Project Title: Traffic Sign Recognition for Self-Driving Cars

Course/Semester: 10

Group Name/No.: 09

Instructor: Md. Sabbir Al Ahsan

Submission Date: 30/10/2025

Members:

1. Name: Badhan Das	• ID: 223016512
2. Name: Raheta Sadeka	• ID: 223020312

1. Project Overview

- **Problem Solving:**
  Traffic signs are critical for ensuring road safety. Human drivers can sometimes miss or misinterpret signs due to fatigue, distraction, or visibility issues. This project aims to automate the process of recognizing traffic signs using machine learning—helping autonomous vehicles make faster and safer driving decisions.

- **Beneficiaries:**
  Self-driving car systems, driver-assistance technologies, and road safety research.

- **Approach (One-liner):**
  Implement a **Convolutional Neural Network (CNN)** in **Python** to classify traffic signs from images using the **GTSRB dataset**.

- **Expected Outcome:**
  A trained model that can accurately detect and classify traffic signs from static images or live video input.

2. Problem Statement & Objectives

Problem:

Autonomous vehicles need to accurately recognize traffic signs in real-time under varying conditions (lighting, angle, occlusion). Traditional computer vision techniques struggle to generalize well, necessitating a deep learning-based solution.

Objectives:

- **O1:** Develop a CNN-based model that classifies different traffic signs with high accuracy.

- **O2:** Implement a prototype system that can predict traffic sign labels from images or webcam feed in real time.

3. Dataset

Source(s):

German Traffic Sign Recognition Benchmark (GTSRB)
http://benchmark.ini.rub.de/?section=gtsrb&subsection=news

Size & Key Features:

- ~50,000 labeled images
- 43 traffic sign classes
- Varying illumination, rotation, and weather conditions

Preprocessing Needed:

- Image resizing (32×32 pixels)
- Normalization (pixel values scaled 0–1)
- One-hot encoding of labels
- Data augmentation (rotation, flipping, zoom) for better generalization

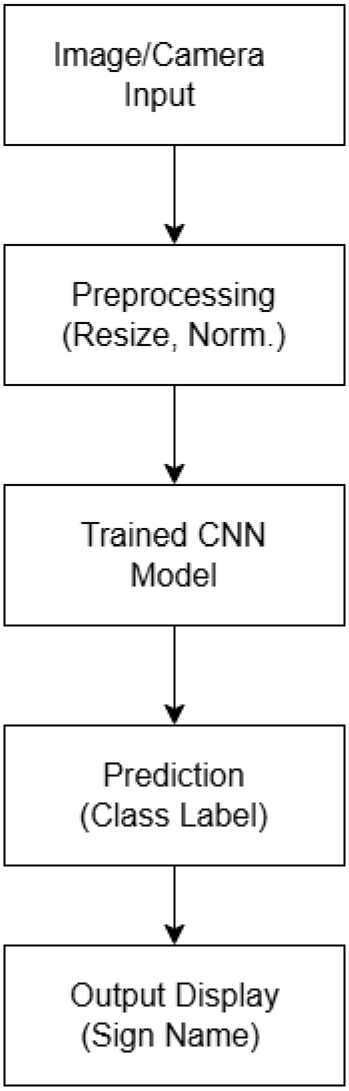
4. Methods / Models

- **Baseline:** Logistic Regression and Random Forest (for simple comparison)
- **DL Model(s):** Convolutional Neural Network (CNN)
- **Feature Extraction/Embeddings:** Automatic feature extraction using convolutional filters
- **Libraries:** NumPy, Pandas, Matplotlib, TensorFlow/Keras/PyTorch, OpenCV, Scikit-learn

5. System Design

- **Workflow:** Input (Image/Camera) → Preprocessing (Resize + Normalize) → Trained CNN Model → Prediction (Class Label) → Output Display (Traffic Sign Name)

- **Diagram:**



6. Evaluation Plan

Metrics:
Accuracy, Precision, Recall, F1-Score, Confusion Matrix

Data Split:

- Train: 70%
- Validation: 15%
- Test: 15%
- Ensure no data leakage across sets

7. Practical Applications

- **Use Case 1:** Integration with self-driving car systems to identify and respond to traffic signs automatically (e.g., slowing down at speed limits, stopping at stop signs).

- **Use Case 2:** Road safety monitoring and driver assistance systems that alert human drivers about nearby traffic signs in real time.