TRAFFIC SIGN RECOGNITION

An SEPM PROJECT Report
Submitted To

Chhattisgarh Swami Vivekanand Technical University
Bhilai, India
For
The Award of Degree of
Bachelor of
Technology *in*
Computer Science & Engineering
By

DIVYAANSH SHARMA
Roll No.- 303302219035
En. No.- BH3718
Semester 6th (CSE)

SRIJAN SHARMA
Roll No.- 303302219106
En. No.- BH3789
Semester 6th (CSE)

SARTHAK CHILAMWAR
Roll No.- 303302219088
En. No.- BH3771
Semester 6th (CSE)

Under the Guidance of
Mr. ASHISH PANDEY
Assistant Professor
Department of Computer Science & Engineering
S.S.I.P.M.T, Raipur

Department of Computer Science & Engineering
Shri Shankaracharya Institute of Professional Management & Technology Raipur (C.G.)

SESSION 2021-22

DECLARATION BY THE CANDIDATE

We the undersigned solemnly declare that the report of the SEPM Project work entitled "**TRAFFIC SIGN RECOGNITON**", is based on our own work carried out during the course of our study under the supervision of Mr. Ashish Pandey.

We assert that the statements made, and conclusions drawn are an outcome of the project work. We further declare that to the best of our knowledge and belief that the report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University/deemed University of India or any other country.

(Signature of Candidate 1)	(Signature of Candidate 2)	(Signature of Candidate 3)
Name: DIVYAANSH SHARMA	**Name: SRIJAN SHARMA**	**Name: SARTHAK CHILAMWAR**
Roll No.: 303302219035	Roll No.: 303302219106	Roll No.: 303302219088
En. No.: BH3718	En. No.: BH3789	En. No.: BH3771
Semester: 6^{th} (CSE)	Semester: 6^{th} (CSE)	Semester: 6^{th} (CSE)

Department of Computer Science & Engineering

Shri Shankaracharya Institute of Professional Management & Technology

Raipur (C.G.)

CERTIFICATE OF THE SUPERVISOR

This is to certify that the SEPM project report of entitled "**TRAFFIC SIGN RECOGNITION**" is a record of bonafide research work/project work carried out by

1. Divyaansh Sharma(Roll No.: 303302219035 & Enrollment No.: BH3718)
2. Srijan Sharma (Roll No.: 303302219106 & Enrollment No.: BH3789)
3. Sarthak Chilamwar(Roll No.: 303302219088 & Enrollment No.: BH3771)

under my guidance and supervision for the award of Degree of Bachelor of Technology of Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G.), India.

To the best of my knowledge and belief the thesis embodies the work of the candidate themselves, has duly been completed, fulfils the requirement of the ordinance relating to the B. Tech degree of the University. Is up to the desired standard both in respect of contents and language for being referred to the examiners.

(Signature of HOD)
Dr. J.P. Patra
Dept. of Computer Science & Engineering
S.S.I.P.M.T, Raipur, (C.G.)

(Signature of the Supervisor)
Mr. Ashish Pandey
Assistant Professor, Dept of C.S.E
S.S.I.P.M.T, Raipur (C.G.)

Forwarded to

Chhattisgarh Swami Vivekanand Technical University, Bhilai

(Signature of the Principal)
Dr. Alok Kumar Jain
S.S.I.P.M.T, Raipur,
(C.G.)

Department of Computer Science & Engineering

Shri Shankaracharya Institute of Professional Management & Technology

Raipur (C.G.)

CERTIFICATE BY THE EXAMINERS

The Thesis/Project work entitled "**TRAFFIC SIGN RECOGNITON**" Submitted by

1. Divyaansh Sharma (Roll No.: 303302219035 & Enrollment No.: BH3718)
2. Srijan Sharma (Roll No.: 303302219106 & Enrollment No.: BH3789)
3. Sarthak Chilmawar (Roll No.: 303302219088 & Enrollment No.: BH3771)

has been examined by the undersigned as a part of the examination and is hereby recommended for the award of the degree of Bachelor of Technology of Chhattisgarh Swami Vivekanand Technical University, Bhilai.

Internal Examiner	**External Examiner**
Date :	**Date :**

ACKNOWLEDGEMENT

We would like to express our sincere thanks to Mr. Ashish Pandey, for her valuable guidance and support in completing our project.

We would also like to express our gratitude towards our respected Dr. J P Patra (Prof. & Head of Department) for giving us this great opportunity to do a project on Handwritten Character Recognition. Without their support and suggestions, this project would not have been completed.

We also pay special thanks for his helpful solution and comments enriched by his experience, which improved our ideas for betterment of the project. We would also like to express our deep gratitude to respected Dr. Alok Kumar Jain (Principal) and college management for providing an educational ambience.

It will be our pleasure to acknowledge, utmost cooperation and valuable suggestions from time to time given by our staff members of our department, to whom we owe our entire computer knowledge and also, we would like to thank all those people who have directly or indirectly helped us by providing books and computer peripherals and other necessary amenities which helped us in the development of this project which would otherwise have not been possible.

(Signature of Candidate 1)	(Signature of Candidate 2)	(Signature of Candidate 3)
Name: DIVYAANSH SHARMA	**Name: SRIJAN SHARMA**	**Name: SARTHAK CHILAMWAR**
Roll No.: 303302219035	Roll No.: 303302219106	Roll No.: 303302219088
En. No.: BH3718	En. No.: BH3789	En. No.: BH3771
Semester: 6th (CSE)	Semester: 6th (CSE)	Semester: 6th (CSE)

- **LIST OF SYMBOLS**

,	Comma
.	Full Stop
‘	Inverted comma
( )	Parenthesis
:	Colon
-	Hyphen
“ ”	Double inverted comma
[ ]	Angle bracket
&	Ampersand
/	Slash
*	Asterisk
;	Semi Colon

- **LIST OF ABBREVIATIONS**

EDA	Exploratory Data Analysis
CNN	Convolutional Neural Network
ML	Machine Learning
DFD	Data Flow Diagram
VS	Visual Studio
SDLC	Software Development Life Cycle
GUI	Graphical User Interface
MLP	Multilayer Perceptron
NN	Neural Network
TSR	Traffic Sign Recogoniton
SRS	Software Requirement Specification

- **LIST OF FIGURES**

- **TABLE OF CONTENT**

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Traffic Sign Recognition (TSR) is a technology by which a vehicle is able to recognize the traffic signs on the road e.g. no entry, traffic signals,speed limits, turn left or right etc. TSR represents an important feature of advanced driver assistance system, contributing to the safety of the drivers, autonomous vehicles as well and to increase driving comfort. In today's world road conditions drastically improved as compared with past decades.

In this project, a deep learning based road traffic signs recognition Method is developed which is very promising in the development of Advanced Driver Assistance Systems (ADAS) and autonomous vehicles.

The system architecture is designed to extract main features from images of traffic signs to classify them under different categories. The presented method uses a modified LeNet-5 network to extract a deep representation of traffic signs to perform the recognition. It is constituted of a Convolutional Neural Network (CNN) modified by connecting the output of all convolutional layers to the Multilayer Perceptron (MLP). The training is conducted using the German Traffic Sign Dataset of 43 different classes of traffic signs containing 50,000 images along with TensorFlow library.

The results will show the 94% accuracy.

The project is divided into two sections:

1. Training part
2. Testing part

The aim behind the **training** component is like instructing a child by presenting them with numerous sets of characters that are somewhat similar but not same and telling them that the result of this is "this." To train the freshly constructed neural network with so many characters, one must use this concept.

A fresh dataset is tested in the **testing** section. This section is always followed by the training section. The youngster must first be taught to recognize the character. The next step is to take the test to determine whether or not the response was correct. If not, you must train him more rigorously by providing additional datasets and entries. Similar to that, the algorithm must also be tested.

CHAPTER 2

LITERATURE REVIEW
&
PROBLEM IDENTIFICATION

2.1 LITERATURE REVIEW

Since the appearance of the first paper in Japan in 1984, road sign recognition has become one of the important research fields. From that time until the present day many research groups have been active in the field and have tried to solve this problem using different approaches. Although initially the main steps towards a solution seem very well defined and straightforward, the details of the approaches used show that there are several alternatives and many ideas as to how better solutions, better robustness, or a better classification rate can be achieved. So far, no one solution method has dominated, and it will clearly take some time before systems are seen in the market. The identification of road signs can be carried out by two main stages: detection, and recognition

2.1.1 Colour-Based Detection of Traffic Signs

Techniques invoked to carry out traffic sign detection varies from one author to another. There is a wide range of techniques used to solve this issue.

Ghica et al. An automated system for traffic sign recognition using convolutional neural network.", used thresholding to segment pixels in a digital image into object pixels and background pixels. The technique is based on calculating the distance in RGB space between the pixel colour and a reference colour. The unknown pixel is considered as an object pixel if it is close enough to the reference colour.[1]

Estevez and Kehtarnavas Study of Traffic Sign Detection and Recognition Algorithms, suggested an algorithm capable of recognizing the Stop, Yield, and Do-Not-Enter traffic warning signs. It consists of six modules: colour segmentation, edge localization, RGB differencing, edge detection, histogram extraction, and classification. Colour segmentation is only used to localise red edge areas; the segmentation is performed sparsely; and interpixel segmentation distance is determined.[2]

Yuille et al designed a sign finder system to help visually impaired people. The author assumed that signs consist of two colours (one for the sign, and another for the text), and sign boundaries are stereotyped (rectangle, hexagonal). Based on a set of tests to determine seeds, a region growing algorithm is used to detect hypothesis regions. o Yabuki et al. [29] proposed a method to detect the road sign by using the colour distribution of the sign in XYZ colour space. They constructed a colour similarity map from the colour distribution, which is then incorporated into the image function of an active net model. It is possible to extract the road sign when it is wrapped up in an active net.[3]

2.1.2 Shape-Based Detection of Traffic Signs

Techniques using shapes could be a good alternative when colours are missing or when it is hard to detect colours. Shape-based techniques should be able to avoid difficulties related to invoking colours for sign detection and robust to handle in-plane transformations such as translation, scaling and rotation. Much effort has been exerted to develop these techniques and the results are very promising. In the following reviewed papers, the authors used shapes as the major source of information to detect traffic signs:

Piccioli et a1. detected road signs by using a priori information of the supposed position of the sign in an image. A Canny edge detector was applied to the search region, and geometrical analysis was carried out on clusters of edge-points to extract the desired shape. The inner region of each candidate was tested against the database of signs by template matching. The correlation of the edge 28 pixels with an appropriate set of circular masks was used to detect circles. Triangles were detected by grouping edges in vertical, horizontal, and oblique segments. [4]

Priese et al. suggested a real-time traffic sign recognition system in which traffic signs are identified by the interpretation of their ideograms realised by different modules in the recogniser. There are modules for the position and direction of arrows, a module for the numerals, and another for prohibition signs, speed limits, and arrows on mandatory signs. [5]

Aoyagi and Asakura proposed a method to detect the traffic signs using brightness only. The object is extracted from the background using the Laplacian filter after using a smoothing filter to remove the noise. To obtain the binary image, a certain threshold is applied, and detection is carried out by genetic algorithms with search ability for the circular pattern which is given as gene information. [6]

2.2 PROBLEM IDENTIFICATION

Considering the object recognition and interpretation abilities of humans, it is a hard task to try to develop a computer-based system which should be able to support people in everyday life. There are a lot of conditions which are changing continuously such as luminance and visibility, which are handled by the human recognition system with ease but present serious problems for computer-based recognition. Looking at the problem of road and traffic sign recognition shows that the goal is well defined, and it seems to be a simple problem. Road signs are located in standard positions, and they have standard shapes, standard colours, and their pictograms are known. To see the problem in its full scale, however, a number of parameters that affect the performance of the detection system need to be studied carefully. Road sign images are acquired using a digital camera for the purpose of the current analysis. However, still images captured from a moving camera may suffer from motion blur. Moreover, these images can contain road signs which are partially or totally occluded by other objects such as vehicles or pedestrians. Other problems, such as the presence of objects similar to road signs, such as buildings or billboards, can affect the system and make sign detection difficult. The system should be able to deal with traffic and road signs in a wide range of weather and illumination variant environments such as different seasons, different weather condition e.g., sunny, foggy, rainy and snowy conditions.

2.2.1 Potential Difficulties

In addition to the complex environment of the roads and the scenes around them road signs can be found in different conditions such as aged, damaged, disoriented etc. and hence the detection and recognition of these signs may face one or more of the following difficulties:

The colour of the sign fades with time as a result of long exposure to sun light, and the reaction of the paint with the air, The presence of obstacles in the scene, such trees, buildings, vehicles and pedestrians or even signs which occlude other signs, The presence of objects similar in colour and/or shape to the road signs in the scene under consideration, such as buildings, or vehicles They could be similar to the road sign in colour, shape or even both. Suppose have two different cases; the first case shows a fence with a similar colour to the road sign. In the second case the post box has similar shape and colour to the signs and it is located very close to the road sign.

Signs may be found disoriented, damaged, or occluded by any kind of obstacles, even by some other signs, The signs have two different damaged signs. The one on the left has damage to the red rim, while the one on the right is very old, rusted, damaged and the colour has faded due to aging.
With various functionality, this project is focused on the end customers. The recognition of characters being main feature, Edge detection can be configured during the picture processing process. This project's primary objective is to develop an expert system for "TSR (English) using Neural Network" that is capable of accurately identifying a specific character in a type format using neural network method.

2.2.2 Proposed System

This project consist of traffic sign recognition to make a computer application that recognizes different traffic signs like speed limits, no entry, traffic signals, turn left or right, etc. present in the image into different categories. In implementation, network is trained in such a way that it learns many filters of size 5 by 5 and then pass it through a ReLU activation function followed by 5 by 5 max pooling in both dimensions. Then the output of Maxpooling layers is taken to apply it to fully connected layers.

Fully connected layers contain around 500 units which will pass through another ReLU activation that enables to combine them into classes, which are useful for identifying the image.

CHAPTER 3

SYSTEM ANALYSIS

3.1 REQUIREMENT ANALYSIS

The specification of operational properties of software, along with information on how it interacts with other system components and the requirements that software must meet, are the outcomes of requirement analysis. Software engineers (also known as Analysts or Modelers in this role) can build models that depict user scenarios, functional activities, problem classes and their relationships, system and class behavior, and the flow of data as it is transformed by expanding on the requirements from earlier requirement engineering tasks.

The process of discovery, improvement, modelling, and specification goes into the requirements analysis task. The scope is detailed, having been initially created by us and adjusted during project planning. Models are constructed for the necessary data, information, control flow, and operational behavior.

3.1.1 Requirement Specification

A. Functional Requirements

The functional specifications of a system outline its functions.

1. The created system must be able to identify handwritten English text in the image.
2. If the user's input is not in the proper format, the system must display an error message.
3. The system must offer the user a high-quality service.
4. The system must offer precise character recognition.

B. Normal Requirements

Since the consumer has explicitly specified these requirements, they must be met for the customer to be satisfied.

1. The user interface for the application should be graphical.
2. It should be possible to input characters with different font sizes and styles.
3. By contrast, a database should be able to detect computer-based English characters.
4. The application should be able to match input traffic signs to patterns that have been stored.
5. Each image should have a minimum of 10*50 (characters * patterns) accessible.

C. Expected Requirements

These conditions fall under the category of implicit conditions. Even if the customer expects them, these needs are not clearly articulated by the customer.

1. Application should accept an image.
2. A user-friendly and simple to install application is required.
3. The process of character recognition is made more accurate by the use of neural network.
4. The program also recognizes Traffic signs.
5. Each image must have a minimum of 26*50 (characters * patterns) available.

D. Excited Requirements

The customer has neither indicated nor expected these criteria. However, the developer could add additional unanticipated criteria to satisfy the client.

1. Using NN training, the application interprets all of the traffic signs.
2. Using this program, it is necessary to recognize continuous handwritten characters.
3. The suggested system should be able to distinguish alphanumeric characters with unique symbols.
4. Creation of an TSR system for raucous images.

E. Nonfunctional Requirements

These requirements as their name would imply are those that do not immediately affect the particular services that the system provides.

Performance: An input image including traffic signs will be recognized roughly 90 percent of the time.

Functionality: This program will meet the necessary functional standards.

Availability: If an image has written text, only then will this system be able to retrieve the traffic signs.

Flexibility: It enables users to quickly load the image.

Learnability: The software makes learning easier and is very simple to use.

3.2 SOFTWARE REQUIREMENT SPECIFICATION (SRS)

3.2.1 System Requirement

This project needs the help of hardware and software requirements to be fit in the computer. The user and the toolkits and hardware and software requirements are required also.

3.2.2 Hardware Requirements

- RAM: At least 4 GB.
- Processor: Intel(R) core (TM) i3 or more. 2.00 Ghz.
- Hard Disk: 10 GB
- Internet connectivity: Yes (Broadband or Wi-Fi)

3.2.3 Software Requirements

- Windows 7/8/8.1
- Language: Python
- Visual Studio (VS)
- NumPy (version 1.16.5)
- cv2 (OpenCV) (version 3.4.2)
- Keras (version 2.3.1)
- TensorFlow (Keras uses Tensor Flow in backend & for some Image Preprocessing) (version 2.0.0)
- Matplotlib (version 3.1.1)
- Pandas (version 0.25.1)
- Pil (version 1.1.5)

3.3 TYPE OF SDLC MODEL

In our system, the incremental model is employed as the process model. The system's process model is depicted in the Figure 3.1. Software engineering must have a development strategy that covers the process, method, and tool levels; this approach is frequently referred to as the process model. This strategy is necessary to prevent genuine problems in an industrial context. Based on the scope of the project and its intended use, a software engineering process model is selected. We have decided to use an incremental model for our project.

1. A small number of customer requirements are swiftly implemented and provided to the customer using these models.
2. Expanded and modified standards are gradually implemented.
3. It mixes the iterative philosophy of prototyping with components of the linear sequential model.
4. A deliverable increment of the software is produced by each linear sequence.
5. Each linear sequence has four sections (Figure 3.1):
   - Analysis
   - Design
   - Code
   - Testing

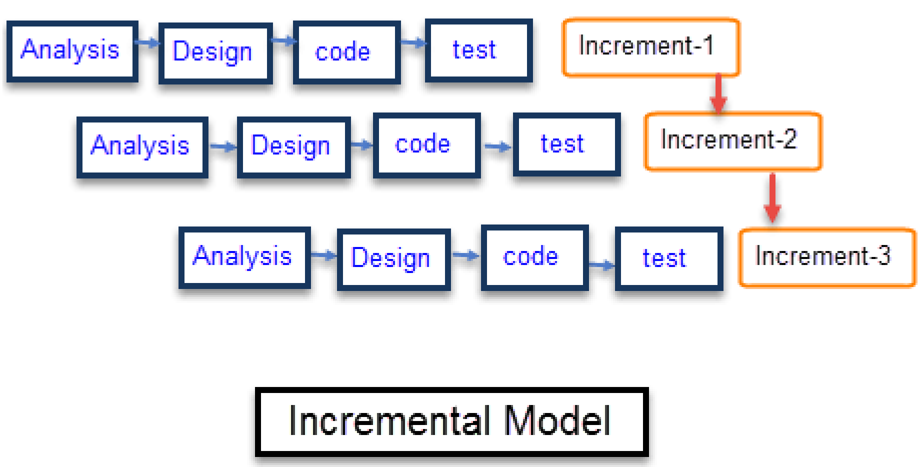


Figure 3.1: Incremental Model

1. **Analysis:**
   Evaluation of understanding the information domain, necessary operations, behavior, performance, and interface are all included. System and software requirements are outlined in writing and discussed with the customer.

2. **Design:**
   It is a collection of procedures that encompasses the four characteristics of procedural detail, software architecture, program data structure, and interface representation.

3. **Coding:**
   This stage involves translating the design to machine code.

4. **Testing:**
   It focuses on the logical internals of the software and makes sure that every assertion is true in order to find any hidden problems. In an incremental model, the first increment serves as the customer's primary model and is designed as such. Following the initial delivery, the product improves as more elements are added.

3.3.1 Advantages of Incremental Model

1. Quickly and early in the software life cycle, produces functioning software.

2. More adaptable—changing the requirements and the scope are less expensive.

3. More straightforward to test and fix within a shorter iteration.

4. Each built can receive customer feedback.

3.4 Data Flow Diagram (DFD)

Data flow diagrams (DFDs), often known as bubble charts, are graphical representations of information flow and the transformers that are utilized when data is transferred from source to destination. It accurately depicts the system requirements. DFD may be further divided into many levels, such as level 0 (Figure 3.2), level 1 (Figure 3.3 & Figure 3.4), and so on (Figure 3.5).

Figure 3.2: DFD (level 0) for TSR System

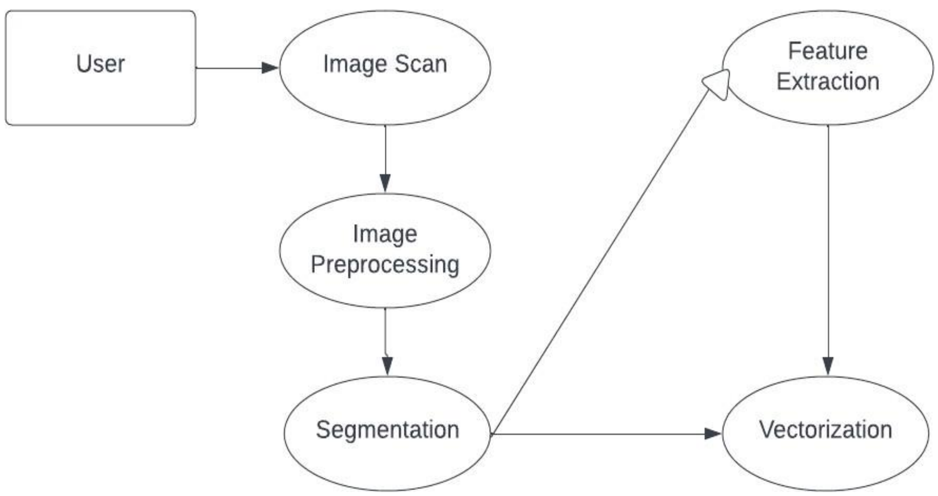


Figure 3.3: DFD (level 1) for Image Reader

Description:

- The image is uploaded by the user. (Figure 3.3)
- The image is preprocessed using different Image processing algorithms like Inverting Image, Gray Scale Conversion and Image Thinning.
- After preprocessing of the image, Segmentation is done. This is done with the help of removing the borders; dividing text into rows, rows into words, word into letters.
- Once the character is segmented, we generate the binary glyphs and calculate the summation of each row and columns values for Feature Extraction. (Figure 3.3)
- Further Vectorization is done.

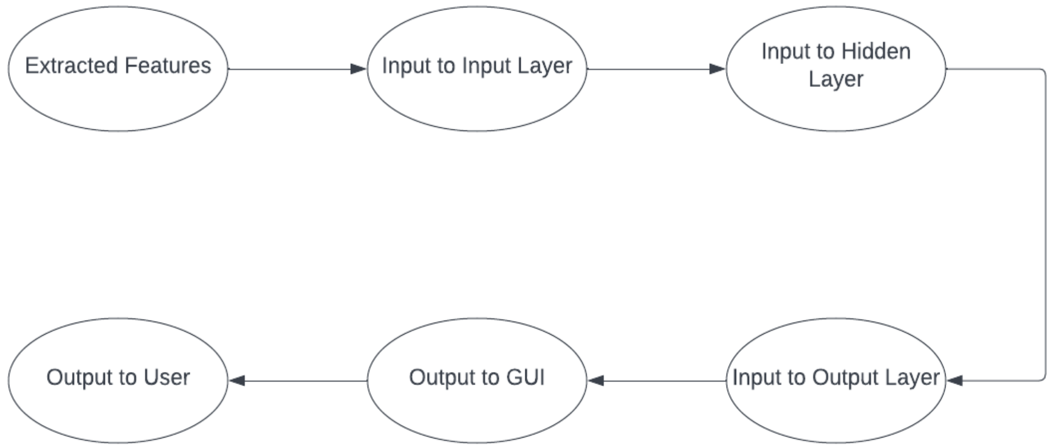


Figure 3.4: DFD (level 1) for Recognition Unit

Description:

- In here Figure 3.4, the extracted features of the image from last step i.e., Figure 3.3 is taken.
- Further processing of the characters is done through different layers in the recognition unit to process an output for the user.

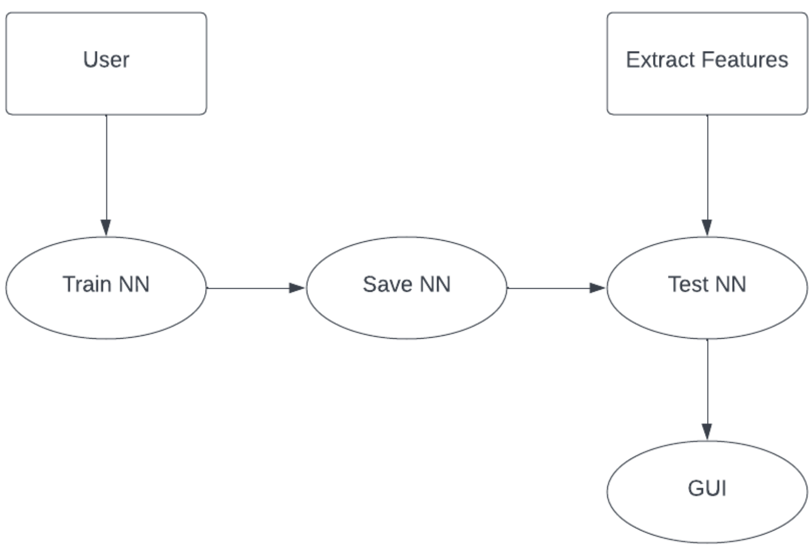


Figure 3.5: DFD for Neural Network

Description:

- The Neural Network is being trained in Figure 3.5 through the pre-processed images and extracted features of the image to give a Graphical User Interface (GUI)

3.5 WORKFLOW DIAGRAM

A workflow diagram, commonly referred to as a workflow, is a visual representation of the business process. The process demonstrates step-by-step how your job is accomplished from beginning to end using standardized symbols and forms. It also indicates who is in charge of the job at each stage of the process. A thorough workflow analysis must be completed before designing a workflow because it can reveal any potential flaws. You may define, standardize, and pinpoint crucial sections of your process with the use of a workflow analysis.

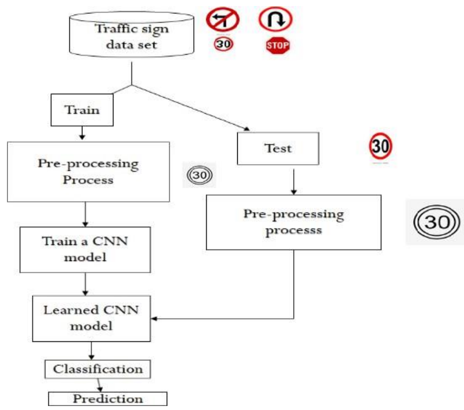


Figure 3.6: Workflow Diagram of TSR System

Description:

- In Figure 3.6, the workflow of the TSR System is depicted; which includes the processes as follows:
- User input the image; the image is pre-processed, segmented and the its features are extracted for further use.
- Then the extracted features are trained through neural network in order to recognize the characters in the image, at last an output is displayed on the screen of extracted characters.

3.6 USE CASE DIAGRAM

Use case illustration the functionality of the system as seen by external users is modelled in the use case view. A use case is a transaction between user and the system that expresses a coherent unit of functionality.

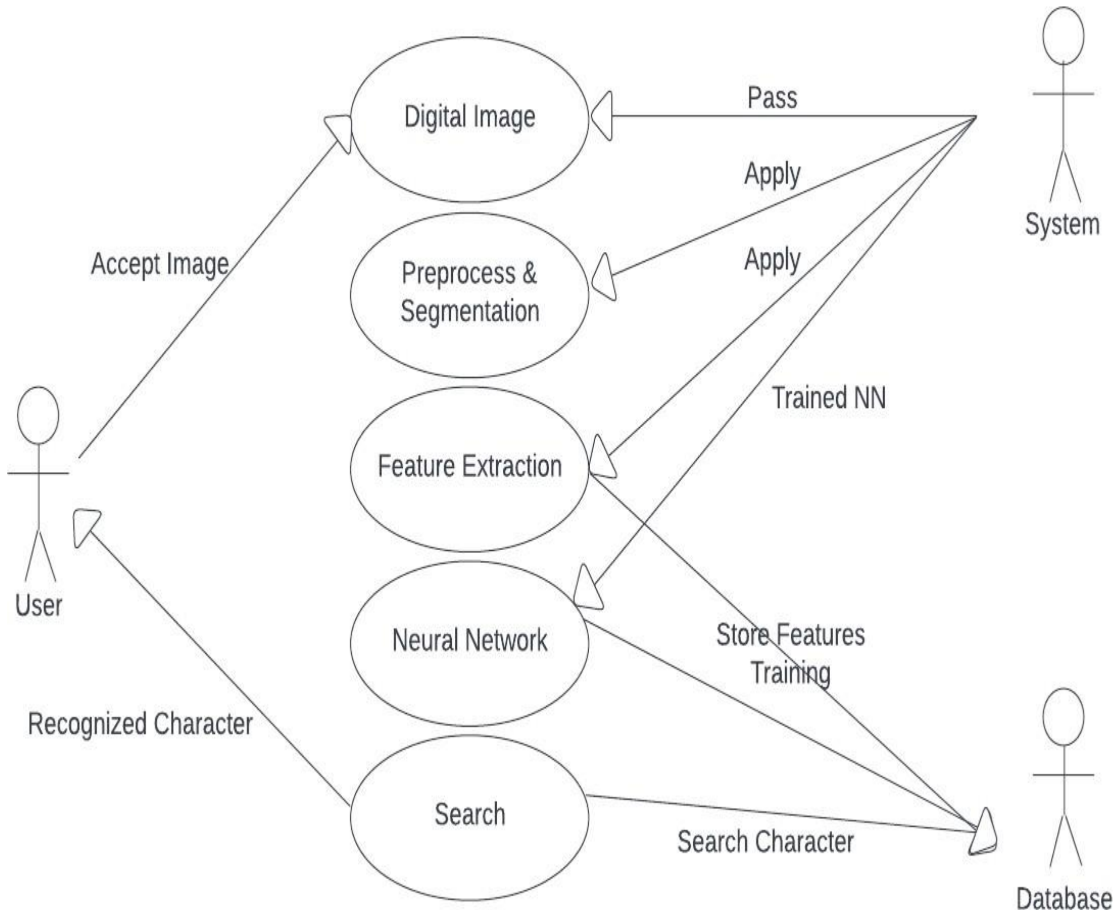


Figure 3.7: Use Case Diagram

Description:

- In Figure 3.7, the use case of the TSR System is shown where it depicts how the user, system and database works to provide the desired output.
- The user gives the input image upon which the system works to pre-process, segment and extract its features and finally train it using neural network to give out the desired output.

3.7 SEQUENCE & ACTIVITY DIAGRAM

A sequence diagram is a graphic representation of a scenario that demonstrates item interaction in a time-based sequence of what occurs first and what follows as Figure 3.8. Sequence diagrams define object roles and assist decide class responsibilities and interfaces by supplying crucial information. Because they are straightforward and simple to understand, this style of diagram works well at the first analysis stage of design. Use cases are typically related to sequence diagrams.

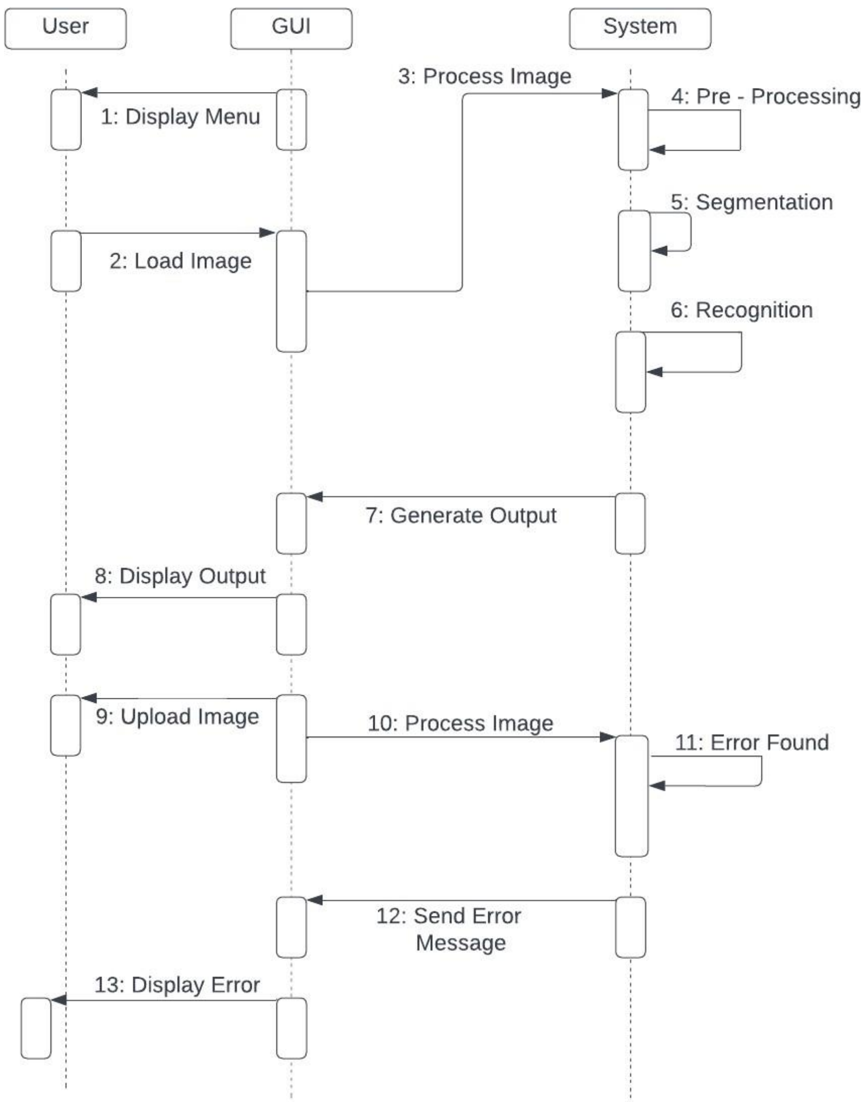


Figure 3.8: Sequence Diagram of the System

CHAPTER 4

SNAPSHOT

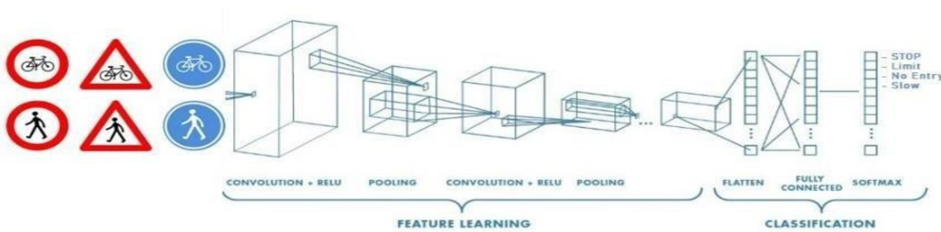
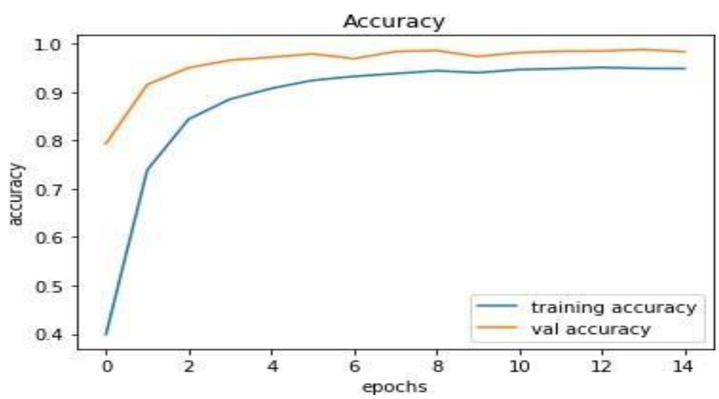


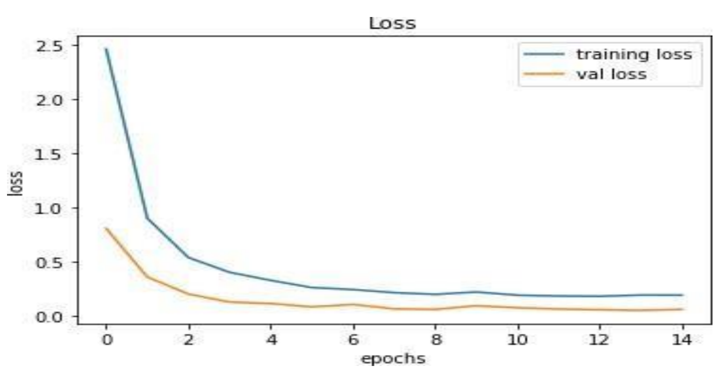
Figure 4.1: Neural Network with many convolutional layers

PLOT ACCURACY GRAPH

The graph for accuracy and loss is plotted. Figures 6-7 showing accuracy and loss respectively.



Accuracy of trained model CNN4.



Loss of trained model CNN 4.

Figure 4.2: Graphs

Figure 4.3 Showing results of General caution

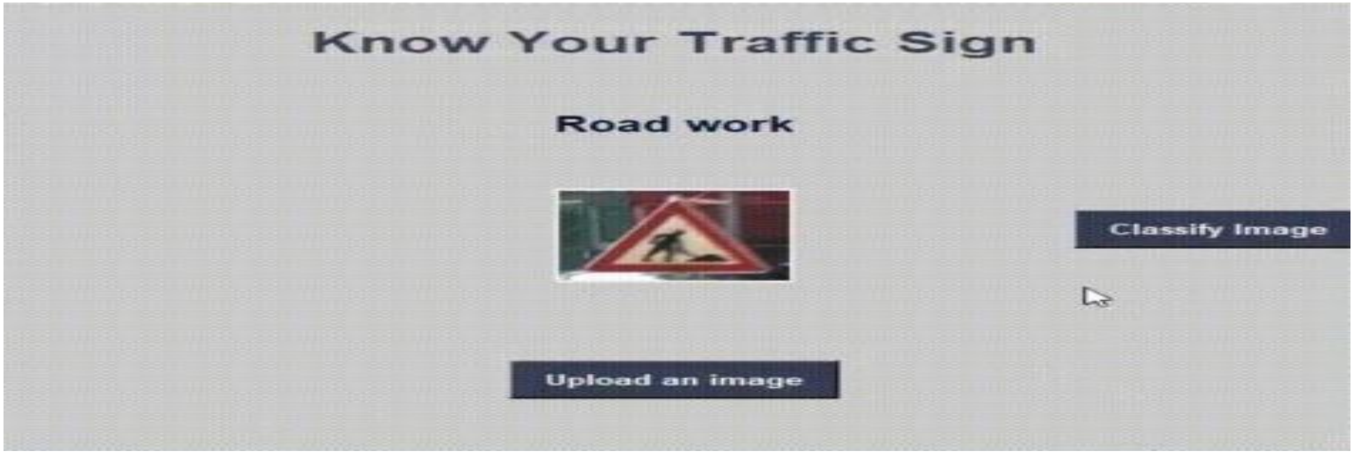


Figure 4.4 Showing results of Road Work

CHAPTER 5

CONCLUSION

5.1 CONCLUSION

An efficient traffic sign detection and recognition system is developed. The dataset went through a preprocessing, building CNN model and training and testing stages.

It got partitioned into training, testing and validating datasets. The final Deep CNN architecture proposed in this work consists of two convolutional layers, two maxpooling layers, three dropout layer and 2 dense layers.

This project is successfully classified the traffic signs classifier with 95% accuracy in 20 epochs and visualized how accuracy and loss changes with time, which is pretty good from a simple CNN model.

The techniques implemented in this project can be used as a basis for developing general purpose, advanced intelligent traffic surveillance systems. Future work will include increasing the size of the dataset and publishing it so that it can be used by other researchers for benchmarking purposed.

CHAPTER 6

FUTURE SCOPE

& REFERENCE

6.1 FUTURE SCOPE

The traffic signs focus on reduction of the traffic load on existing road network through various travel demand management measures.

Traffic signs should remove the encroachments, congestion and improve the traffic signal, road condition and geometrics features at intersection. In the future, these systems may take control of the vehicle under some circumstances.

Traffic sign detection and recognition systems are essential components of Advanced Driver Assistance Systems and self-driving vehicles. In TSR systems, the aim is to remind or warn drivers about the restrictions, dangers or other information imparted by traffic signs, beforehand.

6.2 REFERENCE

[1] Narejo, S., Talpur, S., Memon, M., y Rahoo, A. (2020):" An automated system for traffic sign recognition using convolutional neural network." 3C Tecnología. Glosas de innovación aplicadas a la pyme. Edición Especial, Noviembre 2020.

[2] Arasanipalai Mohan Aditya1, Sagar Moharir2 :"Study of Traffic Sign Detection and Recognition Algorithms"1, 2National University of Singapore, Institute of Systems Science, 25 Heng Mui Keng Terrace, 65162093, Singapore.

[3] Alexander Shustanov and Pavel Yakimov:"A Method for Traffic Sign Recognition with CNN using GPU"Samara National Research University, Moskovskoye shosse 34, Samara, Russia .

[4] Mohamed Yusof Radzak Electrical, Electronics and Automation Section, Universiti Kuala Lumpur Malaysian Spanish Institute, Kedah, Malaysia."Study on Traffic Sign Recognition International Journal of Research Studies in Computer Science and Engineering (IJRSCSE) Volume 2, Issue 6, June 2015, PP 33-39".

[5] Danyah A. Alghmghama , Ghazanfar Latif a, b*, Jaafar Alghazo a , Loay Alzubaidi a ."Autonomous Traffic Sign (ATSR) Detection and Recognition using Deep CNN "College of Computer Engineering and Sciences, Prince Mohammad bin Fahd University, Khobar, Saudi Arabia. b Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, Malaysia .

[6] Arjun Dieep*1, Ivin Johnson k*2, Ashley Benny*3:"TRAFFIC SIGN DETECTION BASED ON CONVOLUTIONAL NEURAL NETWORK" International Research Journal of Modernization in Engineering Technology and Science Rosemol Xavier, Asst. Professor, Department of Computer Science, De Paul Institute of Science & Technology, Angamaly, Kerala, India

Website:

https://www.python.org/

https://opencv.org/

https://www.kaggle.com/

https://colab.research.google.com/