VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Presentation on Road Sign Detector and Driver Alert System

Presented By

Aditya Panditrao (22MCB0032)

Sarthak Tripathi (22MCB0030)

Guided By

Prof. Dr.Yokesh Babu S

Date -23/01/2023

Contents

- Domain
- List of Issues
- Issues to be focused
- Introduction
- Literature Survey
- Problem Statement
- Proposed Solution
- Methodology
- Experiments and Results
- Conclusion and Future Work
- References

Domain

- Road Safety: This project will focus driver’s safety using Image processing techniques.
- Image Pre-Processing
- Deep Learning: The deep learning algorithm CNN is used due to its ability to recognize patterns in images.

List of Issues

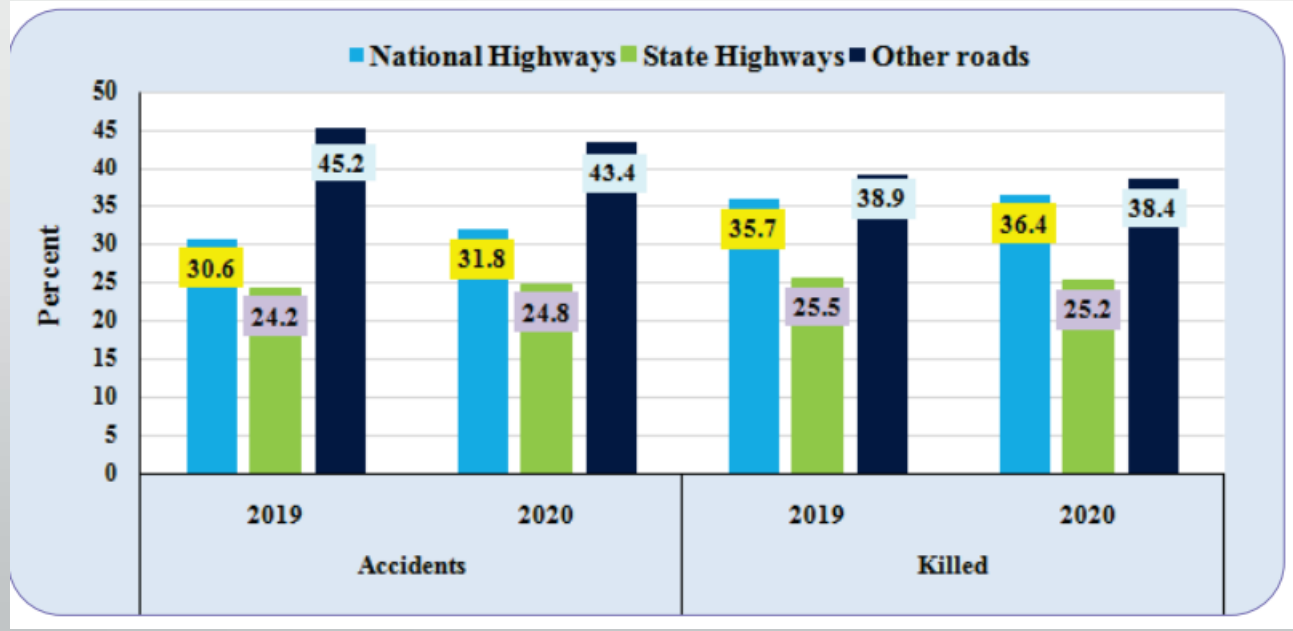
- Bad lighting conditions make it harder to gather colour and contour information of traffic sign.
- Occluded Traffic sings with sunlight and dark.
- Motion blur and car vibration makes harder to retrieve images from the camera.
- Damaged traffic sign.

Issues to be focused

- Identify, categorize and warn to the driver for traffic signs.
- Safety of the drivers and the fellow travelers
- Clearer Diagnosis of road signs

Road Accident Statistics

During the calendar year 2020, there were close to 5lakh road accident in India, which resulted in more than 1.3lakhs death and inflicted injured on 5.2lakh persons. These numbers translate into one road accident every minute and a road accident death every 4 minute. More than half of these victims are in active age group of 25-65 years. Following graph gives percentage of the total number of accident took place and people killed from 2019-2020.



Introduction

Road traffic has been the major issue that most cities are facing these days. People neglect the importance of traffic signs which could prove deadly. Currently, the maps services assist us about some details of navigation, but it fails to provide essential information such as “U-turns, prohibited roads, speed-limits, speed breakers and diversions etc.”, which are generally present on the signboards. These are of utmost importance for road safety but the drivers often miss out on them. There is urgent need for a mechanism which would be able to detect these signs and alert the drivers. So, an expert system based on CNN, will detect & recognize the signboards in real time.

- Traffic sign recognition system is a crucial research direction in computer vision and a significant section of Advanced Driver Assistance System (ADAS). It can be grouped into two technologies, traffic-sign detection and traffic-sign recognition. The correctness of detection will directly lead to the final identification results. Traffic signs contain necessary messages about vehicle safety and they show the latest traffic conditions, define road rights, forbid and allow some behaviors and driving routes, cue dangerous messages and so on. They can also help drivers identify the condition of the road, so as to determine the driving routes.
- Traffic signs have some constant characteristics that can be used for detection and classification, among them, color and shape are important attributes that can help drivers obtain road information. The image of traffic signs is often affected by some external factors, such as weather conditions. Therefore, traffic-sign recognition is a challenging subject and also a valuable subject in traffic engineering research.

Literature Survey

Author & Year	Adopted Scheme	Features	Challenges
Md Tarequl Islam (2020)	Two Convolutional Neural Network using Classifier.	HSV Color space is used, RGB to HSV conversation, Moderate to high accuracy.	Ability to label the signs, manually clicked pictures & online downloaded. Classifier Challenges.
V.KOKILA, Mr. N.VASUDEVAN(2020)	(ADS) auto driving frameworks. Sign based strategies, Shape-based techniques.	Advanced machine learning Classification. Image pre-processing methods.	Image quality, Non-informational Pixel, Unwanted picture edges.
Chai K. Toh (2018)	Wireless digital traffic signs	Wireless Transmission of traffic signs Reduce complex image recognition	Positioning of Traffic Signs Cost of Installation is high
Ying Sun (2019)	Convolutional Neural Network	Better Image recognition Minimal Error	Need Consideration on climatic factors.
Arturo de la Escalera(1997)	Image processing, Neural networks, traffic signs recognition Using Neural Network Model, Keras.	Accuracy of image recognition.	The highly probable sign regions are extracted. The most preferred are "pure" colors for better visibility.

Problem Statement

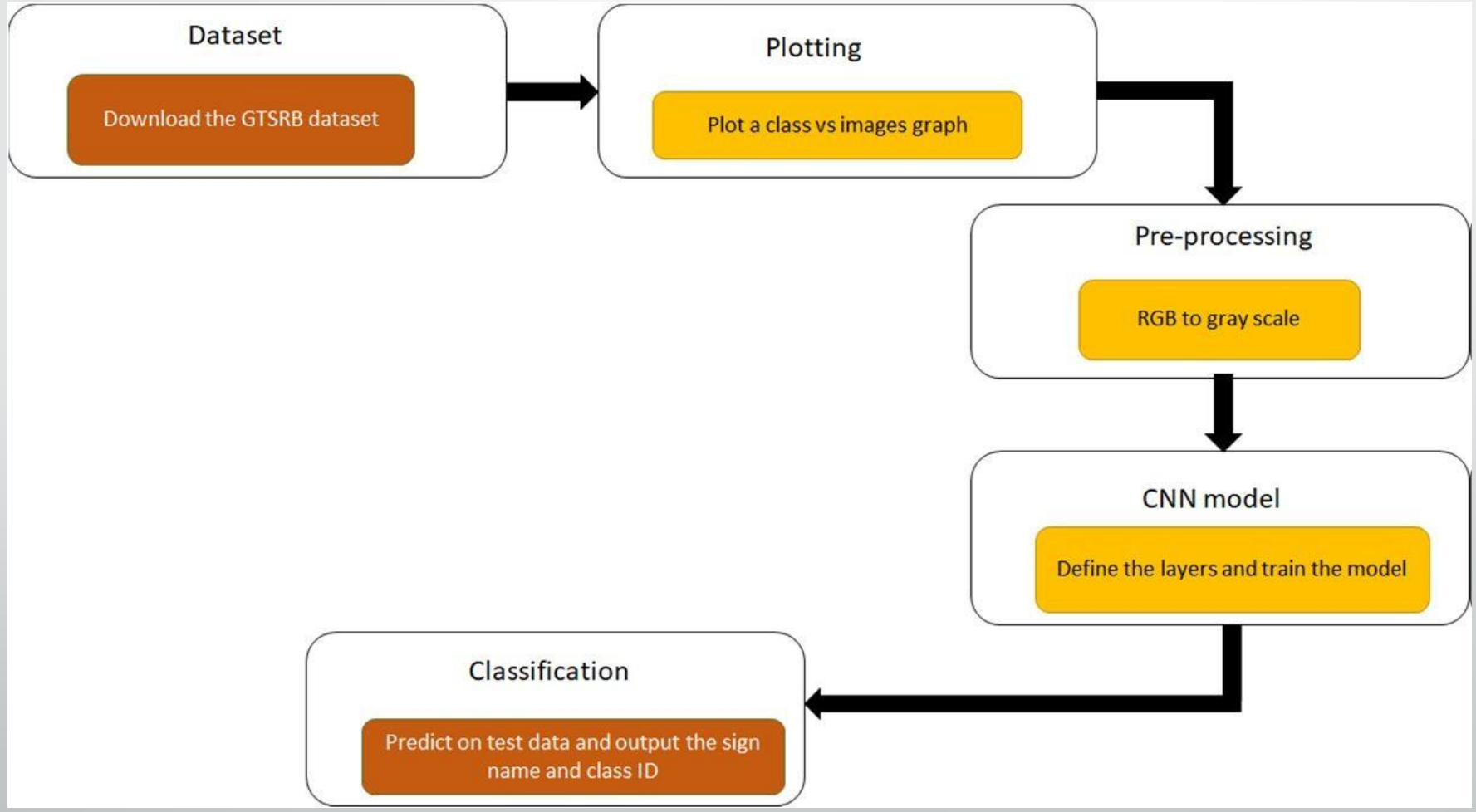
The objectives of the system is first to detect and then classify signs. This works well when we have lots of images as well as not clear ones.

For classifying images we are using Convolutional Neural Network Model

Why CNN is better than other Models

- CNN are hugely popular is because of their architecture — the best thing is there is no need for feature extraction. The system learns to do feature extraction and the core concept of CNN is, it uses convolution of image and filters to generate invariant features which are passed on to the next layer. The features in next layer are convoluted with different filters to generate more invariant and abstract features and the process continues till one gets final feature / output (let say face of X) which is invariant to occlusions.
- another key feature is that deep convolutional networks are flexible and work well on image data. As one researcher points out, convolutional layers exploit the fact that an interesting pattern can occur in any region of the image, and regions are contiguous blocks of pixels. But one of the reasons why researchers are excited about deep learning is the potential for the model to learn useful features from raw data. Now, convolutional neural networks can extract informative features from images, eliminating the need of traditional manual image processing methods.

Proposed Solution



Methodology

- Explore the Dataset
- The presence of a generalised dataset is crucial before moving on to the detection or classification.
- The GTSRB (German Traffic Sign Recognition Benchmark) dataset is the most popular of them.
- There are numerous images in it.
- The variety, backdrop, and colour variation of the traffic signs will aid in the model's performance accuracy.
- For this project, we are using the public dataset available at Kaggle:

Traffic sign Dataset

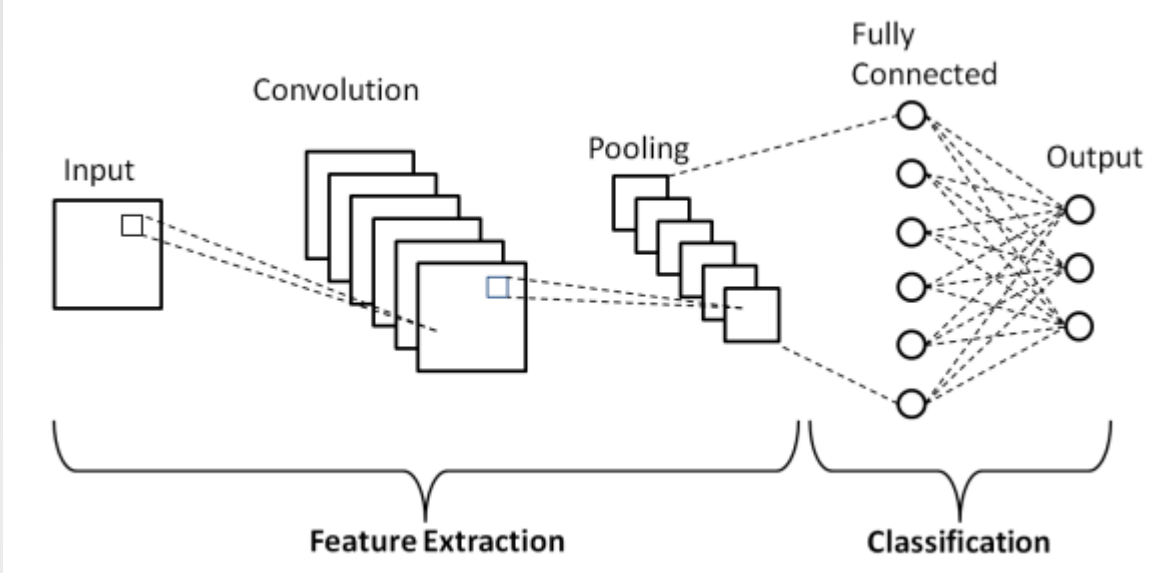
- The dataset contains more than 50,000 images of different traffic signs. It is further classified into 43 different classes the dataset is quiet varying, some of the classes have many images while some classes have few images. The size of the dataset is around 300MB. The dataset has a train folder which contains images inside each class and a test folder which we will use for testing our model.

- Pre-Processing

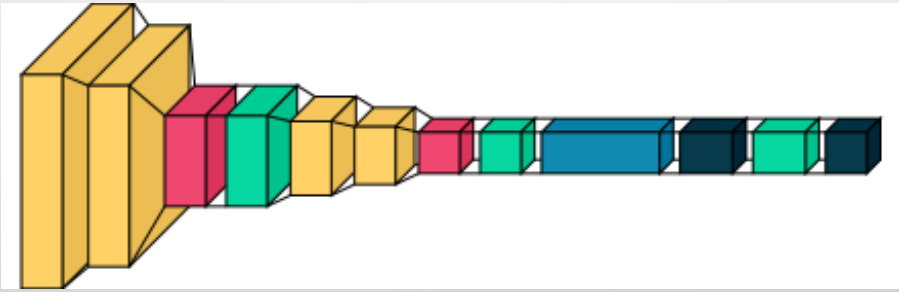
```
for i in range(classes):
    path = os.path.join(cur_path,'train',str(i))
    images = os.listdir(path)
    for a in images:
        try:
            image = Image.open(path + '\\'+ a)
            image = image.resize((30,30))
            image = np.array(image)
            data.append(image)
            labels.append(i)
        except Exception as e:
            print(e)
```

✓ 10m 14.8s

```
cannot identify image file 'D:\\Traffic_Sign_Recognition\\train\\0\\archive - Shortcut.lnk'
```



Schematic diagram of a basic convolutional neural network (CNN) architecture.

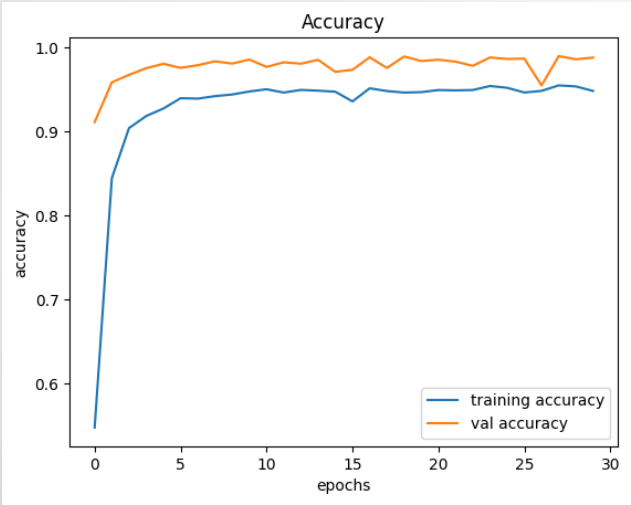


Visualization CNN model with its layers

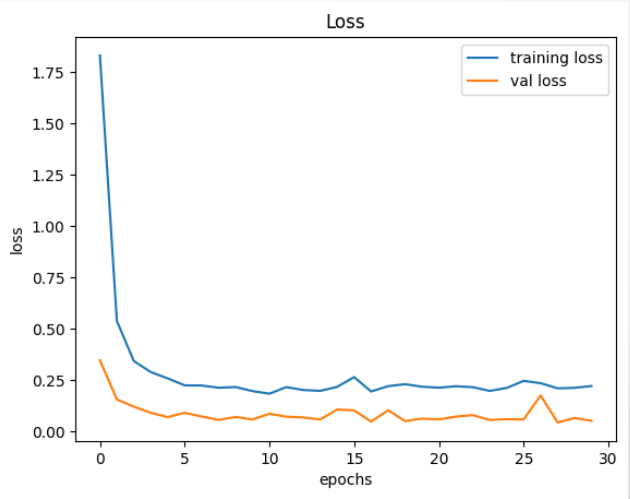
- Test the model with test dataset

```
def testing(testcsv):
    y_test = pd.read_csv(testcsv)
    label = y_test["ClassId"].values
    imgs = y_test["Path"].values
    data=[]
    for img in imgs:
        image = Image.open(img)
        image = image.resize((30,30))
        data.append(np.array(image))
    X_test=np.array(data)
    return X_test,label
```

Accuracy and Loss Graph for testing data:



Accuracy



Loss

- Accuracy Score for testing Data

```
# Score
score = model.evaluate(X_test, y_test, verbose=0)
print('Test Loss', score[0]*100)
print('Test accuracy', score[1]*100)
```

✓ 8.1s

```
Test Loss 4.988387227058411
Test accuracy 98.81407618522644
```

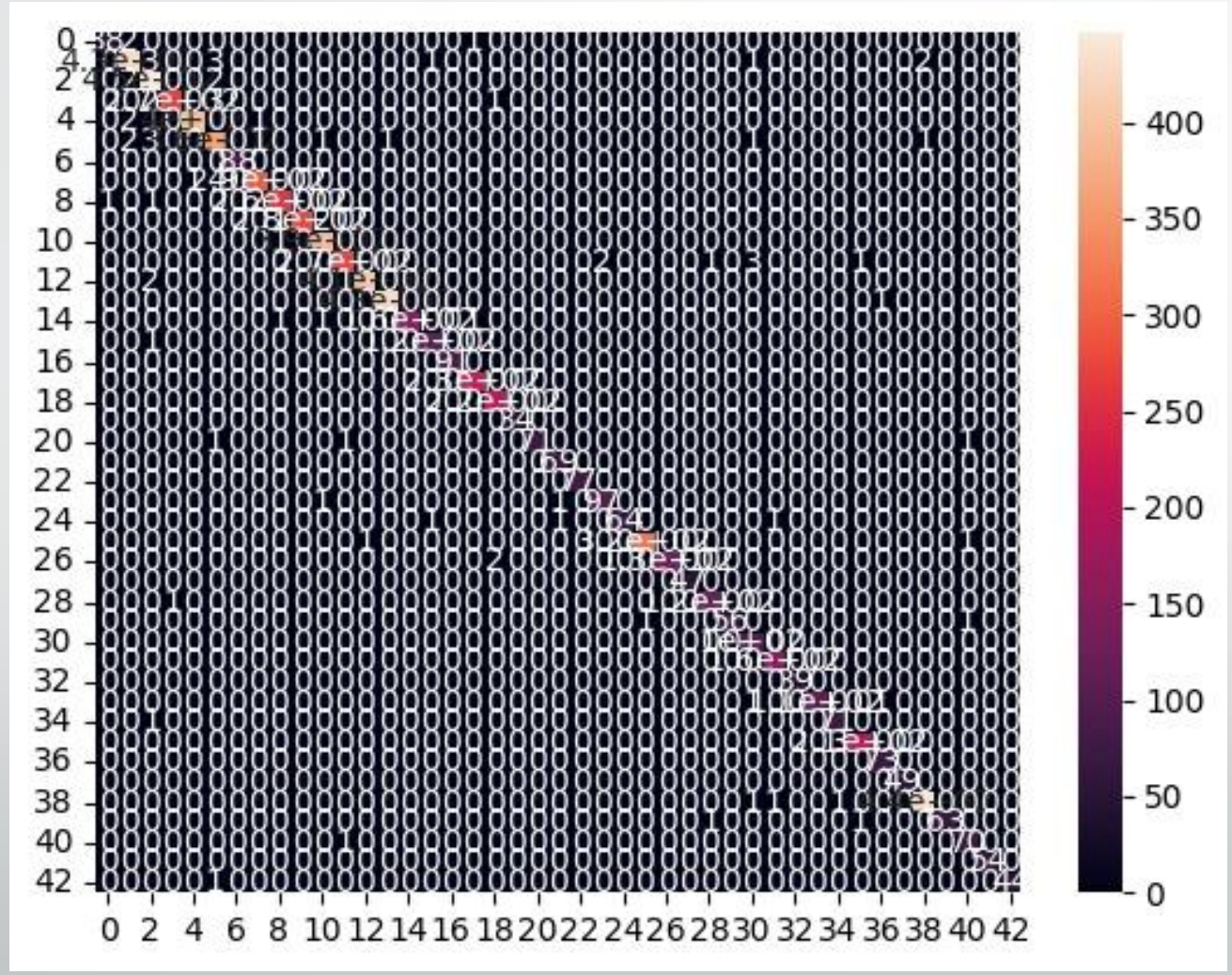
Experiments

- Building the front-end for the Model

```python
from flask import *
import os
from werkzeug.utils import secure_filename
from keras.models import load_model
import numpy as np
from PIL import Image

app = Flask(__name__)

# Classes of trafic signs
classes = { 0:'Speed limit (20km/h)',
            1:'Speed limit (30km/h)',
            2:'Speed limit (50km/h)',
            3:'Speed limit (60km/h)',
            4:'Speed limit (70km/h)',
            5:'Speed limit (80km/h)',
            6:'End of speed limit (80km/h)',
            7:'Speed limit (100km/h)',
            8:'Speed limit (120km/h)',
            9:'No passing',
            10:'No passing veh over 3.5 tons',
            11:'Right-of-way at intersection',
            12:'Priority road',
            13:'Yield',
            14:'Stop',
            15:'No vehicles',
            16:'Vehicle > 3.5 tons prohibited',
            17:'No entry',
            18:'General caution',
            19:'Dangerous curve left',
            20:'Dangerous curve right',
            21:'Double curve',
            22:'Bumpy road',
            23:'Slippery road',
            24:'Road narrows on the right',
            25:'Road work',
            26:'Traffic signals',
            27:'Pedestrians',
            28:'Children crossing',
            29:'Bicycles crossing',
            30:'Beware of ice/snow',
            31:'Wild animals crossing',
            32:'End speed + passing limits',
            33:'Turn right ahead',
            34:'Turn left ahead',
            35:'Ahead only',
            36:'Go straight or right',
            37:'Go straight or left',
            38:'Keep right',
            39:'Keep left',
            40:'Roundabout mandatory',
            41:'End of no passing',
            42:'End no passing vehicle > 3.5 tons' }
```

```python
model = load_model('model/TSR2.h5')

def test_on_img(img):
    data=[]
    image = Image.open(img)
    image = image.resize((30,30))
    data.append(np.array(image))
    X_test=np.array(data)
    Y_pred = model.predict(X_test)
    return image,Y_pred

def image_processing(img):
    plot,prediction = test_on_img(img)
    s = [str(i).strip().replace("\n","") for i in prediction][0][1:-1]
    s=[float(b) for b in s.split(" ")]
    a=[[index, num] for index, num in enumerate(s) if (num == max(s))][0]
    # print(type(a),type(a[0]), a)
    # print(type(s),type(s[0]), s)
    # a = int("".join(s))
    return [a[1], classes[a[0]]]

@app.route('/', methods=['GET', 'POST'])
def index():
    return render_template('index.html')

@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        # Get the file from post request
        f = request.files['file']
        f.save(secure_filename(f.filename))
        file_path = os.path.abspath(secure_filename(f.filename))
        # Make prediction
        result = image_processing(file_path)
        result = '{0} ({1:02} %)'.format(result[1], round((result[0] * 100), ndigits=3))
        os.remove(file_path)
        return result
    return None

if __name__ == '__main__':
    app.run(debug=True)
```

Model Evaluation through Confusion Matrix



Result

The trained CNN's test results reveal that the model is 98.81% accurate at identifying and detecting traffic signs. The experimental finding demonstrates the CNN model's strong recognition accuracy for detecting and identifying road signs.

Conclusion and Future Work

- This project presents a method for recognizing traffic signs using deep learning. This approach is efficient in identifying and recognizing traffic signs by utilizing image pre-processing, road sign detection, and classifying. Results of the testing show that the accuracy of this method is 98.81%.
- In future, we will keep extending our datasets to cover all classes of our traffic signs. Meanwhile, more newly developed models for visual object recognition, such as Mask R-CNN, CapsNet, and Siamese neural network would be included. Capsule neural network (CapsNet) has been employed for effectively identifying a class of traffic signs which have spatial relationships. Compared with the well-known deep neural networks, capsule networks tackle the topological relationship between visual objects. In addition, we will adopt professional evaluation metrics to assess the performance of our models from multiple aspects in future.

References

1. Chen T, Lu S, “Accurate and Efficient Traffic Sign Detection Using Discriminative Adaboost and Support Vector Regression,” IEEE Transactions on Vehicular Technology, 2016, 65(6):4006-4015.
2. https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c
3. J. Crissman and C. E. Thorpe “UNSCARF, a colour vision system for the detection of unstructured roads,” in Proc. IEEE Int. Conf. Robotics and Automation, Sacramento, CA, Apr. 1991, pp. 2496–2501.
4. https://www.carandbike.com/news/traffic-sign-recognition-function-and-intelligence-2963101
5. Who, “Global status report on road safety,” Who, Geneva, Switzerland, 2018.
6. https://towardsdatascience.com/using-convolutional-neural-network-for-image-classification.
7. https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/
8. https://www.kaggle.com/datasets/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign.
9. https://pyimagesearch.com/2019/11/04/traffic-sign-classification-with-keras-and-deep-learning/