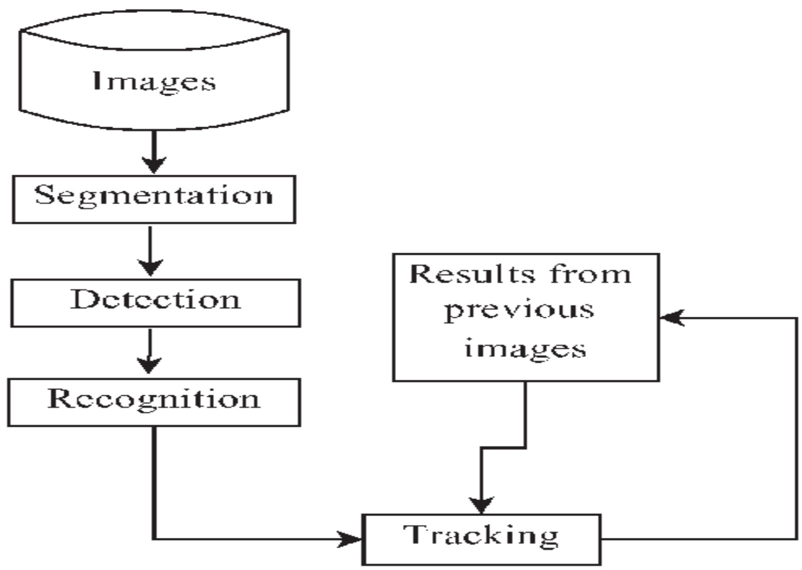


Fig: System Architecture

The above architecture describes the work structure of the system.

- The data is collected and image processing is done after pre-processing.
- After completing the pre-processing with the collected data and then by applying machine learning algorithms it detects and recognizes the traffic signal.

7.2 Problem Statement:

Automatic detection and recognition of traffic signs plays a crucial role in management of the traffic-sign inventory. The issue of detecting and recognizing a large number of traffic-sign categories suitable for automating traffic-sign inventory management. We adopt a convolutional neural network (CNN) approach, the mask R-CNN, to address the full pipeline of detection and recognition with automatic end-to-end learning. We propose several improvements that are evaluated on the detection of traffic signs and result in an improved overall performance.

CONCLUSION

In this paper, we have discussed that how our proposed system detects the traffic signal and recognizes using machine learning algorithms. The proposed system is also scalable for detecting and recognizing the traffic sign by image processing. The system is not having complex process to detect and recognize that the data like the existing system. Proposed system gives genuine and fast result than existing system. Here in this system we use cnn algorithm to detect and recognize the traffic sign.

REFERENCES

[1] Aziz S, Mohamed E, Youssef F (2018) Traffic sign recognition based on multi-feature fusion and ELM classifier. Proc Comput Sci 127:146–153.

[2] Jang, C., Kim, H., Park, E., Kim, H. (2016). Data debiased traffic sign recognition using MSERs and CNN. In 2016 International Conference on Electronics, Information, and Communications (ICEIC), Da Nang, Vietnam, pp. 1-4. https://doi.org/10.1109/ELINFOCOM.2016.7562938.

[3] Lai, Y., Wang, N., Yang, Y., & Lin, L. (2018). Traffic signs recognition and classification based on deep feature learning. In 7th International Conference on Pattern Recognition Applications and Methods (ICPRAM), Madeira, Portugal (pp. 622-629).

[4] Rosario G, Sonderman T, Zhu X. (2018) Deep Transfer Learning for Traffic Sign Recognition[C]//2018IEEE International Conference on Information Reuse and Integration (IRI). IEEE: 178–185. MLA.

[5] Hatolkar, Y., Agarwal, P., & Patil, S. (2018). A Survey on Road Traffic Sign Recognition System using Convolution Neural Network.

[6] Huang, Z., Yu, Y., Gu, J., & Liu, H. (2017). An efficient method for traffic sign recognition based on extreme learning machine. IEEE transactions on cybernetics, 47(4), 920-933.

[7] Li, C., Hu, Y., Xiao, L., Tian, L. (2012). Salient traffic sign recognition based on sparse representation of visual perception. In 2012 International Conference on Computer Vision in Remote Sensing, Xiamen, China, pp. 273-278. https://doi.org/10.1109/CVRS.2012.6421274.

[8] Shi W, Xin L, Yu Z et al (2017) An FPGA-based hardware accelerator for traffic sign detection. IEEETransactions on Very Large Scale Integration Systems 4:1362–1372.

[9] Rosario G, Sonderman T, Zhu X. (2018) Deep Transfer Learning for Traffic signal Recognition[C] 2018 IEEE International Conference on Information Reuse and Integration.

[10] Li, H., Gong, M. (2017). Self-paced Convolutional Neural Networks. In IJCAI, pp. 2110-2116.