Traffic sign detection and recognition system with Deep Learning

Overview

This project implements traffic sign detection and recognition system using deep learning. The goal is to accurately identify and classify traffic signs from images, which is essential for autonomous driving systems. It leverages TensorFlow and Keras for building and training CNN models (such as CNN with Augmentation, AlexNet, and LeNet). The dataset is extracted from Kaggle.

Project Objectives

- Develop a machine learning model capable of detecting and recognizing traffic signs.
- Compare multiple deep learning architectures to determine the most effective model.
- Improve accuracy through image preprocessing and data augmentation techniques.
- Analyze and evaluate model performance using standard evaluation metrics.
- Provide insights into challenges faced and potential improvements for future work.

Features

- Data loading and preprocessing from CSV files.
- Image augmentation using ImageDataGenerator.
- Model training and evaluation using TensorFlow/Keras.
- **Preprocessing**: Image resizing, normalization, contrast adjustment, Gray scaling and augmentation (rotation, zoom range, height and width shift range)
- **Model architectures**: CNN with Augmentation, LeNet, and a modified AlexNet
- **Evaluation metrics**: Accuracy, Precision, Recall, and F1-score
- **Blur detection**: Laplacian variance-based method to filter blurry images
- **Random Image Testing**: A function selects a random test image and predicts its class
- **Multi-model testing**: The system can test images on multiple trained models
- **Edge detection**: Canny method to detect edges.

Requirements

Ensure you have the following dependencies installed:

```
pip install numpy pandas matplotlib seaborn tensorflow scikit-learn opencv-python
```

Dataset

The model is trained using the **German Traffic Sign Recognition Benchmark (GTSRB)** dataset, which contains:

Link: https://www.kaggle.com/datasets/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign

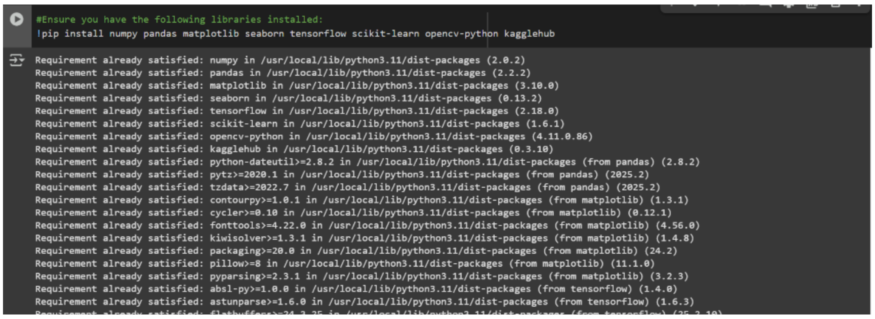
- **39,209** training images
- **12,630** validation images
- **12,569** test images

- **43** different traffic sign classes
- Metadata files: Train.csv and Test.csv.
- Images are preprocessed for training.

Running the Notebook

1. Install the required Dependencies :-

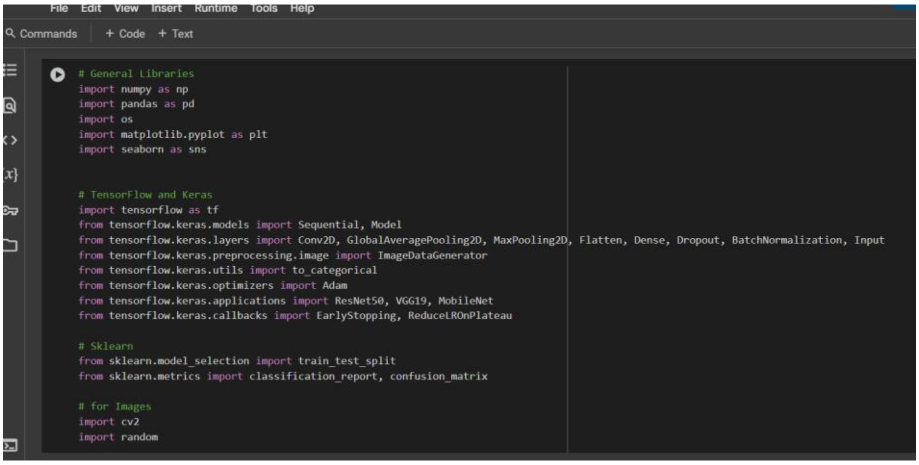
!pip install numpy pandas matplotlib seaborn tensorflow scikit-learn opencv-python kagglehub



```
#Ensure you have the following libraries installed:
!pip install numpy pandas matplotlib seaborn tensorflow scikit-learn opencv-python kagglehub
```

```
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packages (2.0.2)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: seaborn in /usr/local/lib/python3.11/dist-packages (0.13.2)
Requirement already satisfied: tensorflow in /usr/local/lib/python3.11/dist-packages (2.18.0)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.11/dist-packages (1.6.1)
Requirement already satisfied: opencv-python in /usr/local/lib/python3.11/dist-packages (4.11.0.86)
Requirement already satisfied: kagglehub in /usr/local/lib/python3.11/dist-packages (0.3.10)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.3.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (4.56.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (24.2)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (11.1.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (3.2.3)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=24.3.25 in /usr/local/lib/python3.11/dist-packages (from tensorflow) (25.2.10)
```

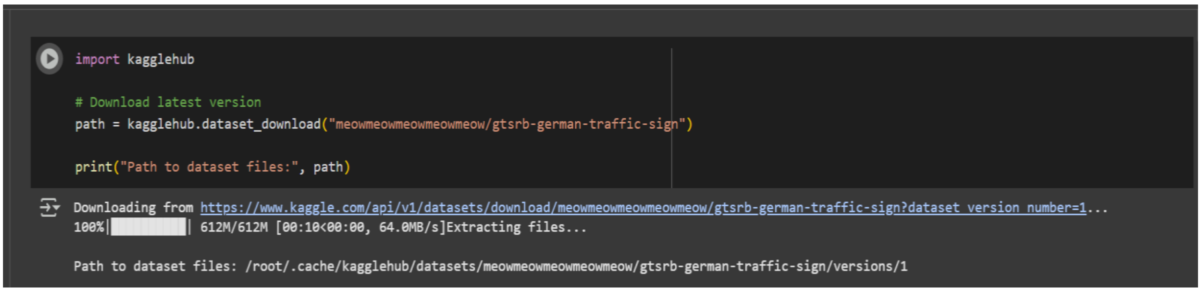
2. Import Require Libraries



```
# General Libraries
import numpy as np
import pandas as pd
import os
import matplotlib.pyplot as plt
import seaborn as sns


# TensorFlow and Keras
import tensorflow as tf
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.layers import Conv2D, GlobalAveragePooling2D, MaxPooling2D, Flatten, Dense, Dropout, BatchNormalization, Input
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.applications import ResNet50, VGG19, MobileNet
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau

# Sklearn
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, confusion_matrix

# for Images
import cv2
import random
```

3. Download and Import Dataset

Run this snippet code to download dataset directly from kaggle.



```
import kagglehub

# Download latest version
path = kagglehub.dataset_download("meowmeowmeowmeowmeow/gtsrb-german-traffic-sign")

print("Path to dataset files:", path)
```

```
Downloading from https://www.kaggle.com/api/v1/datasets/download/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign?dataset_version_number=1...
100%|██████████| 612M/612M [00:10<00:00, 64.0MB/s]Extracting files...

Path to dataset files: /root/.cache/kagglehub/datasets/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign/versions/1
```

4. Pre-Processing

Loads an image, converts it to RGB, resizes, normalizes, and reshapes it for model input.

i. Image resizing

ii. Normalization of image

iii. Gray scaling of image

iv Contrast enhancement of image

5. Implementation

Model training:

i. Cnn Model
ii. LeNet Model
iii. AlexNet Model

RUN ALL THE CODE IN THE G16_Code.ipynb TO GET THE MODEL PERFORMANCE.

Model Performance

Model	**Accuracy**
CNN with Augmentation	96.90%
LeNet	94.11%
Modified AlexNet	95.27%