RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Mini Project Report On

Traffic Sign Detection And Recognition

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

By

Daryl Antony Luiz (U2103074)

Hrishikesh M. Sreenivas (U2103103)

Edwin M. S. (U2103082)

Gautam Sunilkumar (U2103091)

Under the guidance of

Mrs.Amitha Mathew

Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
May 2024

CERTIFICATE

This is to certify that the mini project report entitled ***"Traffic Sign Detection And Recognition"*** *is a bonafide record of the work done by* ***Daryl Antony Luiz (U2103074)***, ***Hrishikesh M. Sreenivas (U2103103)***, ***Edwin M. S. (U2103082)***, ***Gautam Sunilkumar (U2103091)*** *submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Mrs.Amitha Mathew
Assistant Professor
Dept. of CSE
RSET

Dr.Saritha S.
Professor
Dept. of CSE
RSET

Preetha K.G.
Head of Department
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude towards **Dr. P. S. Sreejith**, Principal of RSET, and **Dr. Preetha K.G.**, Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake my mini project, "Traffic Sign Detection And Recognition".

I am highly indebted to my project coordinators, **Dr.Saritha S.**, Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Mrs.Amitha Mathew** for her patience and all the priceless advice and wisdom she has shared with me.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Daryl Antony Luiz
Hrishikesh M. Sreenivas
Edwin M. S.
Gautam Sunilkumar

Abstract

Traffic Sign Recognition is a critical aspect of intelligent transportation systems, contributing to road safety and efficiency. Our project focuses on developing a Traffic Sign Detection and Recognition system from images. The project aims to detect the traffic signs present in an image and recognize the sign. Convolutional Neural Networks (CNN) is utilized to detect the traffic signs in an image. The Convolutional Neural Networks (CNN) will be trained on traffic signs. A labelled dataset containing various traffic sign images is used to train the CNN to learn distinctive features and patterns associated with different types of traffic signs. The trained model can be used to recognize the detected sign. The proposed system can be utilized for applications in advanced driver assistance systems and autonomous vehicles. The Applications of this project could be: Autonomous Vehicle Navigation, Traffic Rule Compliance, Smart City Infrastructure, Road Network Analysis.

Contents

List of Figures

List of Abbreviations

- CNN-Convolutional Neural Network
- TSR-Traffic Sign Recognition

Chapter 1

Introduction

1.1 Background

Traffic signs serve as critical visual communication tools for guiding drivers, indicating rules, providing warnings, and ensuring road safety. Accurate detection and recognition of these signs are essential for safe and efficient navigation.In recent years, the importance of traffic sign detection and recognition has been amplified by several factors, including advancements in autonomous driving technology, increasing traffic congestion, and the growing emphasis on road safety. Since traffic signs act as a visual medium which is ubiquitous and provide information on various route conditions, it is important for a system to discern different traffic signs according as the situation demands.

1.2 Problem Definition

The traffic sign detection and recognition project aims to accurately identify and classify traffic signs. It plays a crucial role in enhancing road safety, supporting autonomous driving technologies, and optimizing traffic management.

1.3 Scope and Motivation

The traffic sign detection and recognition project aims to accurately identify and classify traffic signs. It plays a crucial role in enhancing road safety, supporting autonomous driving technologies, and optimizing traffic management. Traffic sign detection and recognition helps in enhancing road safety by accurate detection and recognition of traffic signs. Accurate identification of traffic signs ensures that vehicles adhere to speed limits, stop at intersections, and follow other regulations making informed decisions, such as adjusting speed or changing lanes contributing to smoother traffic movement, reduces congestion,

and optimizes road usage.

We aim to help the drivers by notifying the them when a speed limit sign or other traffic sign is detected, keeping the driver informed of speed limit changes and other important road information.

1.4 Objectives

Detection and Recognition of traffic sign from an Input image. Text Label and Audio Generation corresponding to the given Traffic Sign image.

1.5 Challenges

Challenges in overcoming limitations posed by the conditions (lighting,camera angle, obstruction, speed) to achieve real-time performance.balancing accuracy with speed to avoid missing visual objects during recognition.

1.6 Assumptions

- Imaging device calibration and positioning: The onboard cameras are accurately calibrated and positioned.
- Lighting conditions: The system assumes varying lighting conditions (day, night, shadows).
- Road types and environments: TSR adapts to urban, rural, and highway environments

1.7 Societal / Industrial Relevance

Traffic sign detection and recognition project could play a pivotal role in,

- Road Safety Enhancement: Automated detection and recognition of traffic signs contribute to improving road safety by allowing pre-cautionary measures to be planned earlier.
- Efficient Traffic Management:Can be used to calculate efficient routes with less risks.

- Autonomous Vehicles Development: Can have significant impact on vehicle performance since traffic signs can point out to nature of route beforehand.
- Urban Planning and Infrastructure Development: The data on traffic signs reflects the topographic and cultural significances.

1.8 Organization of the Report

- **Introduction:** Chapter 1 covers the background of the project, the problem definition, the scope and motivation, the objectives,the societal and industrial relevance, the assumptions and the challenges faced by the project.
- **Software Requirements Specifications:** Chapter 2 outlines the functional and nonfunctional requirements of the traffic sign detection and recognition software. It defines the overall description of the software, external interface requirements, system features, and other nonfunctional requirements necessary for the deployment of the software.
- **System Architecture and Design:** Chapter 3 provides an overview of the project's technical framework. It includes discussions on the system overview, architectural design, identified datasets, proposed algorithms, implementation strategies, module division, and a work presented as a Gantt chart for project planning and management.
- **Results and Discussion:** Chapter 4, results and discussions portrays the results obtained on realizing the project's aims and provides feedback on overall effectiveness of the project.
- **Conclusion:** Chapter 5, summarises the project and concludes it with possible suggestions that can be done to improve the project.

Chapter 2

Software Requirements Specification

2.1 Introduction

2.1.1 Purpose

The purpose of this software is to take in images of traffic signs as inputs, detect traffic signs, recognize the traffic sign and then produce the labels of the corresponding traffic signs as results.

2.1.2 Product Scope

The purpose of this software is to train the model to accurately detect and recognize traffic signs. By recognizing traffic signs, the software can contribute to smarter traffic flow management and also aids in addressing road safety issues.

The software aims to achieve better detection accuracy rate than the state of art method and ensure the system's adaptability to various environmental conditions (lighting, weather, things passing by etc).

This software aligns with corporate goals of innovation in smart city solutions and supports business strategies aimed at improving urban living standards. It serves as a stepping stone towards the broader vision of intelligent transportation systems where data-driven decisions lead to safer and more efficient urban environments.

2.2 Overall Description

2.2.1 Product Perspective

During the 1980's researchers Akatsuka and Imai laid the foundation for TSDR aimed to recognize traffic signs and enhance driver safety. As computer vision and mobile programming evolved, integrating TSR into vehicles became feasible. Developers harnessed platforms like OpenCV to create mobile apps capable of detecting traffic signs using object detection technology.

- Functionality and Interfaces:
    * TSDR processes images to detect and locate traffic signs.
    * The recognized signs are then classified based on their specific meanings (e.g. speed limits, warnings, prohibitions).
- Components:
    * Preprocessing Techniques: Enhance image quality by reducing noise, adjusting brightness, and correcting distortions.
    * Classification Techniques: Use machine learning algorithms (e.g., deep learning models) to classify detected signs.
    * Performance Evaluation Metrics: Assess the accuracy and response time of the TSDR system.
- Challenges and Advancements:
    * Real-Time Constraints: Overcoming limitations posed by the conditions (lighting, camera angle, obstruction, speed) to achieve real-time performance.
    * Recognition Speed: Balancing accuracy with speed to avoid missing visual objects.

2.2.2 Product Functions

- Image Input Handling: Accept preprocessed images of various classes of traffic signs.

- Sign Detection: Utilize the CNN to detect distinctive features in traffic signs.Identify and localize traffic signs within the images.
- Sign Recognition: Classify the detected signs into predefined categories using CNN model.
- Label Output: Generate and output accurate labels corresponding to the recognized traffic signs.
- Data Management: Store and manage the data of misdetected and recognized signs for further analysis and reporting.

2.2.3 Operating Environment

2.2.3.1 Hardware Platform

- Standard PC architecture
- Minimum 4GB RAM
- Required Nvidia GTX 1060 minimum GPU for image processing
- Processor (Intel i5 or better)

2.2.3.2 Software Platform

- Operating System: Compatible with minimum Windows 7, Linux distributions such as Ubuntu 18.04 LTS.
- Software Components: Requires Python 3.6 or higher, OpenCV library, and machine learning frameworks like TensorFlow, Keras or PyTorch.
- Google Colab for model training.

2.2.4 Design and Implementation Constraints

2.2.4.1 Hardware Limitations

- Processing Power: The TSR system must operate efficiently on resource-constrained hardware (e.g., embedded processors, GPUs).

- Camera Resolution: The system's performance depends on the quality of input images.
- limited functionality on systems without standard output devices such as monitors, speakers etc.

2.2.4.2 Regulatory and Safety Constraints:

- Privacy Protection Measures: The TSR system must adhere to international and local privacy laws and regulations in sourcing and usage of associated datasets.
- Cybersecurity: Protect against tampering or malicious attacks.

2.2.4.3 Software Dependencies and Interfaces:

- Operating System Compatibility: Ensure compatibility with the chosen OS (e.g.,Windows).
- Text to Speech engine: Should execute text to speech engine without performance overhead.
- Python: Should be able to run python and its associated libraries.

2.2.5 Assumptions and Dependencies

- The system is capable of running python and associated libraries.
- The target system is able to use web based technologies such as flask.
- The target system is capable of multimedia processing.

2.3 External Interface Requirements

2.3.1 User Interface

The web application interface consists of the image to be displayed, two buttons (the browse button and predict button) and sample images. Initially the image will be empty. On clicking the browse button or sample image, it is displayed on the screen and the predict button is clicked for detection and recognition. The traffic sign label is displayed as the output.

2.3.2 Hardware Interfaces

The minimum hardware requirements for the project are:

- Processor: Standard x64 processor running at cock speed of 1.5 GHz or higher is sufficient for basic operation of the software.
- Memory: A Minimum of 4 GB of RAM is required to run the software smoothly.
- Graphics Card: the software can run on integrated graphics.

2.3.3 Software Interfaces

The software shall integrate with specific software components, including deep learning libraries such as Tensorflow or NumPy for model development and training. TensorFlow is fundamental to the system's ability to learn from data,make predictions, and accurately identify traffic signs from images. NumPy provides an efficient interface for dealing with arrays, matrices, and a large collection of mathematical functions to operate on these data structures, which are essential components in the traffic sign detection and recognition. It should also integrate with several web frameworks such as JavaScript, Flask, pyttx3 etc, for output generation

2.4 System Features

2.4.1 Detection and Recognition of Traffic signs

1. **Description and Priority:**

   The traffic sign detection and recognition software utilizes Convolutional Neural Network (CNN) to detect and recognise the image features which closely resemble the form of a traffic sign. In the detection phase the images are pre-processed, enhanced, and segmented according to sign properties such as color or shape. The output is a segmented image containing potential regions which could be recognized as traffic signs. Given the role it holds in classification of signs, it presents high priority in image recognition.

2. **Stimulus/Response Sequences**

   The system is given an input of the form of an image through a webpage interface.

The image is then processed through a CNN model to produce appropriate labels.

3. **Functional Requirements**

   - REQ1: Detect the traffic sign.
   - REQ2: Recognise the traffic sign.
   - REQ3: Provide corresponding class label.

2.4.2 Display output using webpage

1. **Description and Priority:**

   The software utilizes trained model as backend to identify and detect traffic signs from input image and shows localised traffic sign from the input image inside a bounding box along with label identified inside a webpage. Since it is a necessary output it is of high priority.

2. **Stimulus/Response Sequences:**

   The trained CNN model identifies appropriate class number of the input image. The label corresponding to the class number is shown on the webpage along with a bounding box.

3. **Functional Requirements**

   - REQ1: Handle uploaded image.
   - REQ2: Prompt the user based on their actions.
   - REQ3: Shows output image with bounding box to the user.
   - REQ4: Handle input label

2.4.3 Read aloud function for the label

1. **Description and Priority:**

   The webpage provides a read aloud function of the label identified by the model. It provides an extra layer of accesibility for diverse types of user.

2. **Stimulus/Response Sequences:**

   The label identified is converted to speech using pyttsx3 engine and output is obtained.

3. **Functional requirements**

    - REQ1: Enable read aloud function of the label identified.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

1. **Response Time:** The system shall take no longer than 200 milliseconds to detect and recognize traffic signs from the input received from the camera under standard operating conditions.

2. **Accuracy:** The system shall achieve a minimum detection accuracy of 85% and a recognition accuracy of 85% under varying environmental conditions such as daylight, nighttime, and adverse weather conditions.

3. **Scalability:** The system shall be designed to efficiently handle an increase in workload, such as higher resolution input images without significant degradation in performance.

2.5.2 Safety Requirements

1. **Fail-Safe Mechanisms:** In the event of a system failure or a significant drop in recognition accuracy, the system shall revert to a safe state.

2. **Data Integrity:** The system shall ensure the integrity of the data processed. and generated, preventing errors that could lead to misinterpretation of traffic signs.

3. **Compliance:** The TSDR system shall comply with all applicable traffic safety and vehicle operation regulations and standards in the jurisdictions it is deployed

2.5.3 Security Requirements

1. **Tamper Detection:** The system shall include mechanisms to detect and report any unauthorized attempts to tamper with the software or hardware components of the TSDR system.

2.5.4 Software Quality Attributes

1. **Maintainability:** The system shall be designed to facilitate easy maintenance. The code shall be well-documented and adhere to industry-standard coding practices.

2. **Usability:** While the primary users of the TSDR system are the connected systems any user interfaces provided for configuration or monitoring shall be intuitive and user friendly.

3. **Reliability:** The TSDR system shall have a reliability of 99.9% ensuring consistent performance under specified conditions.

4. **Portability:** The software shall be designed to be portable across different operating systems and hardware platforms with minimal changes required.

Chapter 3

System Architecture and Design

3.1 System Overview

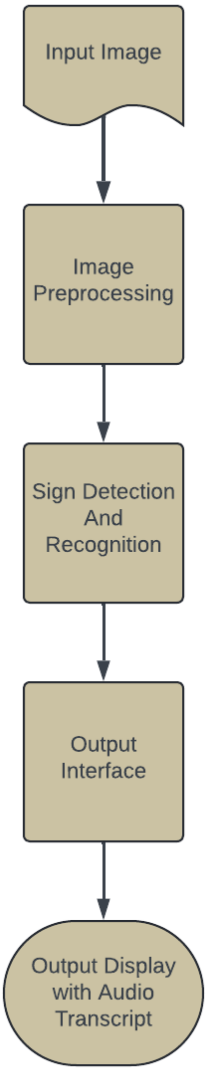


Figure 3.1: High-level view of system

3.1.1 Input image collection

A test input image to be recognised is collected through a web-page and is stored in server.

3.1.2 Image pre-processing

Image is pre-processed to be used in detection and recognition of traffic signs.

3.1.3 Sign Detection and Recognition

The CNN network detects and recognises the traffic signs based on parameters found during training of the model.

3.1.4 Output interface

The output label is produced corresponding to the given traffic sign input image.

3.1.5 Output Display with audio transcript

The generated output label of the image of corresponding class is shown in webpage and is read out.

3.2 Architectural Design

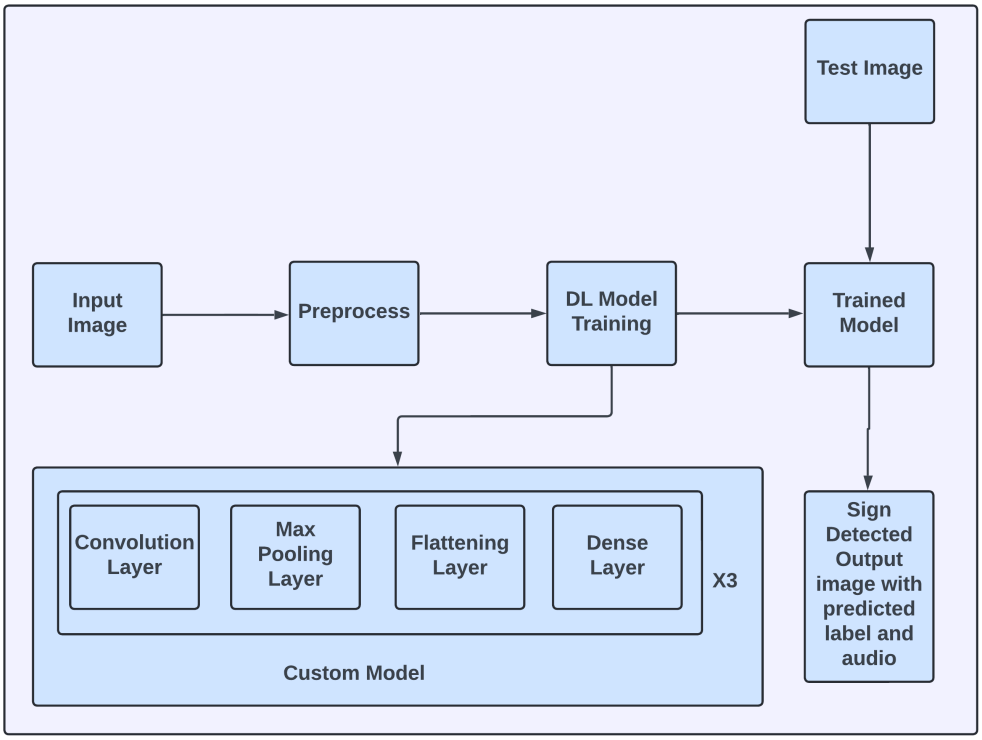


Figure 3.2: Architecture of the System

3.3 Dataset identified

For the current model we use "Indian Traffic Sign " Dataset for training. It consists of 59 classes and have an estimated 13971 images in the Dataset. The Dataset consists images that are preprocessed, grayscaled, and in Standardized lighting. The images are of the dimensions 32x32x3.

3.4 Proposed Methodology/Algorithms

1. **Data Preparation:** Obtain a labeled dataset suitable for your task. Split the data into training, validation, and test sets. Normalize pixel values (usually between 0 and 1).

2. **Architecture Design:** Define the architecture of your CNN. Common components include: Convolutional layers: Detect local patterns and features. Pooling

layers: Downsample feature maps.

3. **Fully connected layers:** Perform classification/regression.

4. **Convolutional Layers:**Apply convolutional filters (kernels) to input images. Convolve the filters across the image to create feature maps. Use activation functions (e.g., ReLU) to introduce non-linearity.Apply convolutional filters (kernels) to input images. Convolve the filters across the image to create feature maps. Use activation functions (e.g., ReLU) to introduce non-linearity.

5. **Pooling Layers:** Reduce spatial dimensions of feature maps. Common pooling methods: max-pooling, average-pooling.

6. **Flatten Layer:** Flatten the 2D feature maps into a 1D vector.

7. **Fully Connected Layers:**Add fully connected (dense) layers for classification/regression. Use softmax activation for multi-class classification.

8. **Compile the Model:** Specify the optimizer (e.g., Adam, SGD), loss function (e.g., categorical cross-entropy), and evaluation metric (e.g., accuracy).

9. **Training:**Feed training data into the model. Backpropagate gradients to update weights. Monitor validation loss and accuracy to prevent overfitting.

10. **Evaluation:** Evaluate the trained model on the test set. Calculate metrics (accuracy, precision, recall, F1-score).

11. **Fine-Tuning and Hyperparameter Tuning:**Experiment with different architectures, layer sizes, and hyperparameters. Use techniques like dropout, batch normalization, and learning rate schedules.

12. **Prediction:**Use the trained model to detect and recognise given traffic signs

3.5 User Interface Design

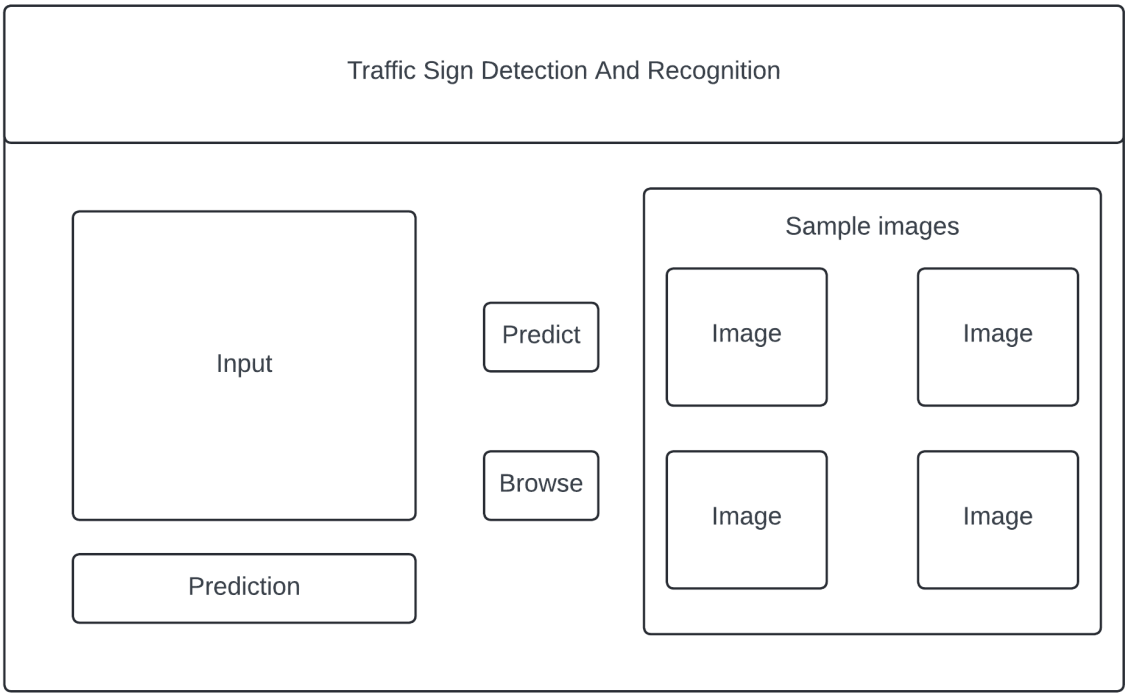


Figure 3.3: Wireframe diagram

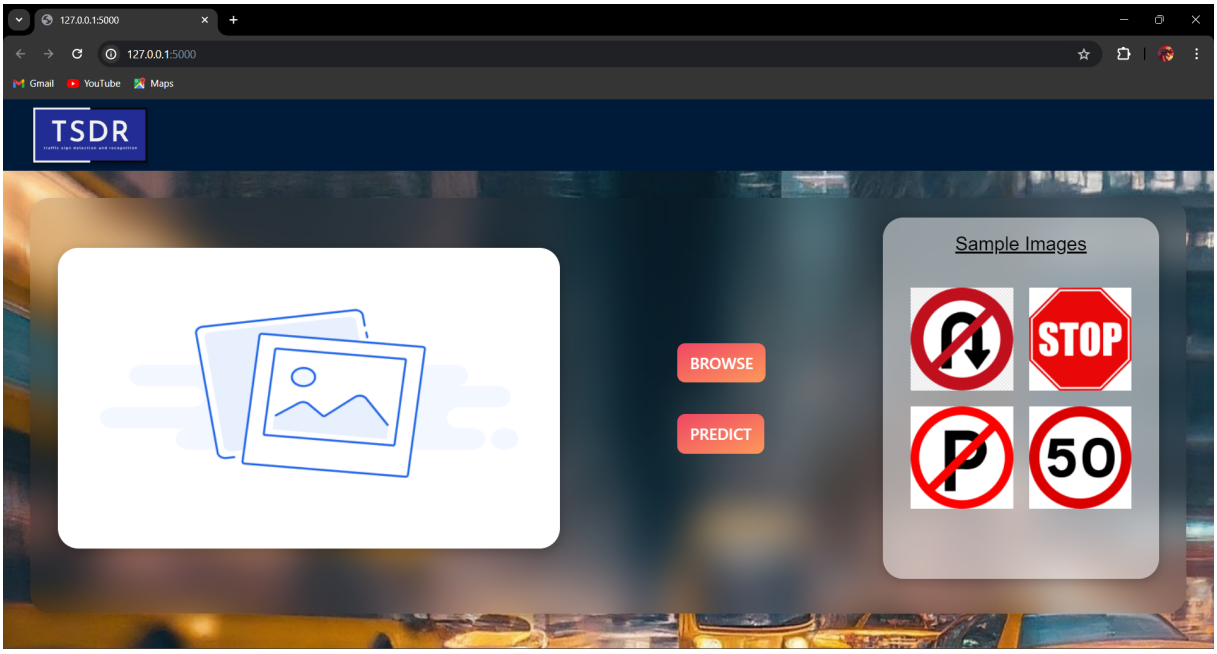


Figure 3.4: landing webpage

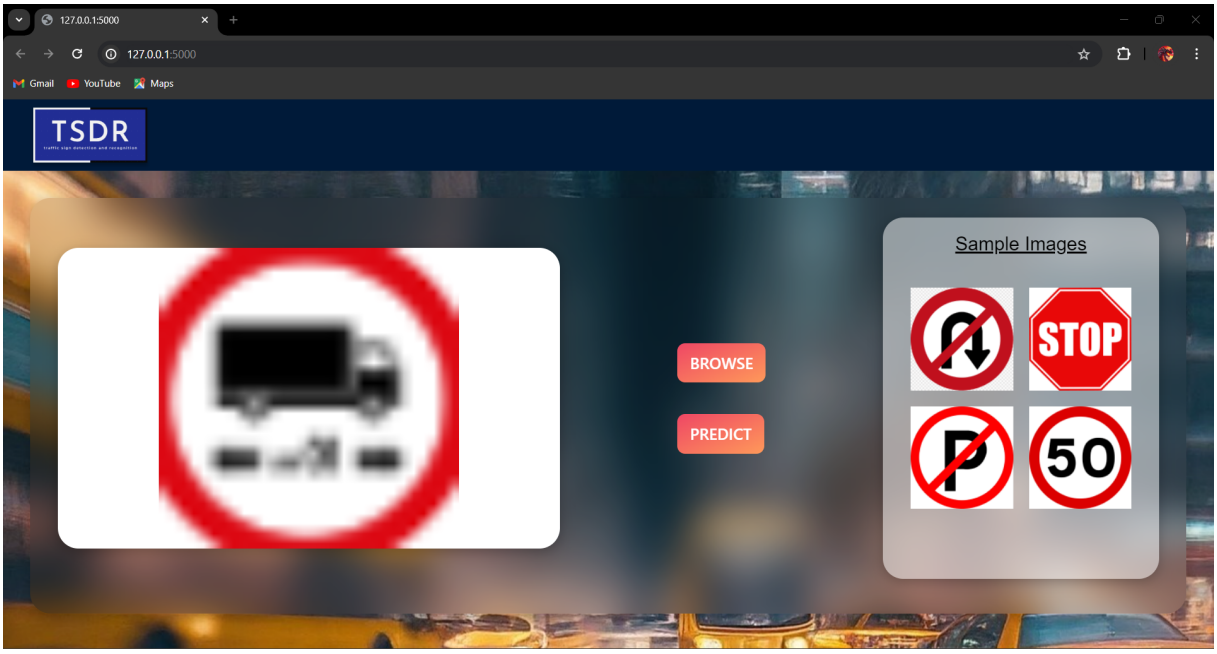


Figure 3.5: Test image is uploaded

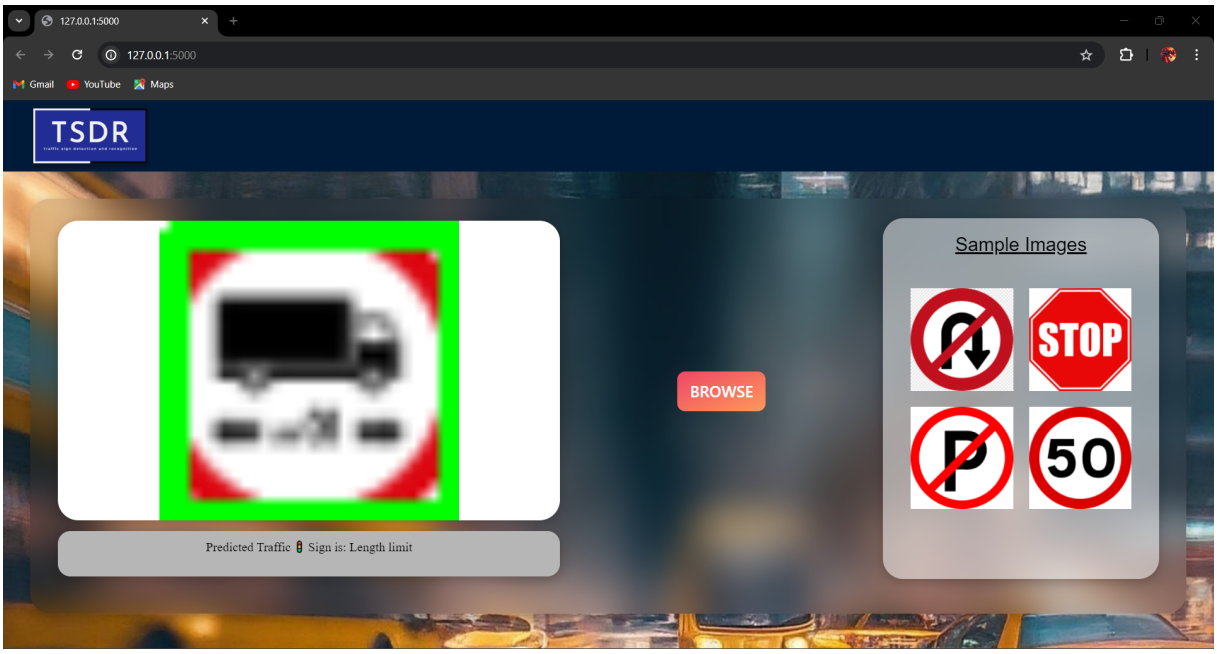


Figure 3.6: Final webpage

3.6 Description of Implementation Strategies

- **Numpy:** It is used to convert image into matrix form for processing.

  Eg: import numpy as np

- **Keras:** It is an open-source library that provides a Python interface for artificial

neural networks. Keras acts as an interface for the TensorFlow library.
Eg:keras.models.load_model

- **Tensorflow:** This library is used to build and interact with the neural network.
  Eg:from tensorflow.keras.optimizers import Adam

- **scikit-image:** Used to process images to HOG so that it can be given as input to CNN.
  Eg:from skimage.feature import hog
  from skimage import exposure

- **scikit-learn:** sklearn library is used to analyze the performance and accuracy of CNN model.
  Eg:from sklearn.model_selection import train_test_split
  from sklearn.metrics import confusion_matrix
  from sklearn.metrics import accuracy_score

- **PIL:** python library to open images,convert image to RGB format,resize the image etc.
  Eg:from PIL import Image

- **pyttsx3:** pyttsx3 is a text-to-speech conversion library in Python
  Eg: from pyttsx3 import *.

3.7 Module Division

1. **Input module(Daryl and Edwin):** Deals with input dataset and image preprocessing of images in the dataset.

2. **Detection and recognition module(Daryl and Edwin):** Deals with classification and recognition of traffic signs

3. **Output module(Hrishikesh and Gautam):** Deals with interfacing and display of output through a web interface

3.8 Work Schedule - Gantt Chart

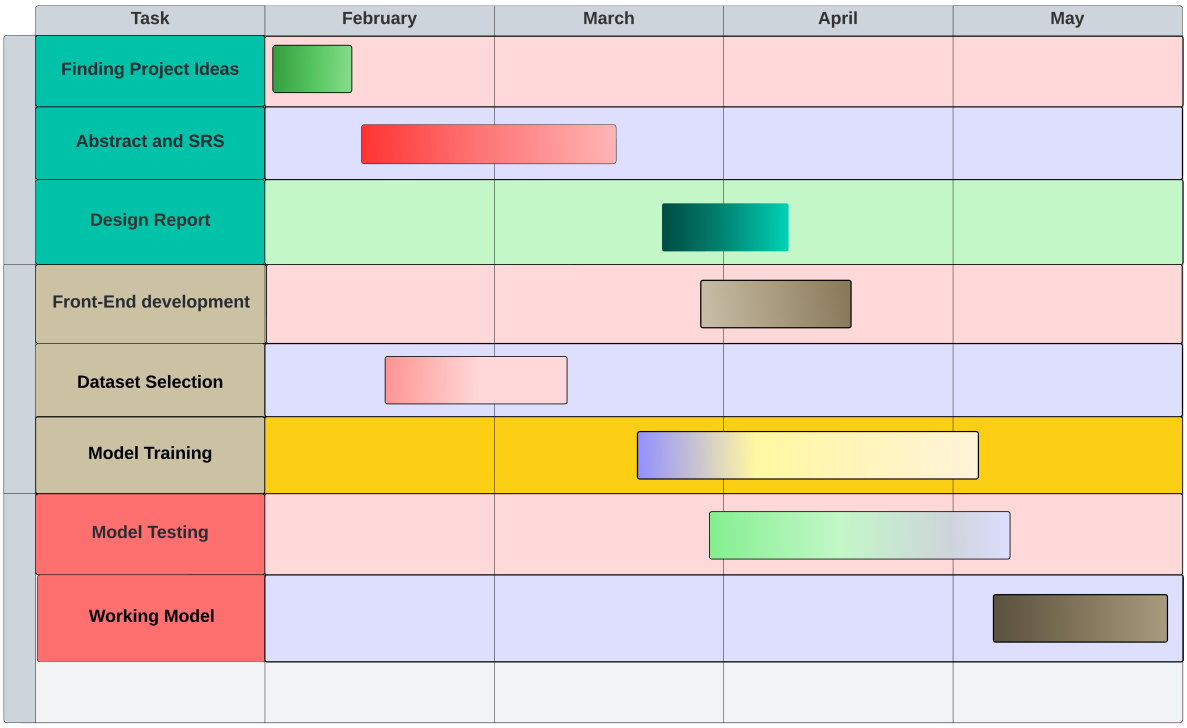


Figure 3.7: Gantt Chart

Chapter 4

Results and Discussions

4.1 Overview

The Traffic Sign Detection And Recognition Software utilizes a custom trained model based on "Indian Traffic Sign" dataset. It detects and recognizes test images of traffic signs and generate output label with a validation accuracy of 87.49%. The output label is displayed on web interface along with bounding box on the particular sign.The web interface also provides an audio that reads the predicted output label.

4.2 Testing

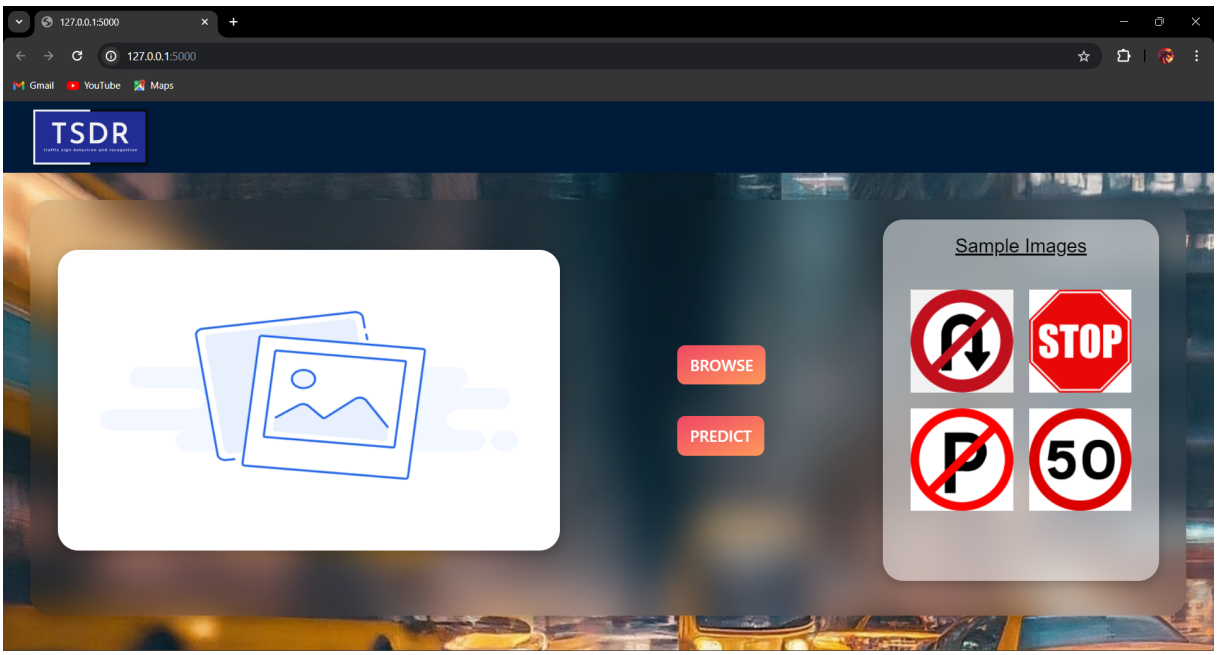


Figure 4.1: Initial webpage

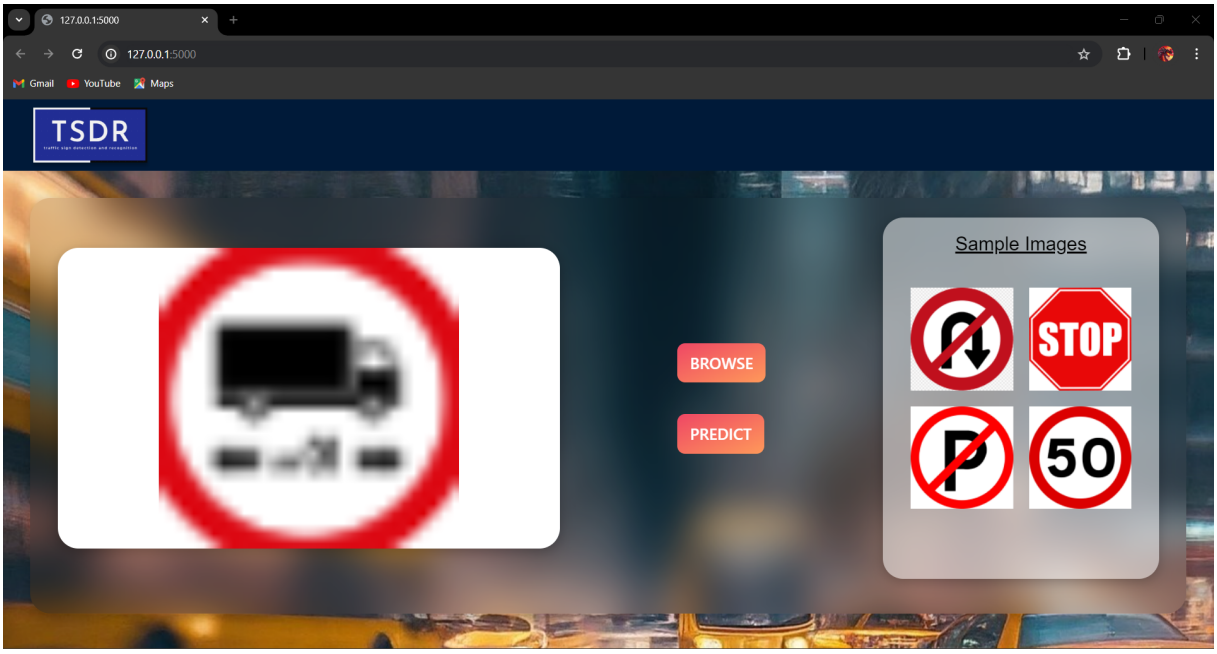


Figure 4.2: Test image is uploaded

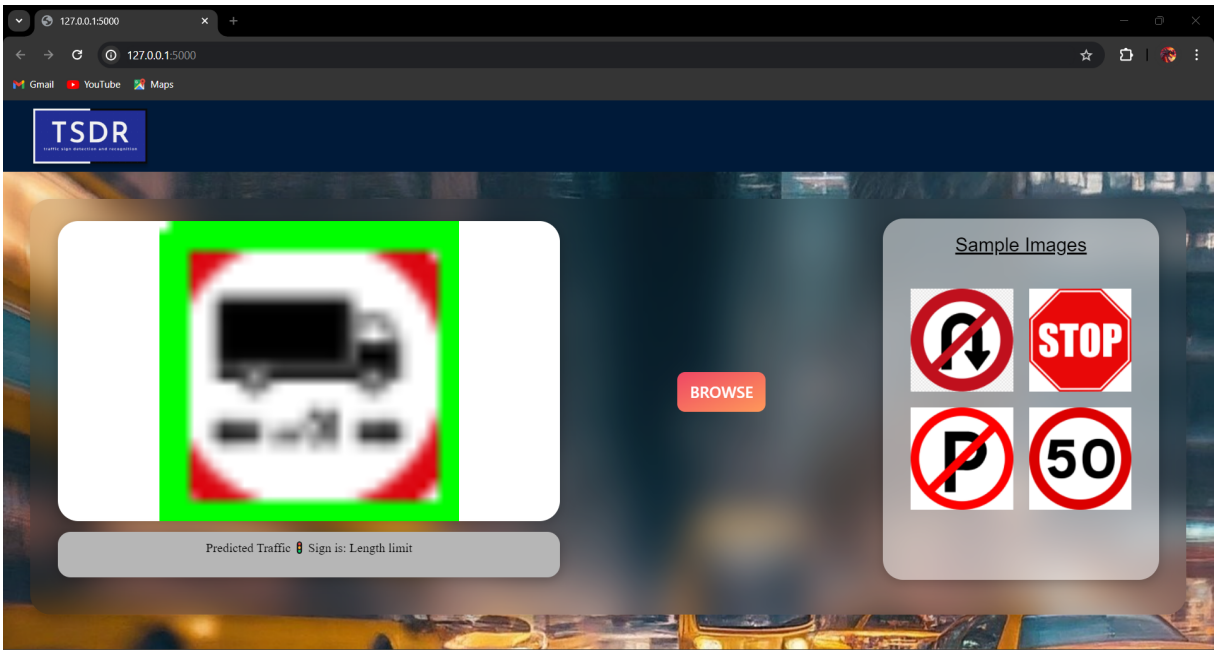


Figure 4.3: Final webpage

4.3 Quantitative Results

- **Accuracy:** The current model achieves a validation accuracy of 87.49%.

- **Confusion Matrix:** For the given model the confusion matrix obtained out of actual label:[23, 34, 41, 4, 6, 55, 3, 49, 7, 35] produces predicted label:[23 34 41 4 6 55 3 50 7 35]

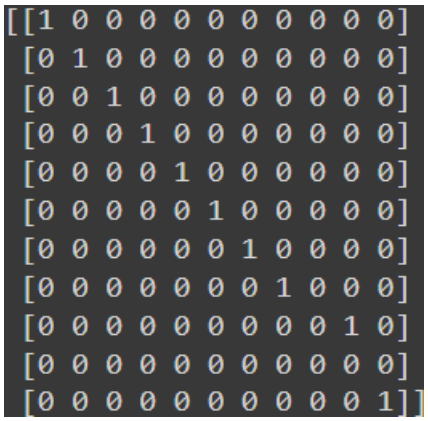


Figure 4.4: Confusion Matrix of model

- **RMSE:** The root mean square error for our model is 0.87

- **Model Performance**

Model	Accurarcy	Precision	Recall	F1 Score
Custom Model	0.8749	0.8918	0.8749	0.8797

Table 4.1: model performance

4.4 Graphical Analysis

4.4.1 Accuracy Over Epochs

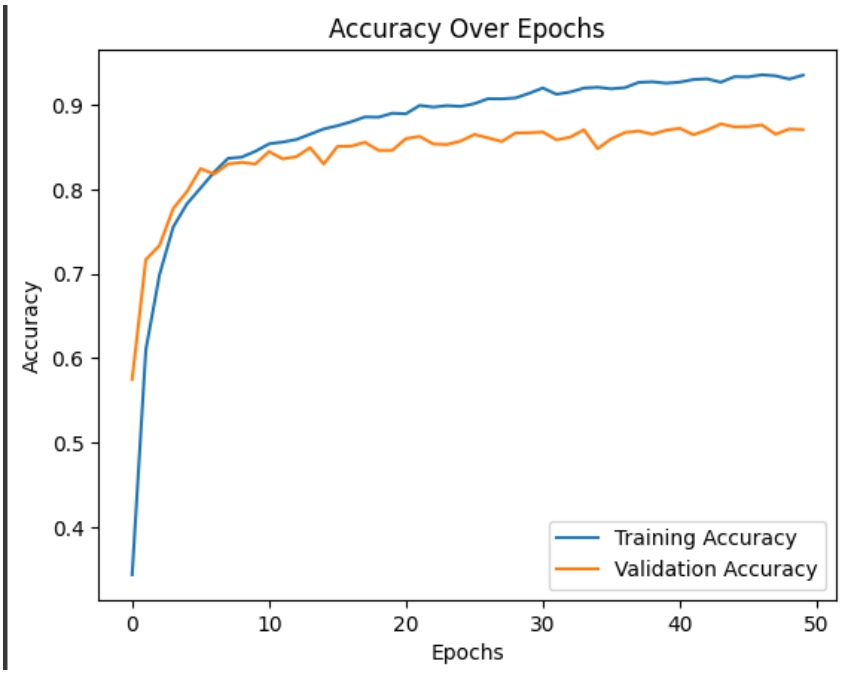


Figure 4.5: Accuracy Over Epoch

- **Training Accuracy:** It refers to the accuracy of a machine learning model on the data it was trained on. It is typically calculated by comparing the model's predictions to the actual target values in the training dataset.
- **Validation Accuracy:** It refers to the accuracy of a machine learning model on a separate dataset called the validation set. This dataset is distinct from the training set and is used to evaluate the model's performance during training. Validation accuracy is crucial for assessing how well the model generalizes to new, unseen data. It helps determine whether the model has learned meaningful patterns from the training data that can be applied to other similar data.

4.4.2 Loss Over Epochs

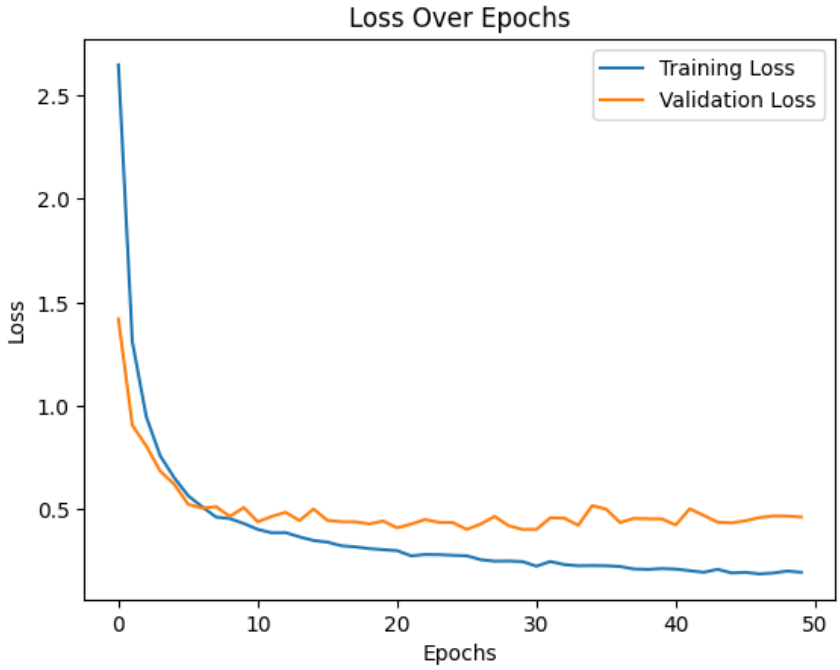


Figure 4.6: Loss Over Epoch

- **Training Loss:** It is a metric commonly used in machine learning to evaluate the performance of a model during the training phase. It measures how well the model is fitting the training data. The loss is typically calculated as the discrepancy between the model's predictions and the actual target values.

- **Validation Loss:** The validation loss is computed using the same loss function as the training loss but on the validation data. A decreasing training loss along with a decreasing validation loss indicates that the model is learning the underlying patterns in the data without overfitting. However, if the training loss continues to decrease while the validation loss starts increasing, it suggests that the model is overfitting to the training data and is not generalizing well.

4.5 Discussion

The software realizes the goal of detection and recognition of traffic signs form given test images. The trained model has a validation accuracy of 87.49%. However, as the figure 4.6 indicates the training loss decreases while the validation loss plateaued. This

suggests that the model shows overfitting. Thereby any cases other than the test images can be reliably identifed by the system.

Chapter 5

Conclusion

5.1 Conclusion

In conclusion, our project successfully achieved its objective of developing a software solution capable of detecting and recognizing traffic sign images using a custom-trained convolutional neural network (CNN) model. By leveraging various layers such as Convolutional, Max Pooling, Flattening, and Dense layers during the training process, our model attained an impressive accuracy rate of 87.49%.The culmination of our efforts is a user-friendly web interface that presents the model's outputs in an accessible manner. Users can conveniently interact with the system, receiving not only visual feedback but also an audio feedback of the identified traffic sign label. This feature enhances accessibility and usability, catering to a broader audience, including those with visual impairments.Overall, our project demonstrates the effectiveness of employing machine learning techniques, particularly CNNs, in real-world applications such as traffic sign recognition. The success of our software underscores the potential for leveraging advanced technologies to enhance safety and efficiency in transportation systems. As we move forward, further refinement and expansion of our model could lead to even greater accuracy and utility in addressing traffic management challenges.

5.2 Future Scope

In future, the project can be improved to increase its functionalities and features such as multiple traffic sign detection which could allow detection and recognition of multiple signs from a single image.

Other features such as Real time recognition of traffic detection and recognition with models like YOLO to apply usable efforts in real world applications such as Traffic monitoring and controlling services.

The UI of the current software can be improved to accomodate new features such as realtime traffic sign monitor, marking of hotspots based on traffic signs detected etc.

Bibliography

[1] Yucong, S., & Shuqing, G. (2021). Traffic sign recognition based on HOG feature extraction. Journal of Measurements in Engineering, 9(3), 142-155.

[2] Wali, S. B., Hannan, M. A., Hussain, A., & Samad, S. A. (2015). Comparative survey on traffic sign detection and recognition: a review. Przeglad Elektrotechniczny, 1(12), 40-44.

[3] Li, W., Li, D., & Zeng, S. (2019, November). Traffic sign recognition with a small convolutional neural network. In IOP conference series: Materials science and engineering (Vol. 688, No. 4, p. 044034). IOP Publishing.

[4] Fleyeh, H., & Dougherty, M. (2005, September). Road and traffic sign detection and recognition. In Proceedings of the 16th Mini-EURO Conference and 10th Meeting of EWGT (pp. 644-653).

[5] Megalingam, R. K., Thanigundala, K., Musani, S. R., Nidamanuru, H., & Gadde, L. (2023). Indian traffic sign detection and recognition using deep learning. International Journal of Transportation Science and Technology, 12(3), 683-699.

[6] Y. Sun, P. Ge and D. Liu, "Traffic Sign Detection and Recognition Based on Convolutional Neural Network," 2019 Chinese Automation Congress (CAC), Hangzhou, China, 2019, pp. 2851-2854, doi: 10.1109/CAC48633.2019.8997240.

Appendix A: Presentation

Traffic Sign Detection and Recognition

FINAL PRESENTATION

Guided By:
Mrs. Amitha Mathew

Daryl Antony Luiz
Hrishikesh M. Sreenivas
Edwin M.S.
Gautam Sunilkumar

Contents

Introduction

- Highly relevant in the field of autonomous driving and advanced driver assistance systems (ADAS).
- TSDR software improves self-driving cars capability.
- Accurate identification of Traffic signs.
- Managing efficient flow of traffic.

Problem Definition

- To design and develop the software to detect and recognize traffic sign images and generates the output label along with audio output and displays it in a web page .

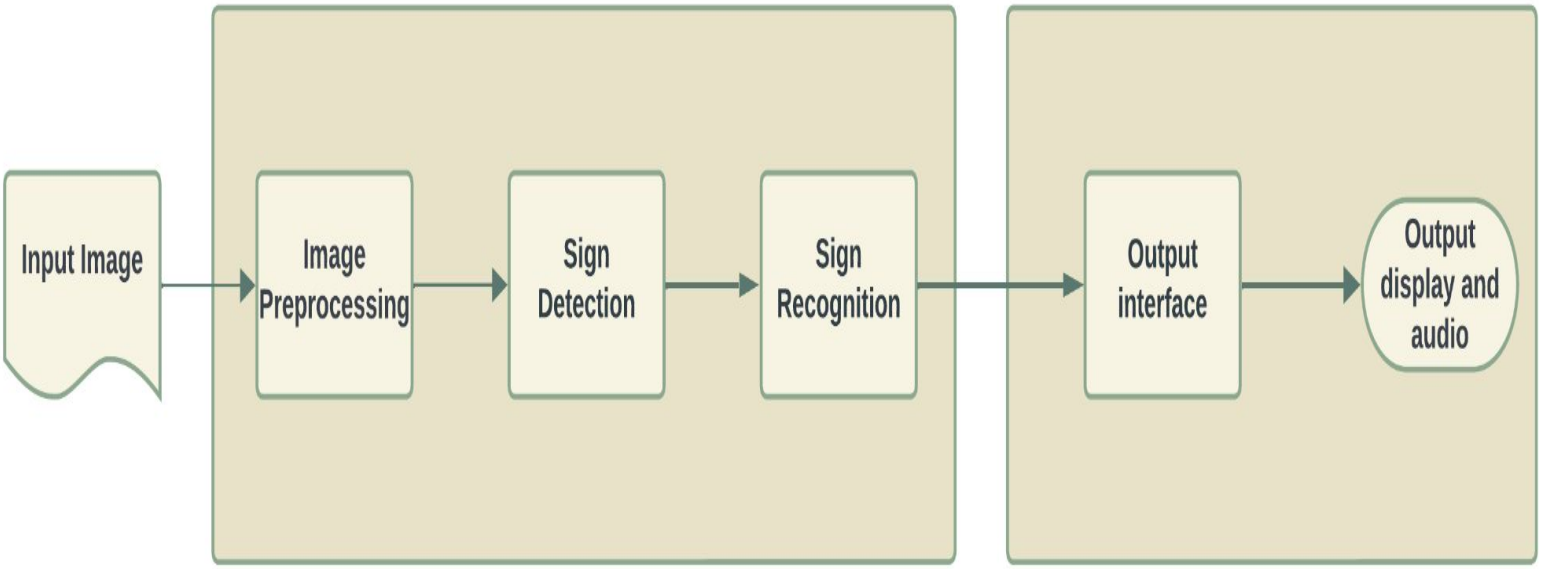
Objectives

- Detection of traffic sign from an input Image
- Image Recognition
- Text Label and Audio Generation corresponding to the given traffic sign image.

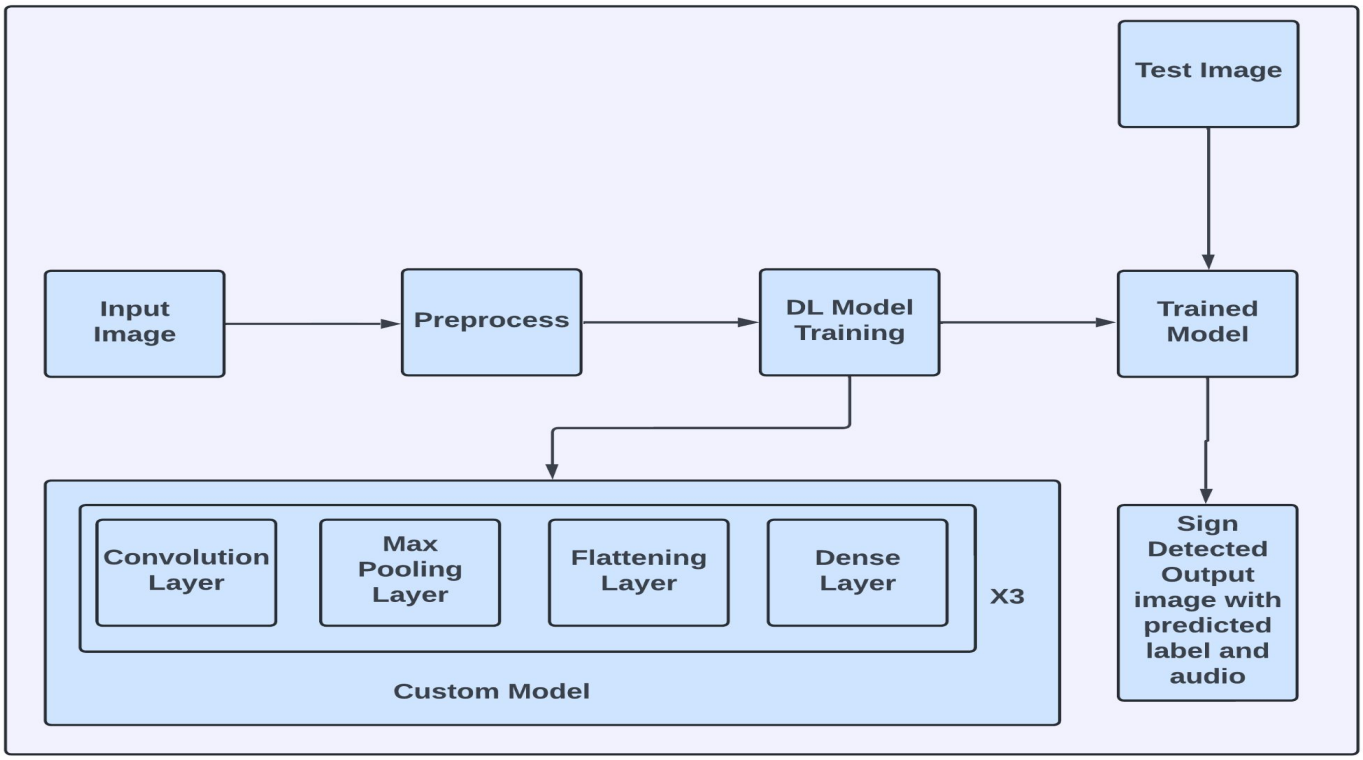
Scope and Relevance

- The TSDR project aims to accurately detect and recognize traffic signs, such as warning signs,regulatory signs etc.
- It enhances road safety for autonomous driving technologies, and optimizing traffic management.
- Processes input images containing traffic signs and return the correct labels or warnings.
- Does not include real-time video processing.

System Overview



System Architecture



Working of the model

Input image collection:The input image to be processed is collected through a webpage.
Image pre-processing:The image is pre-processed to be used in detection of traffic signs.
Sign Detection:The CNN detects the traffic signs based on parameters found during training of the model.
Sign Recognition:The traffic signs are recognized by the trained model.
Output interface:The output label is produced.
Output Display:Output is displayed with audio and bounding boxes on a webpage.

Algorithm

1. Data Preparation:

1.1 Obtain a labeled dataset suitable for the task.

1.2 Split the data into training, validation, and test sets.

1.3 Normalize pixel values (usually between 0 and 255).

2. Architecture Design:

2.1 Convolutional layers: Detect local patterns and features.

2.2 Pooling layers: Downsample feature maps.

2.3 Flatten Layer: Flatten the 2D feature maps into a 1D vector.

2.4 Fully connected layers: Perform classification.

3. Compile the Model: Specify the optimizer loss function and evaluation metric (accuracy, precision, recall, F1-score).

Algorithm

4. Training: Feed training data into the model. Backpropagate gradients to update weights. Monitor validation loss and accuracy to prevent overfitting.

5. Evaluation: Evaluate the trained model on the test set.

6. Fine-Tuning and Hyperparameter Tuning: Experiment with different architectures, layer sizes, and hyperparameters. Use techniques like dropout, batch normalization, and learning rate schedules.

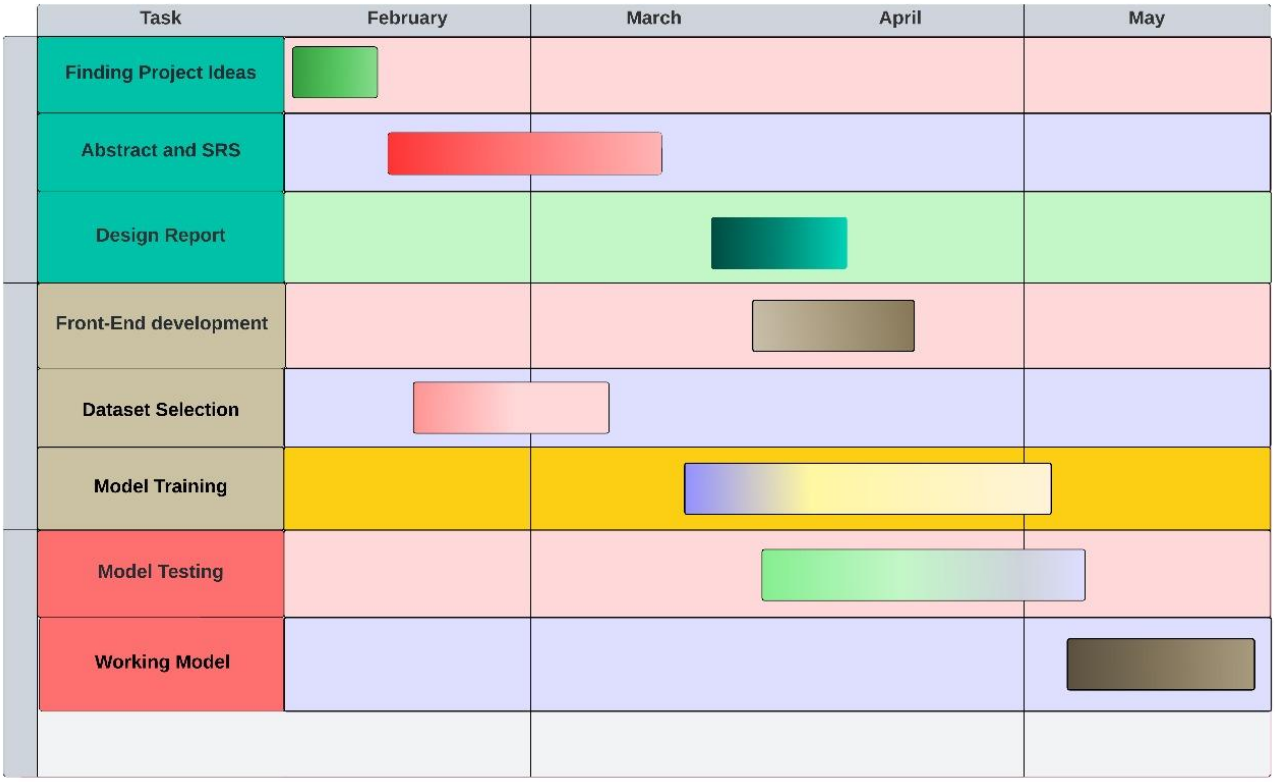
7. Prediction: Use the trained model to detect and recognise given traffic signs

Datasets

Indian Traffic Sign Dataset
https://www.kaggle.com/datasets/neelpratiksha/indian-traffic-sign-dataset

This dataset includes: 58 classes, 13815 images, 32 X 32 Resolution (27MB)

Work Division



Software/ Hardware Requirements

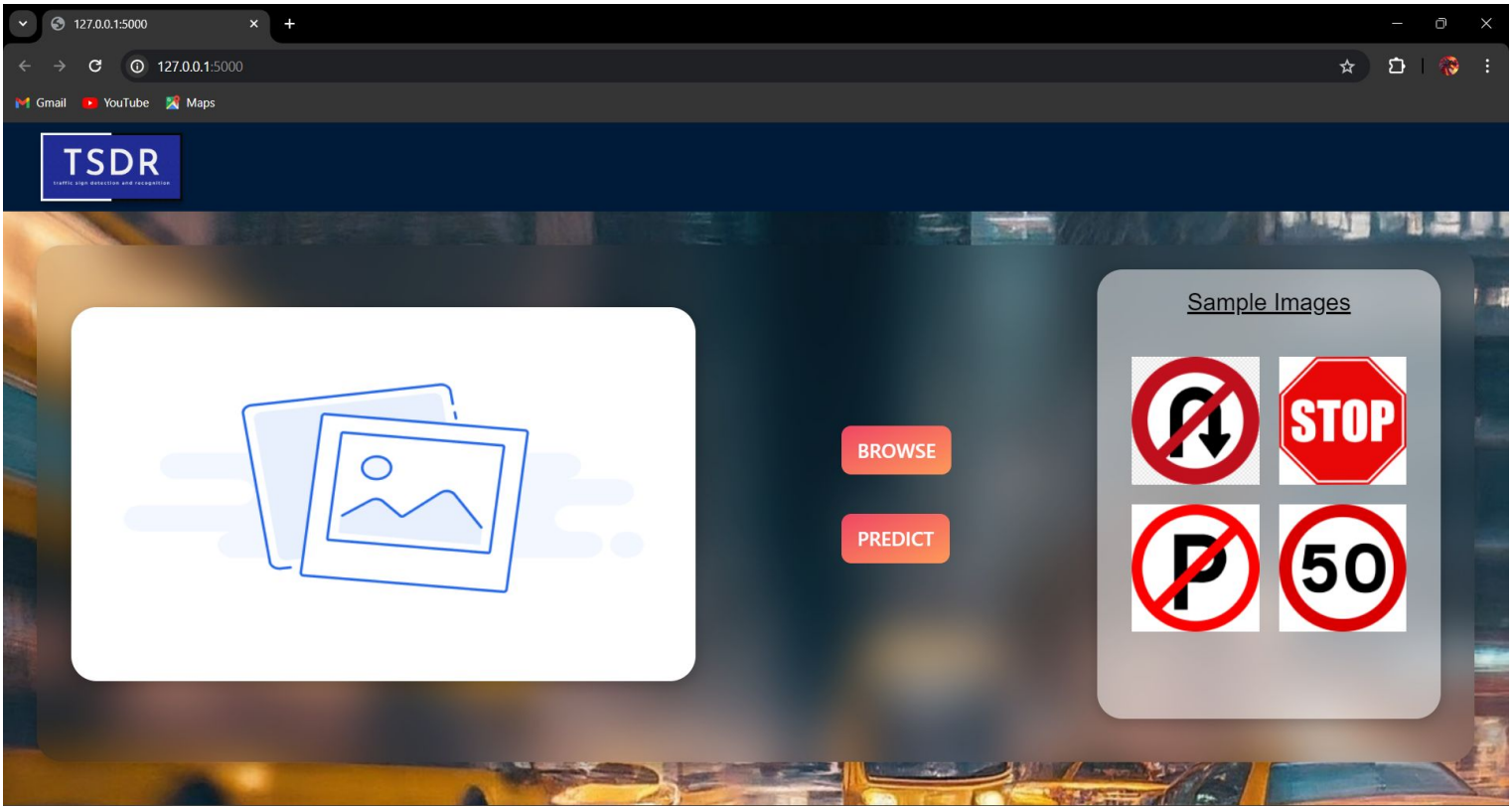
Software:

- Windows 7 or later
- MacOS 10.13 or later
- Ubuntu 18.04 or later
- Python(version 3.6 or later)
- Tensorflow
- Numpy
- Flask

Hardware:

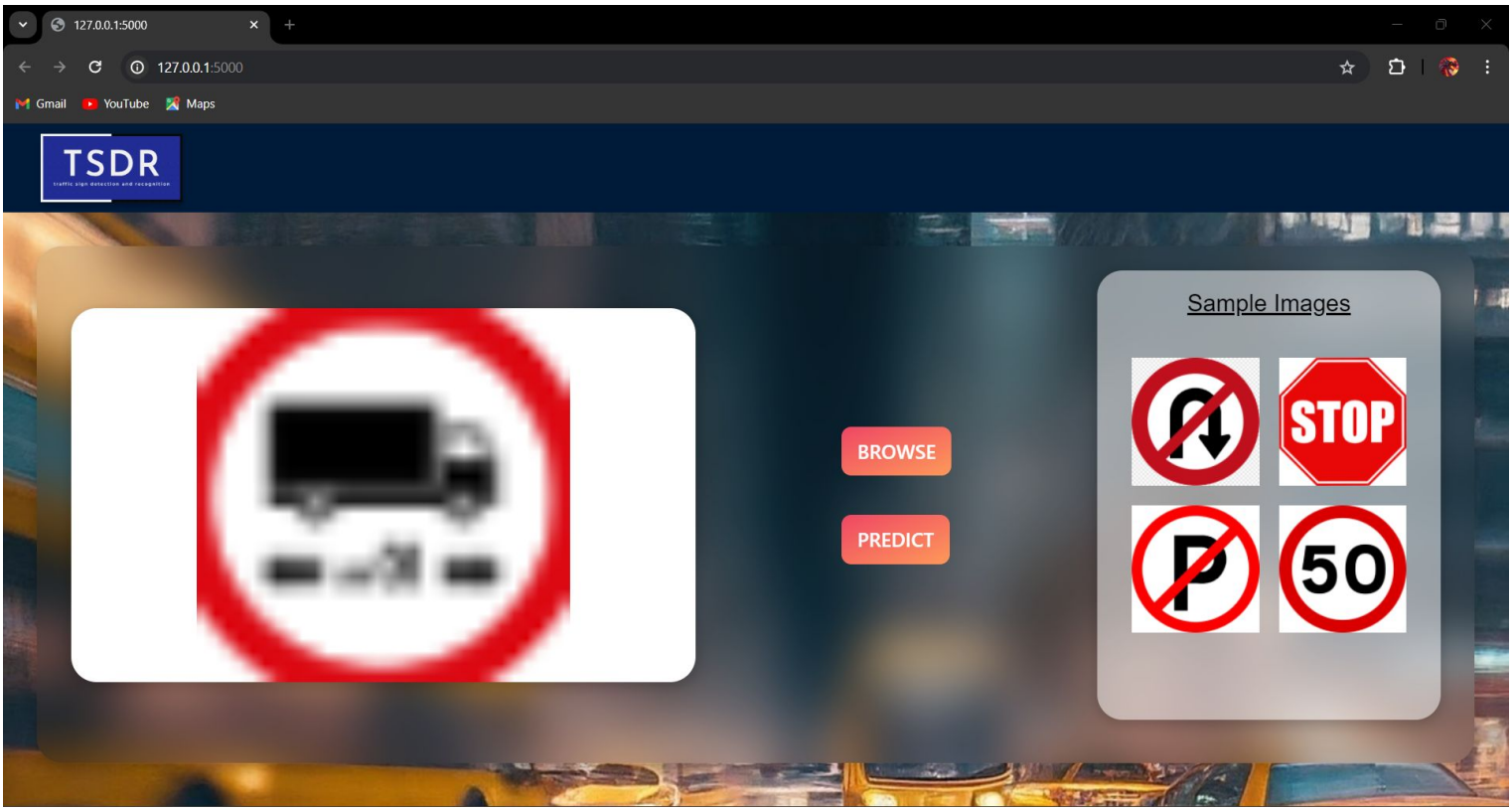
- Processor (Intel i5 or better)
- GPU: Nvidia GTX 1060 minimum
- Sufficient RAM (4GB minimum)

Results



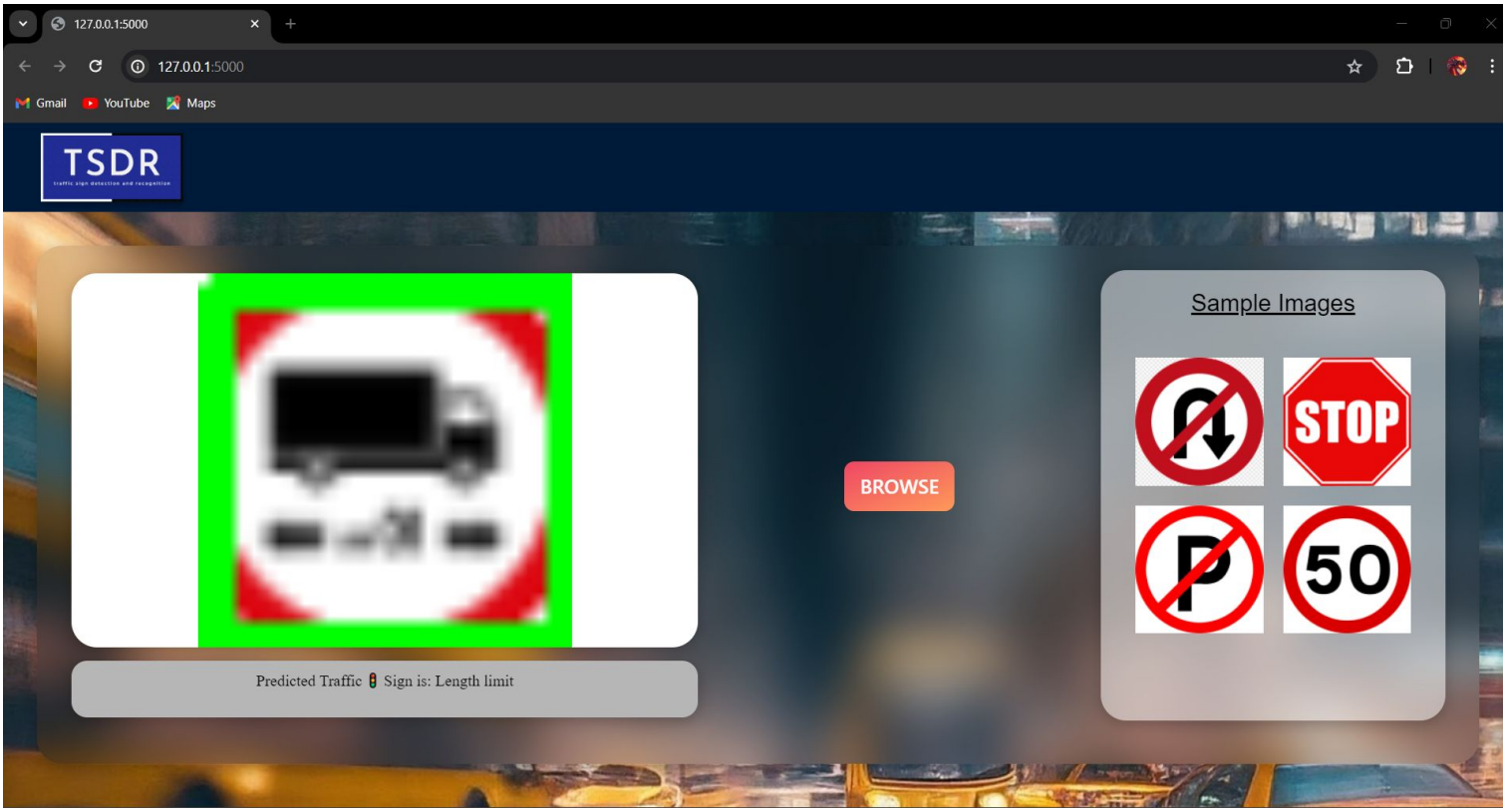
Landing Webpage

Results



Webpage with uploaded input

Results



Webpage with Final Output

Conclusion

- The software aims to develop a traffic sign and recognition system.
- The software will take traffic sign images as input.
- It performs detection and recognition on the given input images.
- It produces a labelled image which highlights the traffic sign and type of sign.

Future Enhancements

- Multiple traffic sign detection.
- Real time video traffic sign detection
- Training on new datasets using YOLO
- Web Page design improvement

References

1. Yucong, S., & Shuqing, G. (2021). Traffic sign recognition based on HOG feature extraction. Journal of Measurements in Engineering, 9(3), 142-155.

2. Wali, S. B., Hannan, M. A., Hussain, A., & Samad, S. A. (2015). Comparative survey on traffic sign detection and recognition: a review. Przeglad Elektrotechniczny, 1(12), 40-44.

3. Li, W., Li, D., & Zeng, S. (2019, November). Traffic sign recognition with a small convolutional neural network. In IOP conference series: Materials science and engineering (Vol. 688, No. 4, p. 044034). IOP Publishing.

4. Fleyeh, H., & Dougherty, M. (2005, September). Road and traffic sign detection and recognition. In Proceedings of the 16th Mini-EURO Conference and 10th Meeting of EWGT (pp. 644-653).

5. Y. Sun, P. Ge and D. Liu, "Traffic Sign Detection and Recognition Based on Convolutional Neural Network," 2019 Chinese Automation Congress (CAC), Hangzhou, China, 2019, pp. 2851-2854, doi: 10.1109/CAC48633.2019.8997240.

6.Megalingam, R. K., Thanigundala, K., Musani, S. R., Nidamanuru, H., & Gadde, L. (2023). Indian traffic sign detection and recognition using deep learning. International Journal of Transportation Science and Technology, 12(3), 683-699.

THANK YOU

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039
(Affiliated to APJ Abdul Kalam Technological University)

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes

After the completion of the course the student will be able to:

CO1:

Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)

CO2:
Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)

CO3:
Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)

CO4:
Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)

CO5:
Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
101003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
101003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
101003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.