Traffic sign Classification

- PRESENTED BY
- 18BCE027 - SAHIL BHINGRADIA
- 18BCE154 -BHOOMI PATEL

Introduction

- Traffic Sign Classification is the most challenging problem for many real-world applications like Autonomous vehicles, ADAS, Sign monitoring for maintenance, etc.
- Some factors such as color fading, weather conditions, undesirable backgrounds, poor visibility because of lightning, shadows, blurred images because of the speed of the vehicle make the task more challenging.
- For an autonomous vehicle, correct classification of the traffic signs must be required.
- any autonomous vehicle must be able to operate safely in the existing environment, it also needs human-level visual recognition of the environment in which it operates. Visual Classification is the most complex task in the domain of visual recognition.

Continued....

- That’s why the Traffic Sign Classification is the most important integral part of the autonomous vehicle
- Most important thing is that we have to create model which will classify the sign in correct way.
- To do this classification task everywhere Deep Learning Networks are used
- We proposed a deep neural network for traffic sign classification using Deep Learning techniques and the German Traffic Sign Recognition Benchmark - GTSRB dataset which performed very well in classification.

Problem Statement

- We used GTSRB dataset for this project. Our challenge is to identify and classify the images of the traffic signs into 43 different classes using Deep Learning Convolutional Neural Network.
- Preprocess the data using normalization and data augmentation techniques to remove the unfavorable features of the images.

Objective

Our dataset consist the image of extracting from the video under different condition like illumination, weather, speed, day, night, etc. The following objects are expected from the model.

- Remove the effects of the illumination
- Scaling and Normalization
- Data augmentation techniques like zoom, rotation, shear, brightness disturbance, Gaussian noise, color inversion, etc.
- Train the CNN model on dataset
- Classify the images into the appropriate class

Examples

Dangerous curve to the left

Turn left ahead

Yield

Stop

Examples

No entry

Speed limit
(50km/h)

General caution

Go straight or right

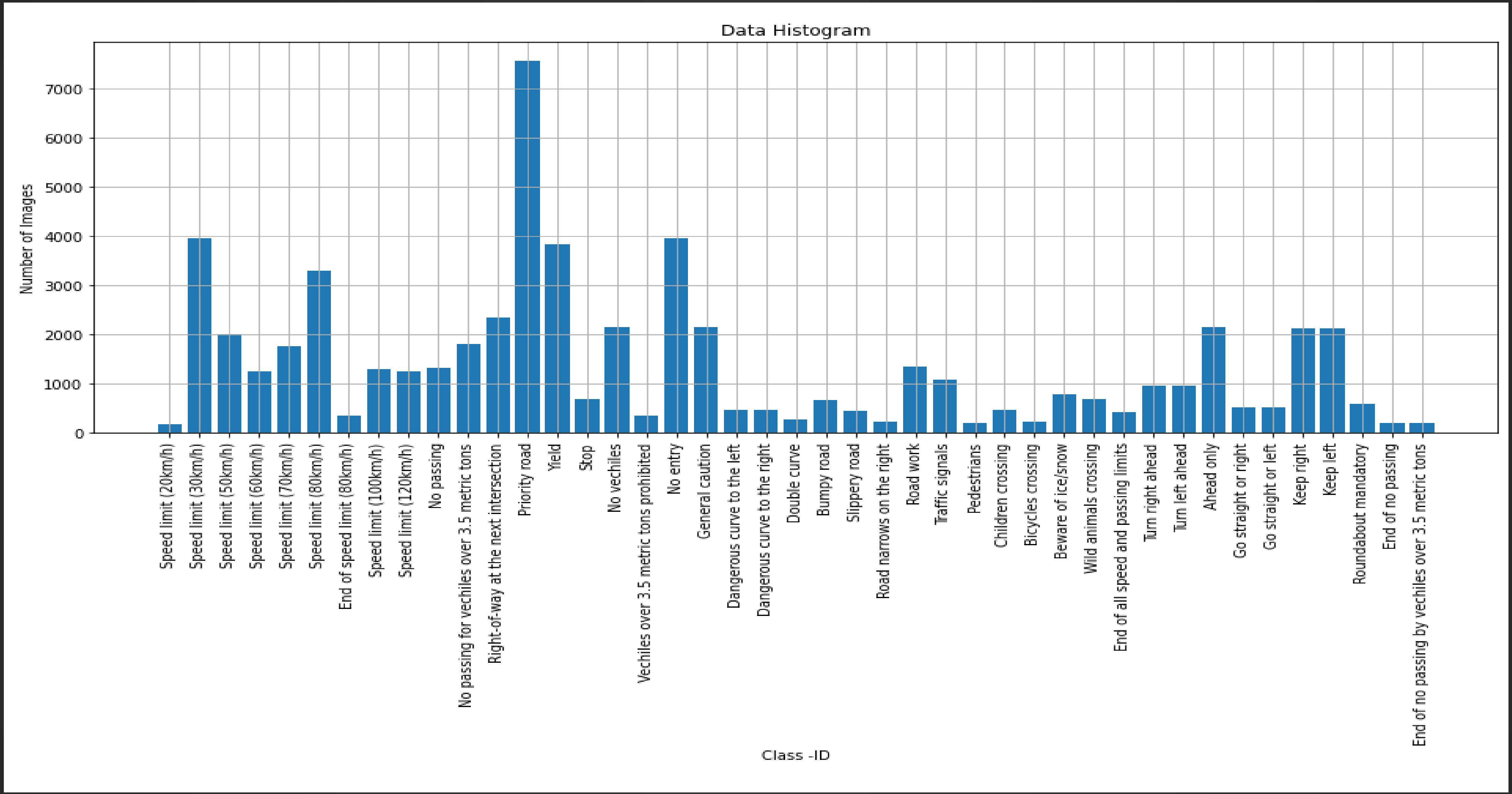
Methodology

Dataset

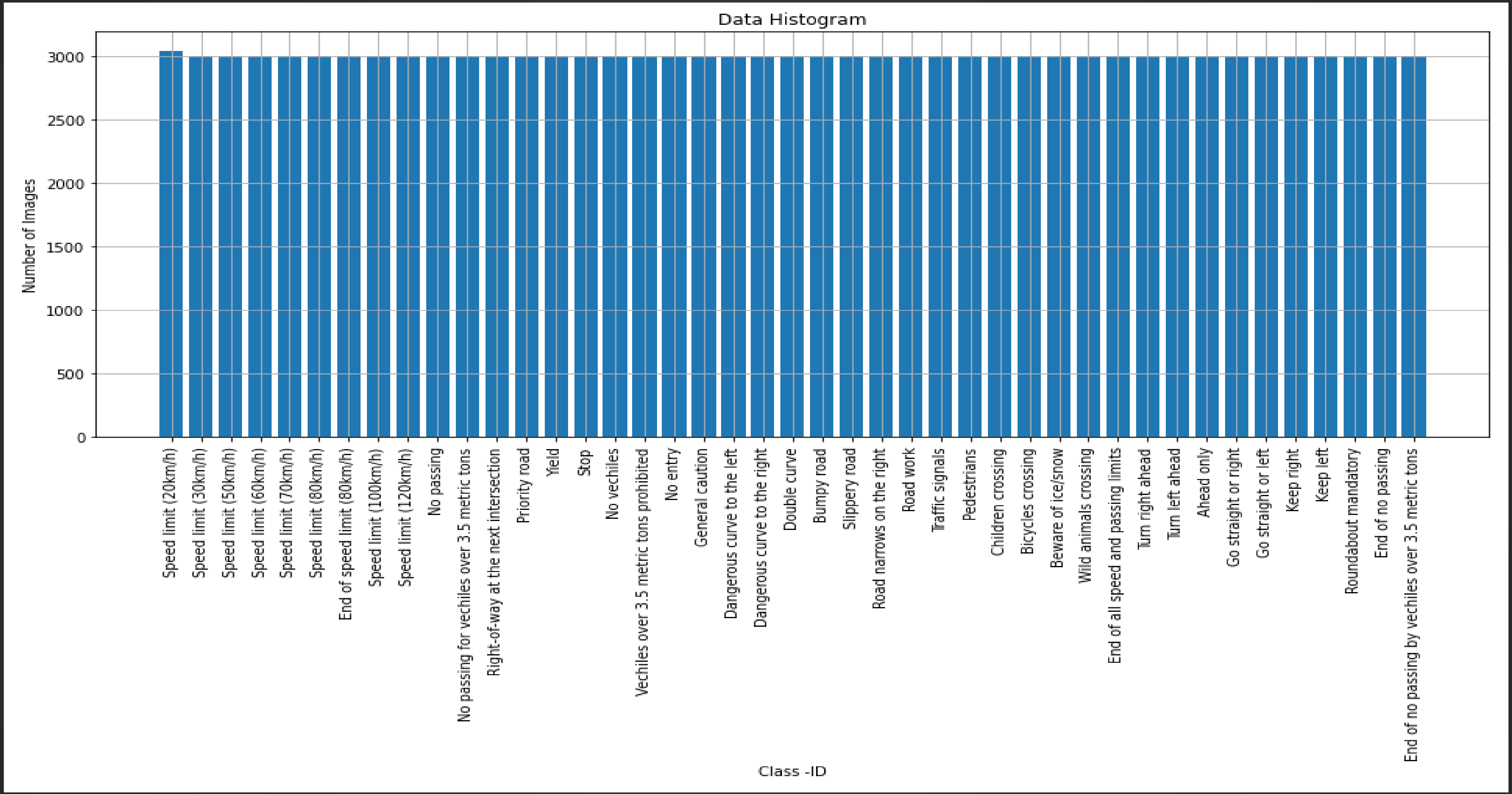
- GTSRB dataset is a multi-class, single-image classification challenge held at the International Joint Conference on Neural Networks 2011.
- It consists of images of the traffic signs taken from the German roads and a class label for each image
  - 43 different classes
  - More than 50,000 Images in total
  - consist of the class label (between 0 to 42) and sign name for each class
  - We divide the whole dataset into Training, Validation, and Testing Images
  - Each image is 32 x 32 pixels made up of 3 color channels formatted in RGB.

Data Augmentation

- we can see that the dataset is very unbalanced. Some classes are significantly better than others.
- we plot 10 random images of the same class. The images are significantly different in terms of contrast and brightness, so we need to do the histogram equalization for extracting good features from the image.
- Following slide show the histogram before and after Data Augmentation.



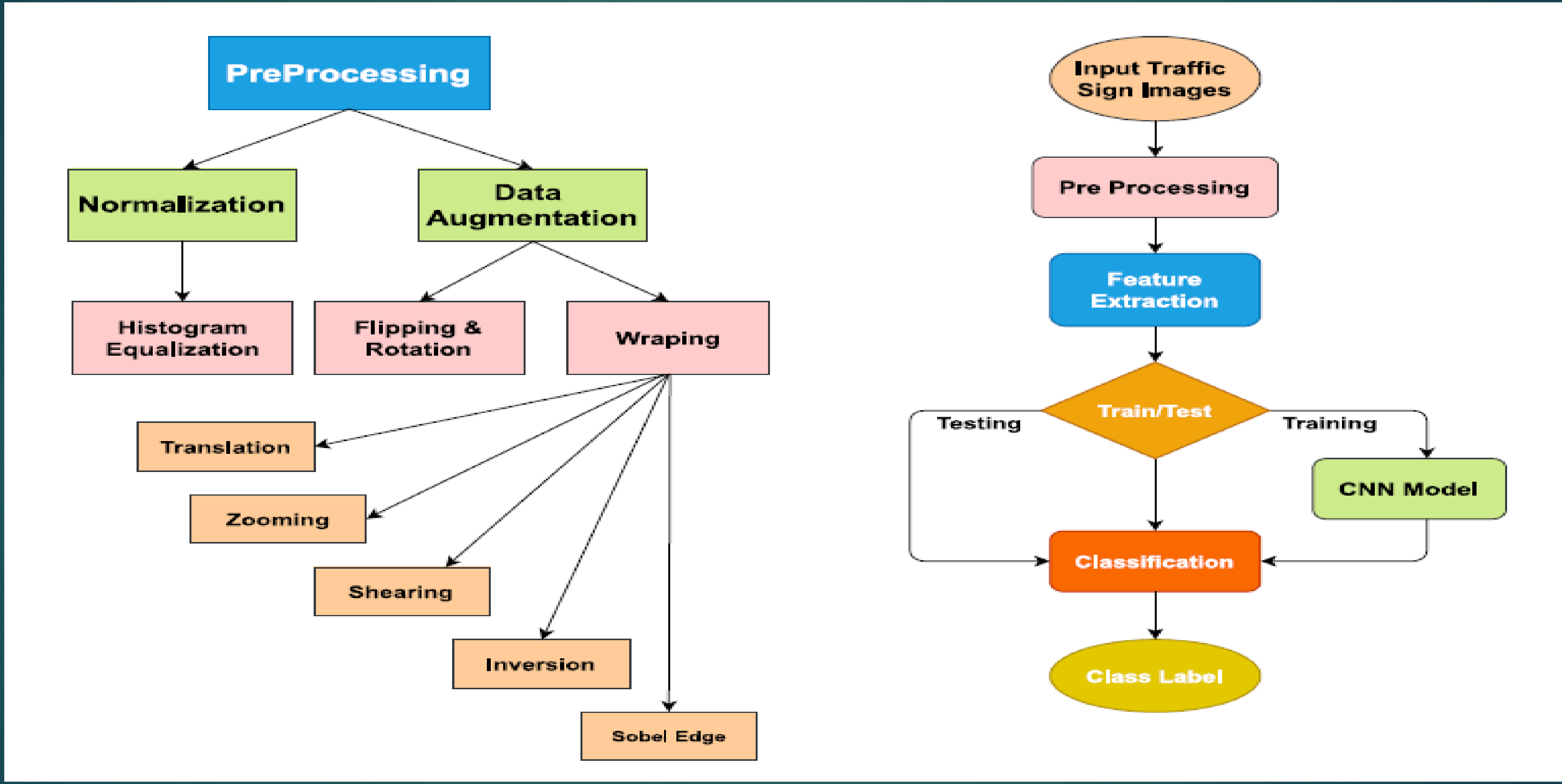
Data Histogram
Number of Images
0
1000
2000
3000
4000
5000
6000
7000
Speed limit (20km/h)
Speed limit (30km/h)
Speed limit (50km/h)
Speed limit (60km/h)
Speed limit (70km/h)
Speed limit (80km/h)
End of speed limit (80km/h)
Speed limit (100km/h)
Speed limit (120km/h)
No passing
No passing for vechiles over 3.5 metric tons
Right-of-way at the next intersection
Priority road
Yield
Stop
No vechiles
Vechiles over 3.5 metric tons prohibited
No entry
General caution
Dangerous curve to the left
Dangerous curve to the right
Double curve
Bumpy road
Slippery road
Road narrows on the right
Road work
Traffic signals
Pedestrians
Children crossing
Bicycles crossing
Beware of ice/snow
Wild animals crossing
End of all speed and passing limits
Turn right ahead
Turn left ahead
Ahead only
Go straight or right
Go straight or left
Keep right
Keep left
Roundabout mandatory
End of no passing
End of no passing by vechiles over 3.5 metric tons
Class -ID


Data Histogram
Number of Images
3000
2500
2000
1500
1000
500
0
Speed limit (20km/h)
Speed limit (30km/h)
Speed limit (50km/h)
Speed limit (60km/h)
Speed limit (70km/h)
Speed limit (80km/h)
End of speed limit (80km/h)
Speed limit (100km/h)
Speed limit (120km/h)
No passing
No passing for vechiles over 3.5 metric tons
Right-of-way at the next intersection
Priority road
Yield
Stop
No vechiles
Vechiles over 3.5 metric tons prohibited
No entry
General caution
Dangerous curve to the left
Dangerous curve to the right
Double curve
Bumpy road
Slippery road
Road narrows on the right
Road work
Traffic signals
Pedestrians
Children crossing
Bicycles crossing
Beware of ice/snow
Wild animals crossing
End of all speed and passing limits
Turn right ahead
Turn left ahead
Ahead only
Go straight or right
Go straight or left
Keep right
Keep left
Roundabout mandatory
End of no passing
End of no passing by vechiles over 3.5 metric tons
Class -ID

Preprocessing Methods

- Flipping
- Rotation
- Translation
- Zoom
- Histogram Equalization
- Normalization
- Shuffling
- Shearing
- Invert
- Sobel Edge
- Projection Transformation
- Gaussian Noise

Proposed Model



Generated Augmented Images

Original	Augmented Intensity = 0.75
	

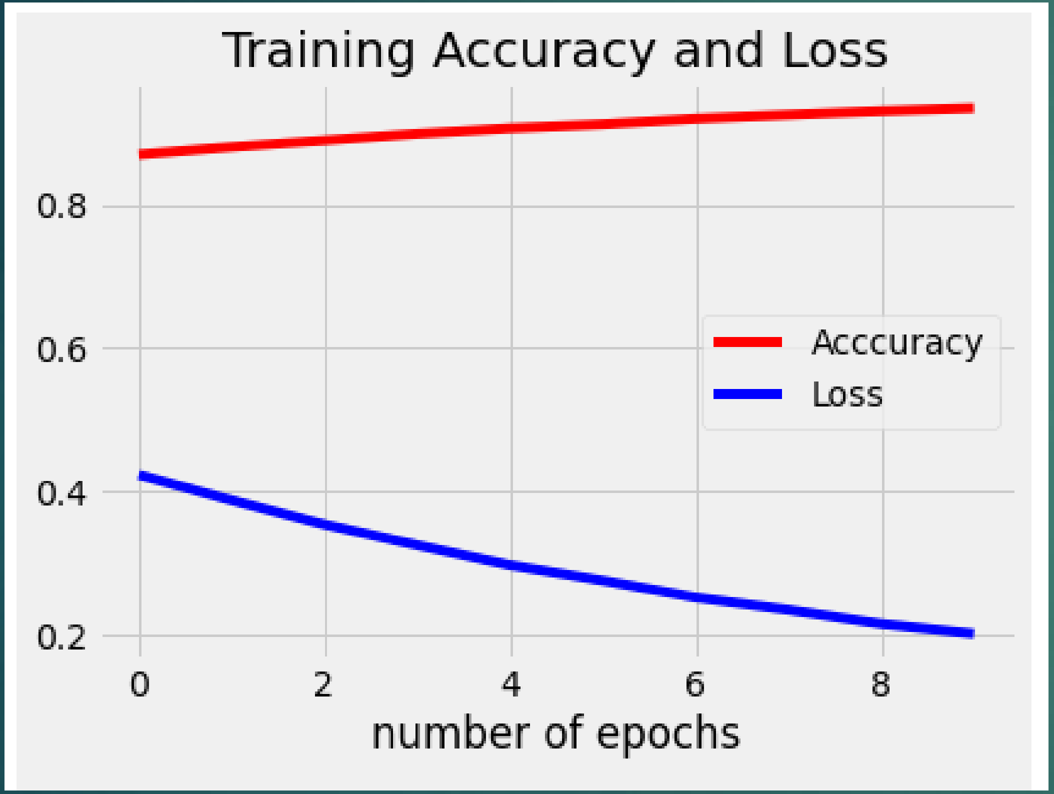
Model

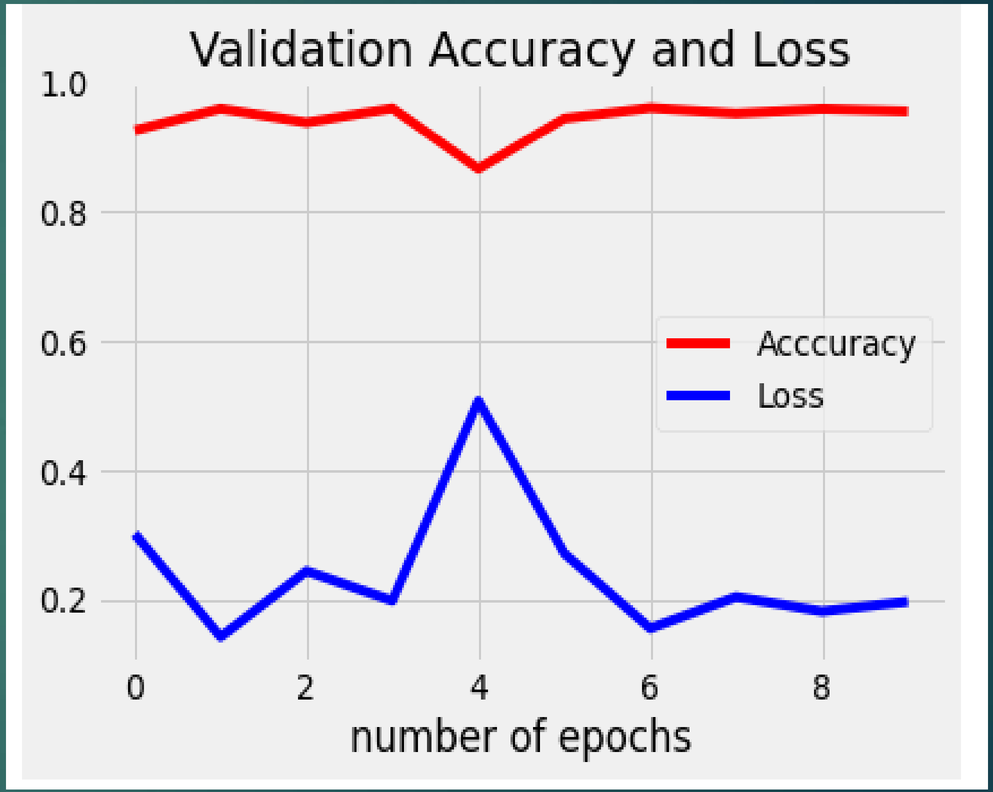
- We use the Deep Neural Network Classifier as our Training Model. After Data augmentation, we get
  - **129100** training images
  - **12630** testing images
  - **4410** validation images
- We use four con2D layers, two MaxPooling2D layers, two dense layers, dropout, and Batch Normalization, Flatten.
- **Activation = SoftMax** as the activation function in the output layer
- **Learning Rate = 0.001, Optimizer = Adam**
- **Batch size = 250 and Epochs = 10**

IMPLEMENTATION RESULTS

- After all the pre-processing of the images, we get the good quality of the image as compared to the original image. After Data augmentation, we get 129100 training examples instead of 34799, 12630 testing images, 4410 validation images.
- We get the Balanced Training Dataset to train the model.
- After training the model we get,
  - Training accuracy as **93.55%**,
  - Validation accuracy as **95.65%**,
  - Testing accuracy as **94.50%**.

Continued….


Training Accuracy and Loss
0.8
0.6
0.4
0.2
0
2
4
6
8
number of epochs
Acccuracy
Loss


Validation Accuracy and Loss
1.0
0.8
0.6
0.4
0.2
0
2
4
6
8
number of epochs
Acccuracy
Loss

CONCLUSION

- This paper proposes the Scaling, Normalization techniques, and Data Augmentation techniques like zoom, rotation, shear, Gaussian noise, color inversion, etc.
- It will almost remove all the unfavorable conditions like color, weather conditions, undesirable backgrounds, poor visibility because of lightning, shadows, blurring, and Improve the image quality.
- And also proposes Deep CNN with fewer number of Parameters. We get the Training accuracy as 93.55%, Validation accuracy as 95.65%, and Testing accuracy as 94.50%. This will give us the best result.

FUTURE SCOPE

- Here we use the Simple Convolutional Neural Network to train our model and we use scaling, normalization, and data augmentation techniques to improve the quality of the images.
- We can try different sharpening techniques. RGB to Gray, thresholding, Morphology and dilations is also applicable. We can try with the different filters like high pass filters, low pass filters, Ideal and Butterworth filters, laplacian filters, adaptive mean filters, etc
- We can try with the different Deep Learning Architectures like VGGNet, GoogleNet, Google Inception, Transfer Learning for Classification.
- Also try real-time traffic sign detection and recognition using a webcam or video.

Thank You