TRAFFIC SIGN CLASSIFIER

Project Objectives:

- Load the German Traffic Sign Dataset
- Explore, summarize and visualize the dataset
- Design, train, validate and test a CNN model architecture
- Predict new images using this model
- Analyze softmax probabilities of these new images
- Report the project results and details

Rubric Criteria:

This section deals how each rubric requirement was met in a step-by-step manner.

Writeup*:* This document serves the purpose of meeting the rubric criteria of submission files, along with other files.

Dataset Exploration

Dataset Summary: According to http://benchmark.ini.rub.de/?section=gtsrb&subsection=dataset , the dataset has more than 40 classes and 50,000 images. It is a large, lifelike database containing reliable ground truth data due to semi-automatic annotation. The imported data is a dictionary with 4 key/value data pairs; 'features' is a 4D array containing raw pixel data of the traffic sign images, 'labels' is a 1D array containing the label/class id of the traffic sign, 'sizes' is a list containing tuples representing the original width and height the image, 'coords' is a list containing tuples representing coordinates of a bounding box around the sign in the image.

Using the numpy library further details were explored and summarized below.

Number of training examples = 34799

Number of testing examples = 12630

Image data shape = (32, 32, 3)

Number of classes = 43

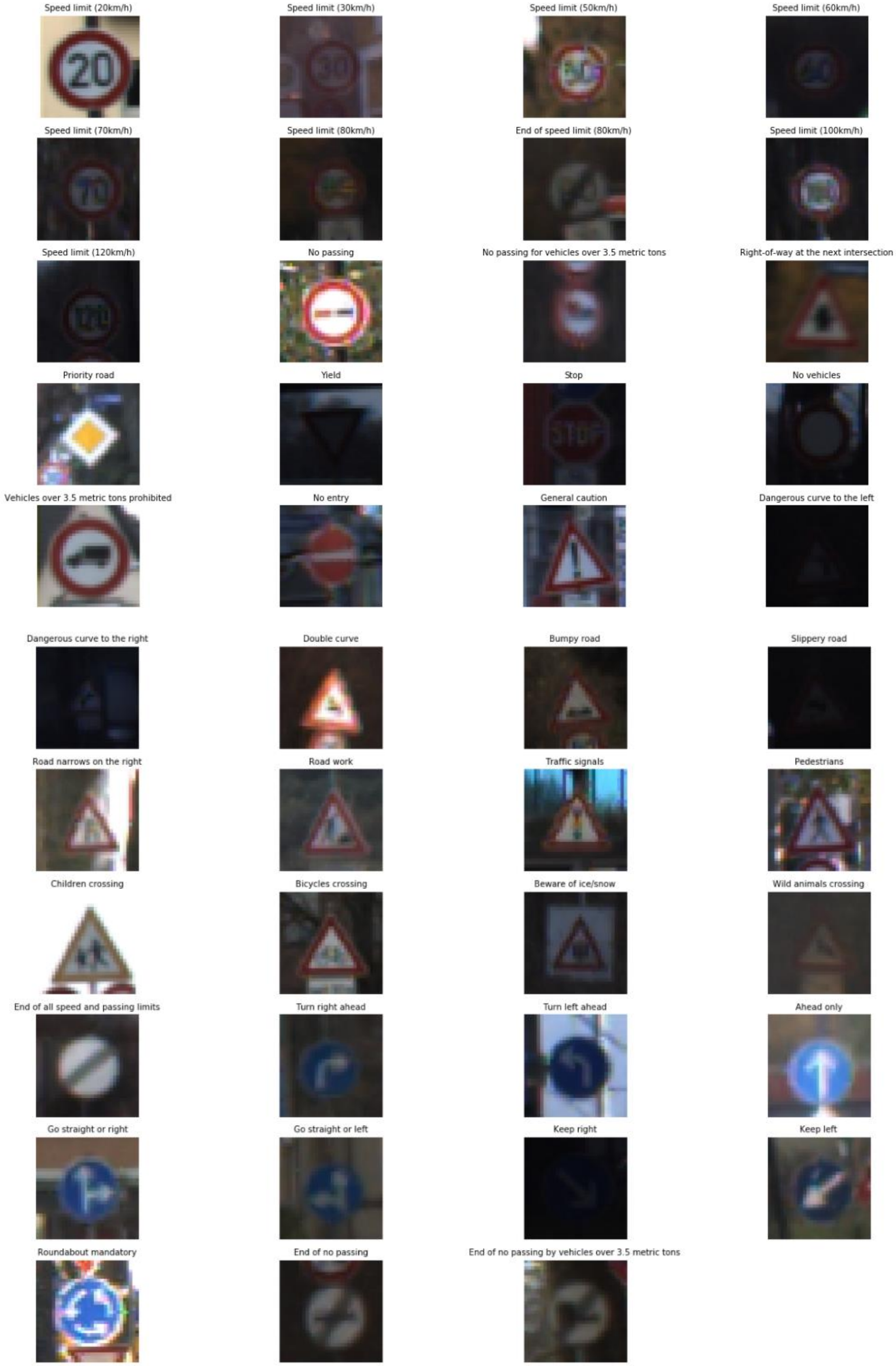
Shape of original image data = (34799, 2)

Maximum dimensions of original image = (243, 243)

Minimum dimensions of original image = (25, 25)

Exploratory Visualization:

All the traffic signs with their description are visualized below:


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 vehicles over 3.5 metric tons
Right-of-way at the next intersection
Priority road
Yield
Stop
No vehicles
Vehicles 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 vehicles over 3.5 metric tons

Design and Test a Model Architecture

Preprocessing: Three different color spaces and two normalization techniques were explored, namely, RGB, Gray, YUV color spaces and 'normalization around 128', min-max normalization. All these options are explored in training the model and based on the validation accuracy, 'grayscale' conversion and min-max normalization were finalized. The 'grayscale' preprocessing intuitively should make the neural network robust to color spaces as the traffic sign dataset is indifferent to colors. Additionally, the dataset was also shuffled.

Model Architecture: Initially, the LeNet architecture was used as the base architecture. However, during training, even on tweaking the hyper-parameters, the network was overfitting and the validation accuracy saturated at around 75%. To prevent overfitting, two dropout layers were added after the fully connected layers, with 'keep-probability' as an additional hyper-parameter to tune. The below table provides all the details.

Layer	**Description**
Input	32x32x1 images
Convolutional Layer 1	Strides of [1,1,1,1] VALID padding, 28x28x6 output
Activation Layer 1	ReLU
Max Pooling Layer 1	Kernel of [1,2,2,1], VALID padding, 14x14x6 output
Convolutional Layer 2	Strides of [1,1,1,1] VALID padding, 10x10x16 output
Activation Layer 2	ReLU
Max Pooling Layer 2	Kernel of [1,2,2,1], VALID padding, 5x5x16 output
Flat Layer	400 element array output
Fully Connected Layer 1	120 element array output
Dropout Layer 1	NA
Activation Layer 3	ReLU
Fully Connected Layer 2	84 element array output
Dropout Layer 2	NA
Activation Layer 4	ReLU
Final Fully Connected Layer 3	43 element array output representing 43 Labels

Model Training: An ADAM optimizer was used for to minimize the loss. The following hyper parameters yield satisfactory results.

Hyper-Parameter	**Value**
Epoch	100
Batch Size	256
Learning Rate	0.001
Keep Prob	0.5

Solution Approach: A satisfactory solution was reached by iteratively varying hyper-parameters until the model validation accuracy was improved. The 'keep probability' hyper parameter reduces overfitting but reaches saturation. Learning rate needs to be optimal for both efficient training time and performance. Epoch and batch size are varied accordingly to train the model sufficiently. The Epoch size was kept considerably high to improve training accuracy to 96%, even though the validation accuracy seemed to saturate earlier. However, there were small improvements in validation accuracy at this Epoch size. The following are the results.

Training Accuracy	97%
Validation Accuracy	95%
Test Accuracy	94%

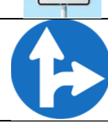
Test Model on New Images:

Acquiring New Images: Seven images were found on the web. They are as follows:

Figure 1 (a) Speed Limit 60kmph (b) Right-of-way at next intersection (c) Priority road (d) Stop (e) General caution (f) Children Crossing (g) Go straight or right

Among these, image (c) and image (f) should prove as difficult classifications for the network as the images deviate from the training dataset due to varied orientation and presence of a text board in conjunction with the sign board.

Performance on New Images:

Image Label	Image	Classification
3		True
11		True
12		True
14		True
18		True
28		False
36		False

The model successfully classified 5 out of 7 images correctly. However, this 71% accuracy is below the test accuracy of 94%. The model did well with label 12, but could not rightly classify label 28. However, the model surprisingly failed to classify the straightforward label 36.

<u>Model Certainty – Softmax Probabilities</u>: The softmax probabilities were computed to indicate certainty of the model. Only top three probabilities are listed as beyond three did not have any considerable impact. The results are shown below.

```
Label ID:  11
Prediction No. 1 is  11 with a Softmax probability of  100.0 %
Prediction No. 2 is  30 with a Softmax probability of  8.7017096809e-23 %
Prediction No. 3 is  27 with a Softmax probability of  2.41488763416e-28 %

Label ID:  3
Prediction No. 1 is  3 with a Softmax probability of  99.2880582809 %
Prediction No. 2 is  2 with a Softmax probability of  0.711943302304 %
Prediction No. 3 is  5 with a Softmax probability of  1.72911307494e-07 %

Label ID:  18
Prediction No. 1 is  18 with a Softmax probability of  100.0 %
Prediction No. 2 is  26 with a Softmax probability of  1.43488095466e-32 %
Prediction No. 3 is  27 with a Softmax probability of  2.84576897252e-36 %

Label ID:  36
Prediction No. 1 is  4 with a Softmax probability of  62.811422348 %
Prediction No. 2 is  1 with a Softmax probability of  8.84567424655 %
Prediction No. 3 is  36 with a Softmax probability of  8.8081471622 %

Label ID:  14
Prediction No. 1 is  14 with a Softmax probability of  99.9995112419 %
Prediction No. 2 is  1 with a Softmax probability of  0.000472736201118 %
Prediction No. 3 is  3 with a Softmax probability of  7.65705863159e-06 %

Label ID:  28
Prediction No. 1 is  42 with a Softmax probability of  73.6819922924 %
Prediction No. 2 is  23 with a Softmax probability of  24.8112365603 %
Prediction No. 3 is  20 with a Softmax probability of  0.769495824352 %

Label ID:  12
Prediction No. 1 is  12 with a Softmax probability of  100.0 %
Prediction No. 2 is  15 with a Softmax probability of  3.58310089482e-23 %
Prediction No. 3 is  40 with a Softmax probability of  6.55762274918e-25 %
```

All the right predictions were classified with great certainty, almost close to 100%. At the two false classifications of label 36 and label 28, the model failed completely as it could not predict the right classification even in top three probabilities. Therefore, the model can be further improved by adding augmented images to the training dataset including rotation, translation, different color spaces. This should improve the model to close to 100% accuracy.

Visualizing the Neural Network's State with Test Images:

Neural Networks are most often treated as black-boxes. However, at each layer the feature maps tend to recognize certain patterns, shapes, geometries in a hierarchical fashion. This was visualized at each layer of the Neural Network on a test image representing a label ID of 3, i.e., speed limit of 60kmph. The visualization is shown below.