



# TRAFFIC SIGN DETECTION

Using CNN and YOLO V8

ATLURI CHARITHA SRI

B220213CS

AKULA JYOSHNAVI

B220668CS

B V D SAHITHI

B220753CS

# INTRODUCTION

## **What is Traffic Sign Detection?**

Traffic sign detection is a computer vision task that involves identifying and locating road signs in images . It plays a crucial role in helping machines interpret traffic environments.

## **Why is it Important in Autonomous Driving?**

Autonomous vehicles must recognize and follow traffic rules. Detecting traffic signs such as speed limits, stop signs, and warnings ensures safe and lawful driving behavior without human intervention

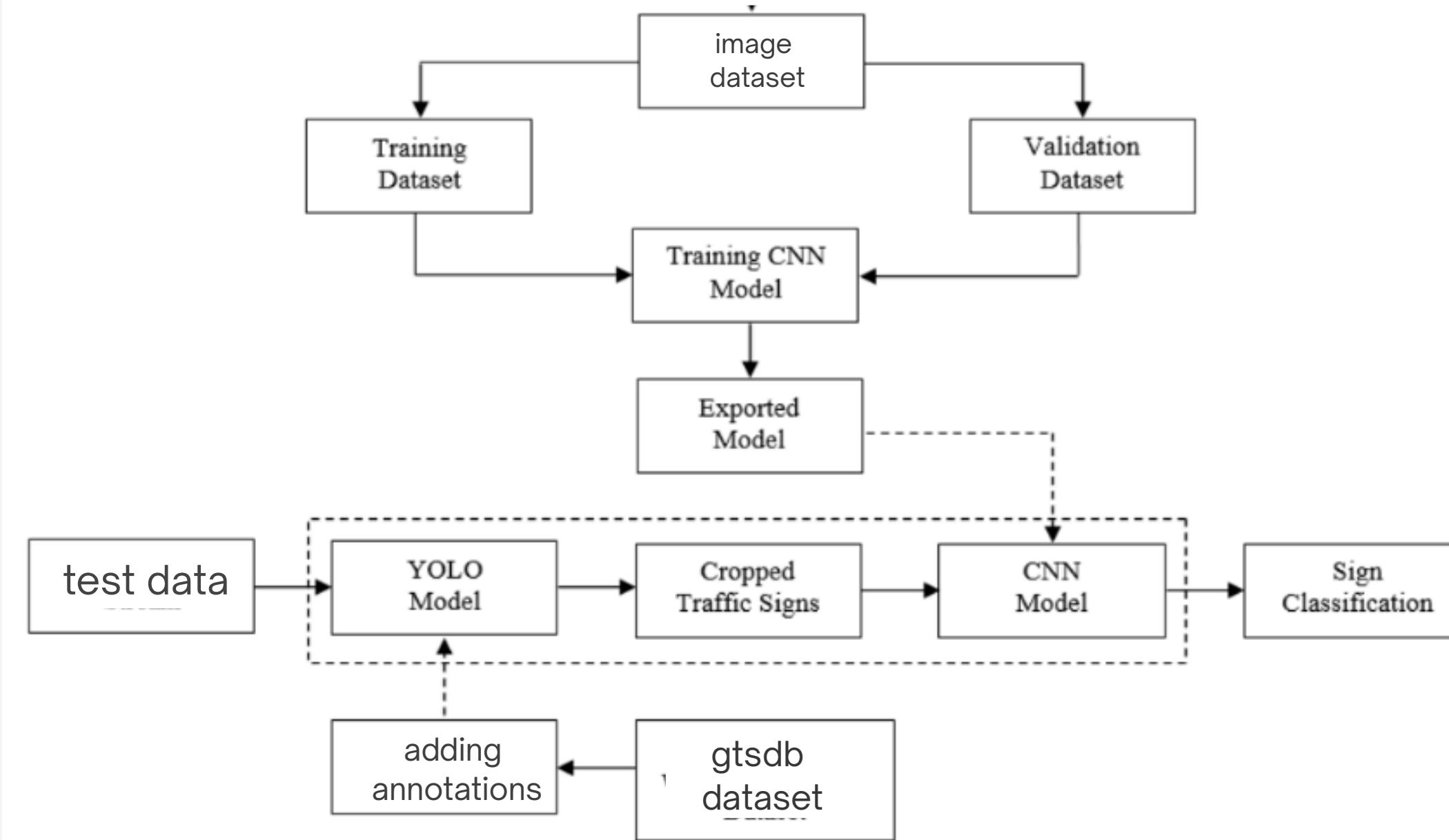


# PROBLEM STATEMENT

- **Objective:** To develop a YOLO and CNN model for detecting and classifying traffic signs using the GTSDB and GTSRB datasets.
- **Detection:** Use object detection techniques (e.g., YOLO) to identify traffic signs in images.
- **Classification:** Classify the detected signs into 43 categories using a CNN-based model.
- **Goal:** Enhance the reliability of intelligent transportation systems and contribute to safer driving through accurate traffic sign recognition.

# WORK FLOW

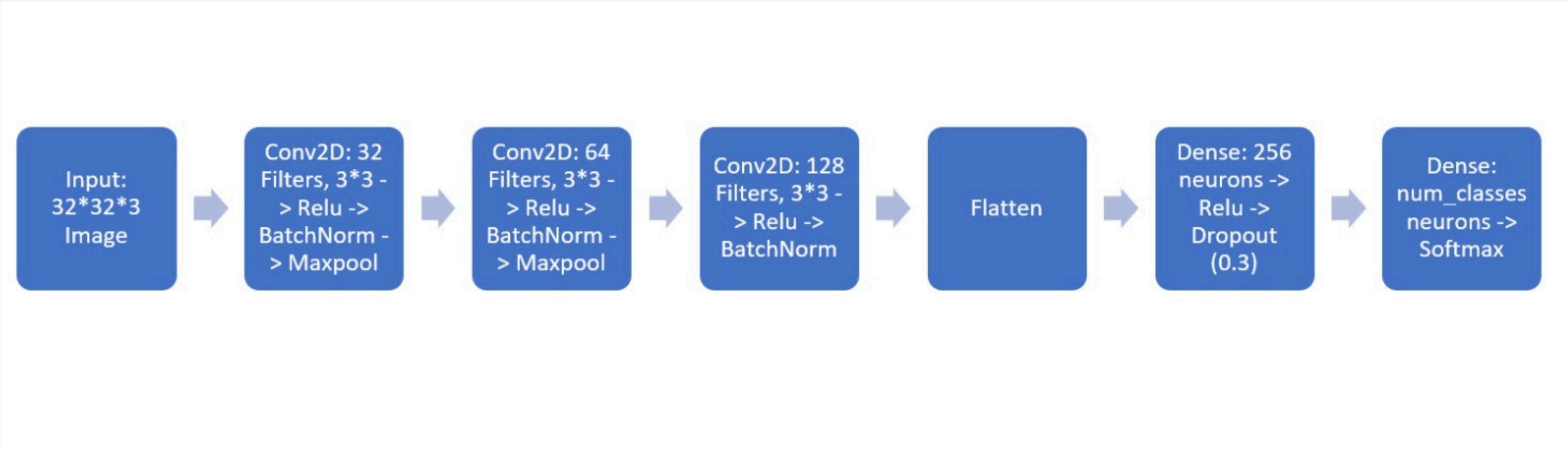
## Brief view of our model



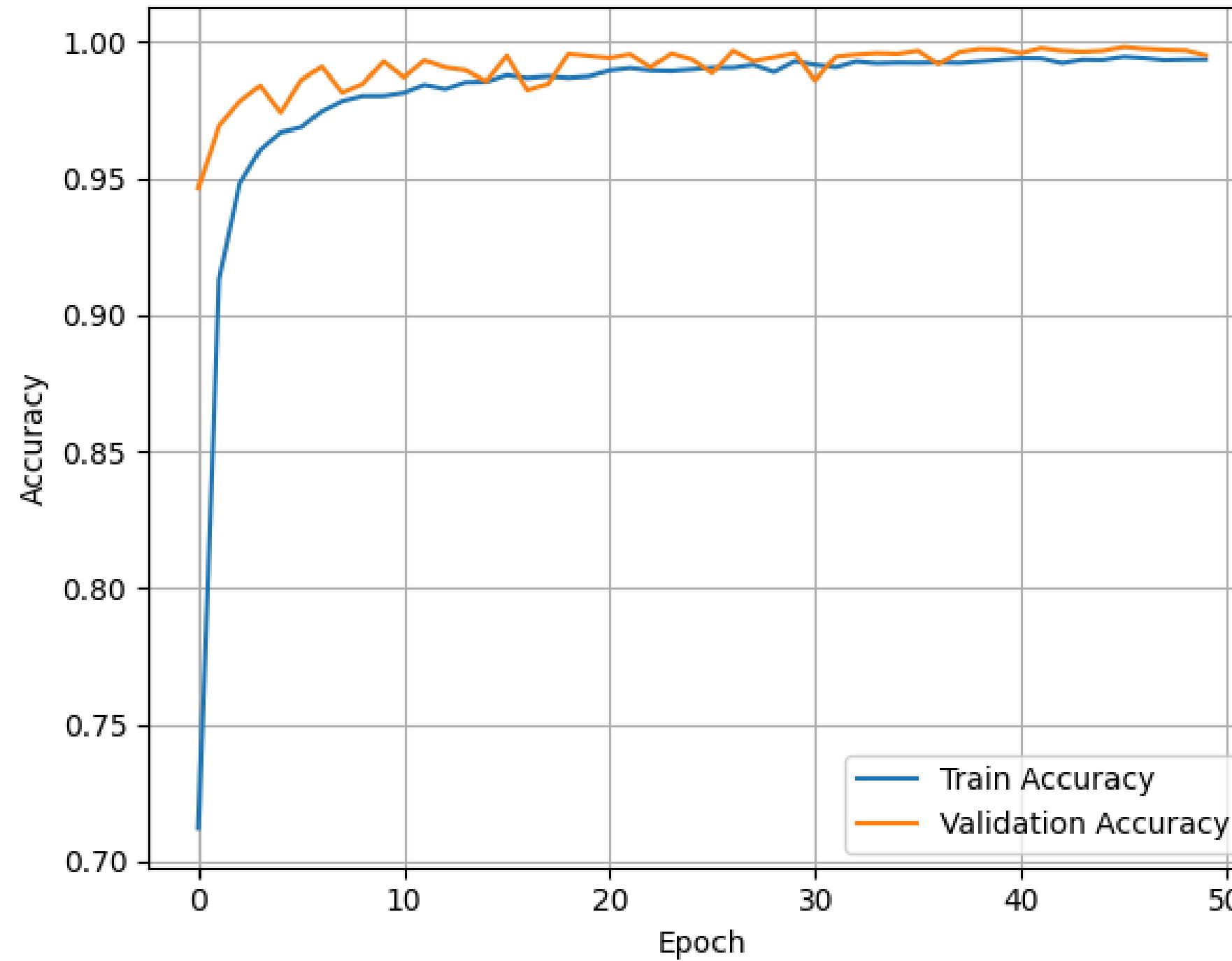
# WHY THIS WORKFLOW

- **YOLOv8 for Detection:** YOLOv8 identifies multiple traffic signs in an image by detecting objects and drawing bounding boxes around them. This model is highly efficient in detecting all relevant signs in a single pass.
- **CNN for Classification:** Once the traffic signs are detected, the regions within the bounding boxes are passed to a CNN. The CNN then classifies the signs into one of 43 categories, ensuring accurate identification of each traffic sign.

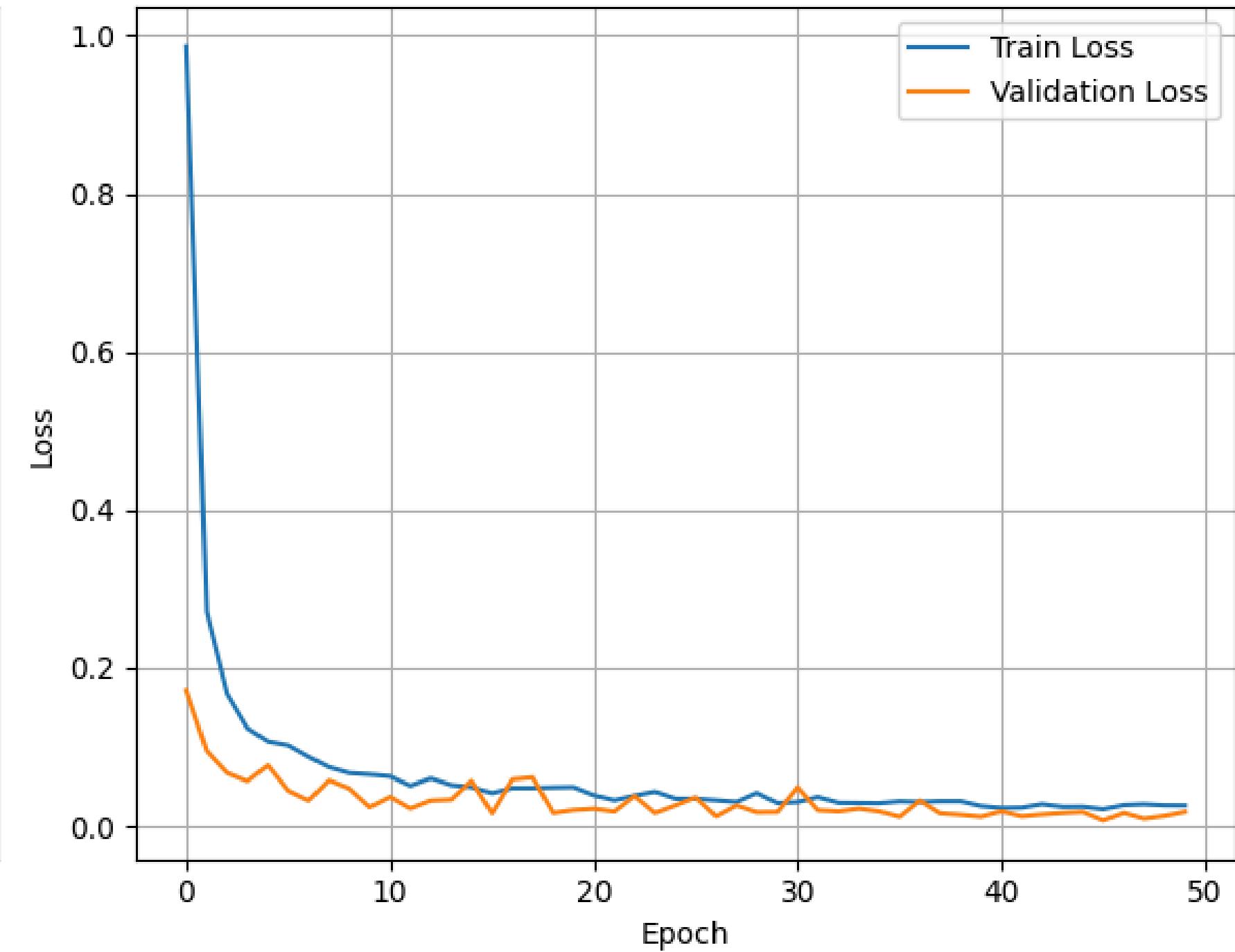
# CNN ARCHITECTURE



### Accuracy over Epochs



### Loss over Epochs



Epoch 46/50

981/981 0s 91ms/step - accuracy: 0.9929 - loss: 0.0269  
True Training Accuracy (no aug): 0.9987 - Loss: 0.0060  
981/981 116s 118ms/step - accuracy: 0.9929 - loss: 0.0269 - val\_accuracy: 0.9968 - val\_loss: 0.0177

Epoch 47/50

981/981 0s 91ms/step - accuracy: 0.9944 - loss: 0.0230  
True Training Accuracy (no aug): 0.9992 - Loss: 0.0026  
981/981 114s 117ms/step - accuracy: 0.9944 - loss: 0.0230 - val\_accuracy: 0.9972 - val\_loss: 0.0111

Epoch 48/50

981/981 0s 90ms/step - accuracy: 0.9937 - loss: 0.0235  
True Training Accuracy (no aug): 0.9989 - Loss: 0.0032  
981/981 115s 117ms/step - accuracy: 0.9937 - loss: 0.0235 - val\_accuracy: 0.9981 - val\_loss: 0.0103

Epoch 49/50

981/981 0s 91ms/step - accuracy: 0.9947 - loss: 0.0198  
True Training Accuracy (no aug): 0.9994 - Loss: 0.0017  
981/981 121s 123ms/step - accuracy: 0.9947 - loss: 0.0198 - val\_accuracy: 0.9981 - val\_loss: 0.0095

Epoch 50/50

981/981 0s 92ms/step - accuracy: 0.9948 - loss: 0.0229  
True Training Accuracy (no aug): 0.9990 - Loss: 0.0033  
981/981 137s 119ms/step - accuracy: 0.9948 - loss: 0.0229 - val\_accuracy: 0.9980 - val\_loss: 0.0063  
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save\_model(model)`. This file format is considered legacy.  
395/395 8s 21ms/step - accuracy: 0.9724 - loss: 0.1787

Test Accuracy: 0.9732

0: 384x640 1 Speed limit (30km/h), 387.7ms

Speed: 5.5ms preprocess, 387.7ms inference, 1.2ms postprocess per image at shape (1, 3, 384, 640)

Detected Traffic Sign Class (CNN): 1

## Detected Signs



0: 384x640 2 No entrys, 433.9ms

Speed: 5.8ms preprocess, 433.9ms inference, 1.3ms postprocess per image at shape (1, 3, 384, 640)

- Detected Traffic Sign Class (CNN): 17
- Detected Traffic Sign Class (CNN): 17

## Detected Signs



