Traffic Sign Recognition

Hyun-Koo KIM

studian@gmail.com

Rubric Points

This will be an overview of my project located here:
https://github.com/studian/SDC_HW02_TrafficSignClassifier.git

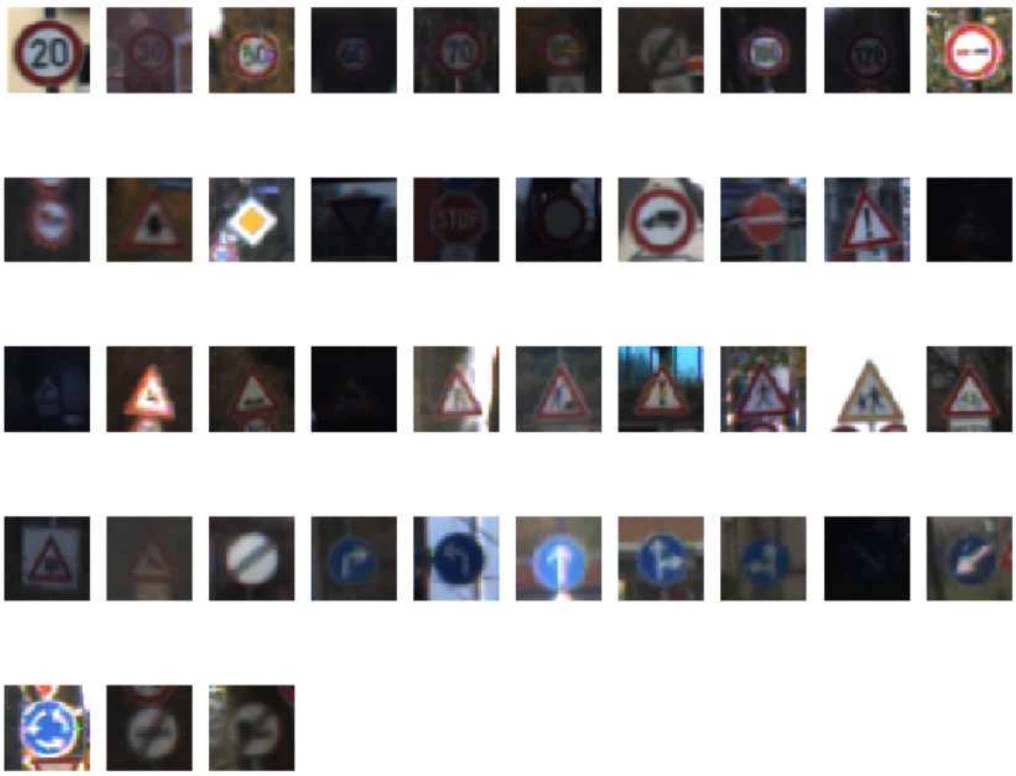
1. Data Set Summary &Exploration

1) Provide a basic summary of the data set. In the code, the analysis should be done using python, numpy and/or pandas methods rather than hardcoding results manually.

- I used the pandas library to calculate summary statistics of the traffic signs data set:
  - The size of training set is ? 34,799
  - The size of the validation set is ? 4,410
  - The size of test set is ? 12,630
  - The shape of a traffic sign image is ? 32x32x3
  - The number of unique classes/labels in the data set is ? 43

2) Include an exploratory visualization of the dataset.

- Visualize the German Traffic Signs Dataset using the pickled file(s). This is open ended, suggestions include: plotting traffic sign images, plotting the count of each sign, etc.
- It is shown 43 class images of train dataset



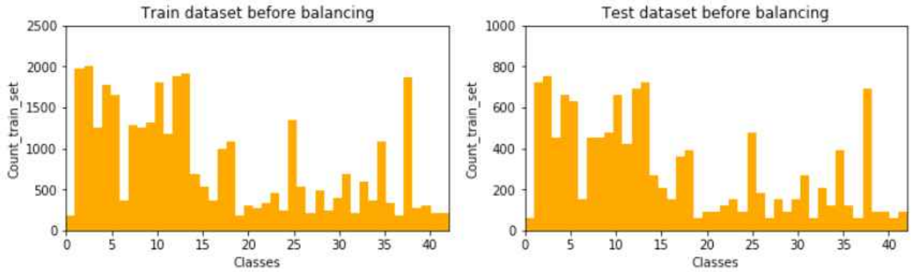
2. Design and Test a Model Architecture

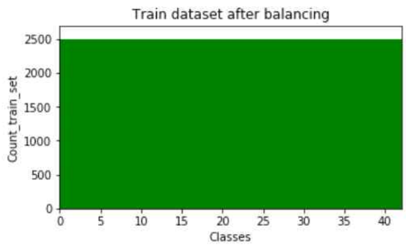
1) Describe how you preprocessed the image data. What techniques were chosen and why did you choose these techniques? Consider including images showing the output of each preprocessing technique. Pre-processing refers to techniques such as converting to grayscale, normalization, etc. (OPTIONAL: As described in the "Stand Out Suggestions" part of the rubric, if you generated additional data for training, describe why you decided to generate additional data, how you generated the data, and provide example images of the additional data. Then describe the characteristics of the augmented training set like number of images in the set, number of images for each class, etc.)

- Pre-process the Data Set (normalization, grayscale, etc.)

a) Dataset balancing

- I make balanced image datasets of 2500 elements in the each classes (all dataset)





b) Normalization

- all images of dataset make resize (32x32x3)
- When it add balanced image datasets,
- Randomly, equalize the histogram of the Y channel
- Randomly, image rotate

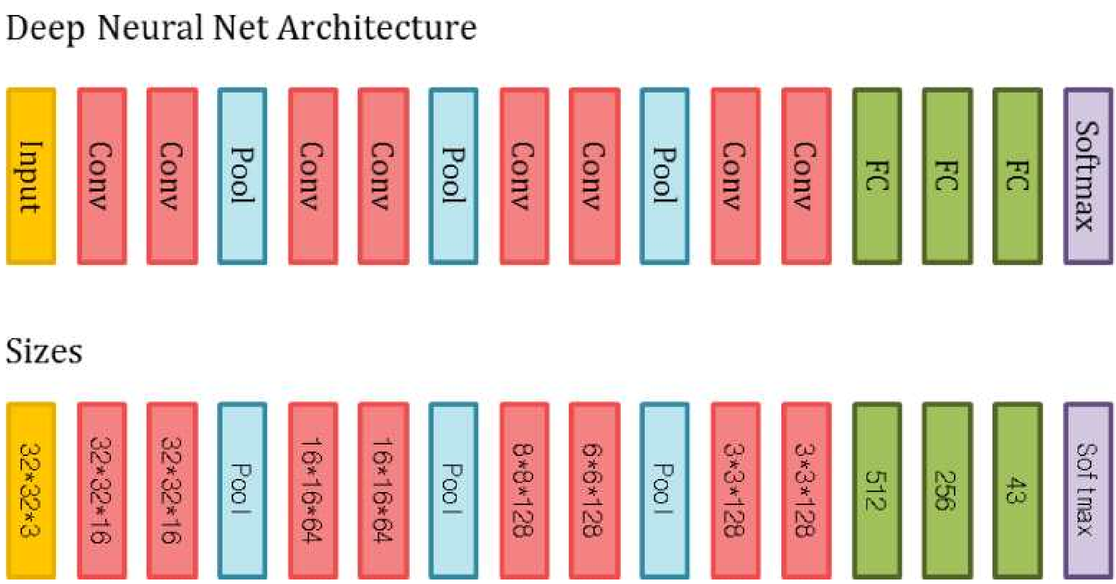
< It is shown 50 elements of balanced train dataset of 0 classID >

c) Shuffle data

- The data is sorted by class.
- I didn't use translation when make an additional test image.
- There were no effect of normalization(0, 255 -> 0, 1) because already all image's value range is same.
- split data into training and validation set (80 : 20)

2) Describe what your final model architecture looks like including model type, layers, layer sizes, connectivity, etc.) Consider including a diagram and/or table describing the final model.

My final model consisted of the following layers:



3) Describe how you trained your model. The discussion can include the type of optimizer, the batch size, number of epochs and any hyper parameters such as learning rate.

- BATCH_SIZE : 256
- EPOCHS : 10
- learning rate : 0.00001
- type of optimizer : ADAM
  - L2_loss
    = (tf.nn.l2_loss(weights['w_conv1']) + tf.nn.l2_loss(weights['w_conv2'])
    + tf.nn.l2_loss(weights['w_conv3']) + tf.nn.l2_loss(weights['w_conv4'])
    + tf.nn.l2_loss(weights['w_conv5']) + tf.nn.l2_loss(weights['w_conv6'])
    + tf.nn.l2_loss(weights['w_conv7']) + tf.nn.l2_loss(weights['w_conv8'])

+ tf.nn.l2_loss(weights['w_fc1']) + tf.nn.l2_loss(weights['w_fc2'])
+ tf.nn.l2_loss(weights['w_fc3']))

- cross_entropy = tf.nn.softmax_cross_entropy_with_logits(logits, one_hot_y)
- loss_operation = tf.reduce_mean(cross_entropy) + 0.00001 * L2_loss
- optimizer = tf.train.AdamOptimizer(learning_rate = rate)
- training_operation = optimizer.minimize(loss_operation)

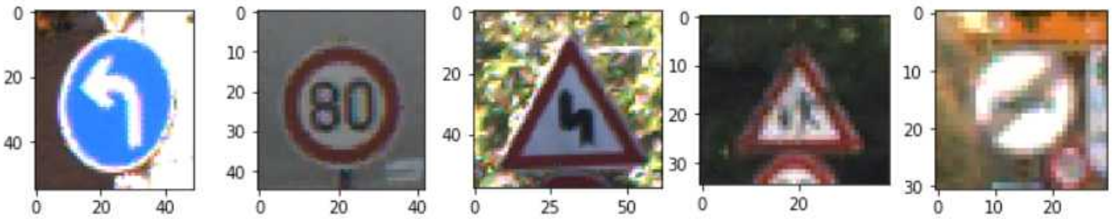
4) Describe the approach taken for finding a solution and getting the validation set accuracy to be at least 0.93. Include in the discussion the results on the training, validation and test sets and where in the code these were calculated. Your approach may have been an iterative process, in which case, outline the steps you took to get to the final solution and why you chose those steps. Perhaps your solution involved an already well known implementation or architecture. In this case, discuss why you think the architecture is suitable for the current problem.

- My final model results were:
  - Validation set accuracy : 0.995
  - Test set accuracy : 0.962

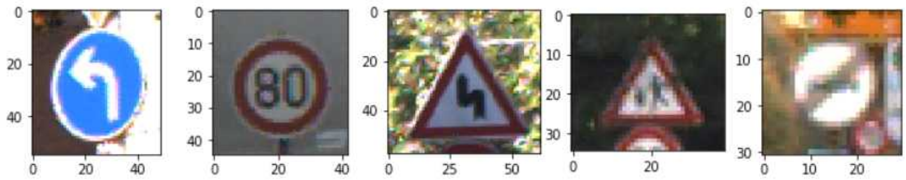
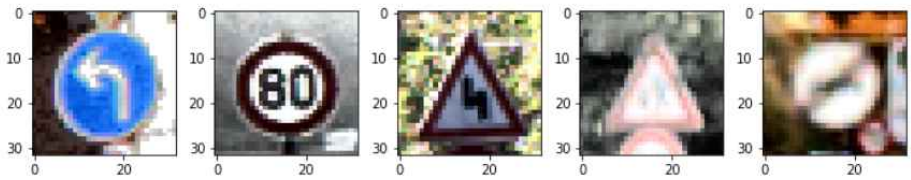
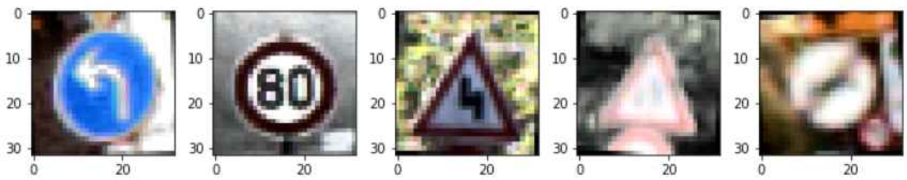
3. Test a Model on New Images

1) Choose five German traffic signs found on the web and provide them in the report. For each image, discuss what quality or qualities might be difficult to classify.

- Here are five German traffic signs that I found on the web:



- Here are fifteen German traffic signs in my test using the five traffic signs:

Types	Images				
Original					
Equalization in the Y channel					
Rotation					
Rotation Degree	2.775	-0.923	-5.01	-6.99	6.85

2) Discuss the model's predictions on these new traffic signs and compare the results to predicting on the test set. At a minimum, discuss what the predictions were, the accuracy on these new predictions, and compare the accuracy to the accuracy on the test set (OPTIONAL: Discuss the results in more detail as described in the "Stand Out Suggestions" part of the rubric).

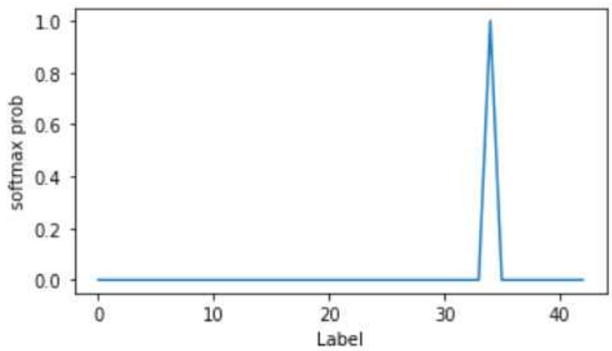
- Here are the results of the prediction:

No.	Image	Types	Prediction
1	Turn left ahead	Original	Turn left ahead
2	Speed limit (80km/h)	Original	Speed limit (80km/h)
3	Double curve	Original	Double curve
4	Children crossing	Original	Children crossing
5	End of no passing	Original	End of no passing
6	Turn left ahead	Eq. Y ch	Turn left ahead
7	Speed limit (80km/h)	Eq. Y ch	Speed limit (80km/h)
8	Double curve	Eq. Y ch	Double curve
9	Children crossing	Eq. Y ch	Children crossing
10	End of no passing	Eq. Y ch	End of no passing
11	Turn left ahead	Rotate	Turn left ahead
12	Speed limit (80km/h)	Rotate	Speed limit (80km/h)
13	Double curve	Rotate	Double curve
14	Children crossing	Rotate	Children crossing
15	End of no passing	Rotate	End of no passing

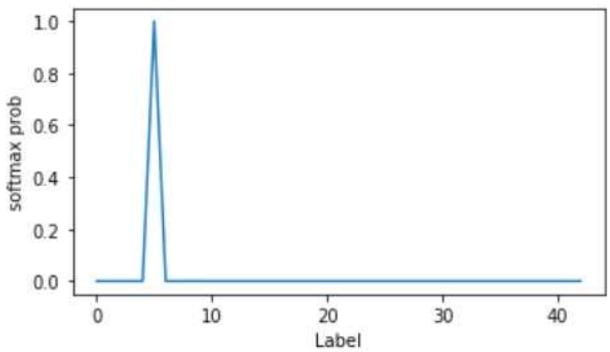
- The model was able to correctly guess 15 of the 15 traffic signs, which gives an accuracy of 100%.

3) Describe how certain the model is when predicting on each of the five new images by looking at the softmax probabilities for each prediction. Provide the top 5 softmax probabilities for each image along with the sign type of each probability. (OPTIONAL: as described in the "Stand Out Suggestions" part of the rubric, visualizations can also be provided such as bar charts)

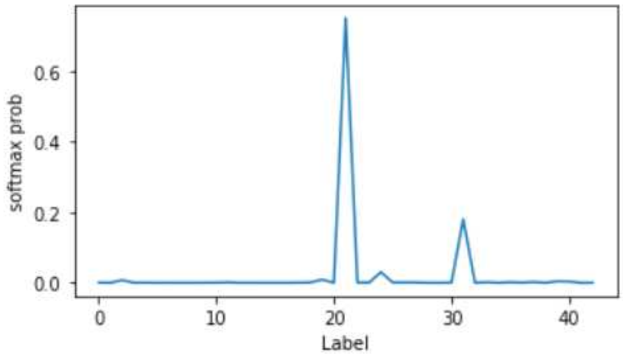
- Test image (1) : Turn left ahead (Original)



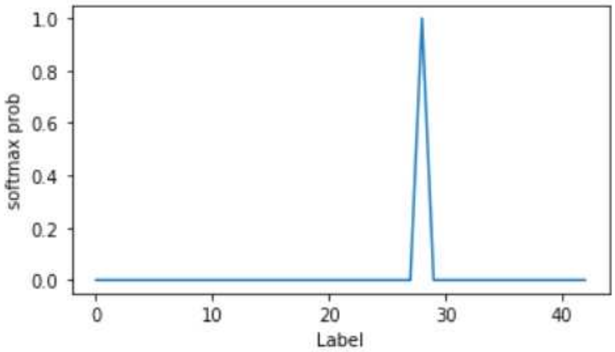
- Test image (2) : Speed limit (80km/h) (Original)



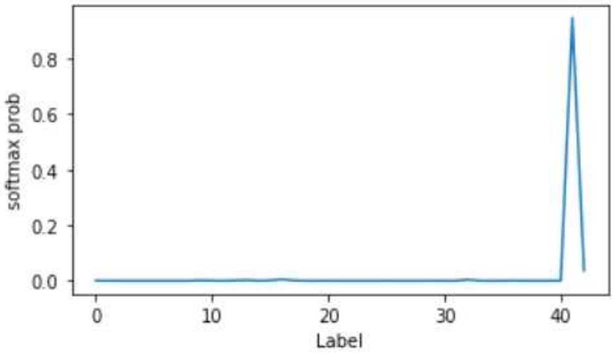
- Test image (3) : Double curve (Original)



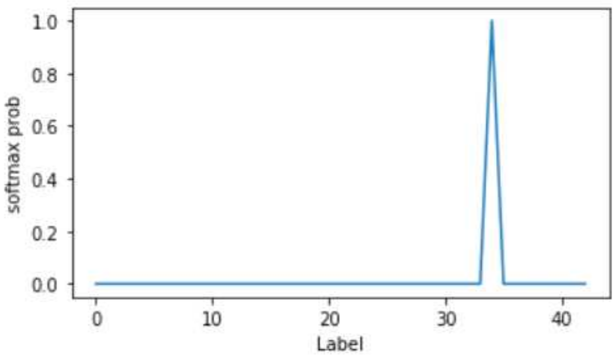
- Test image (4) : Children crossing (Original)



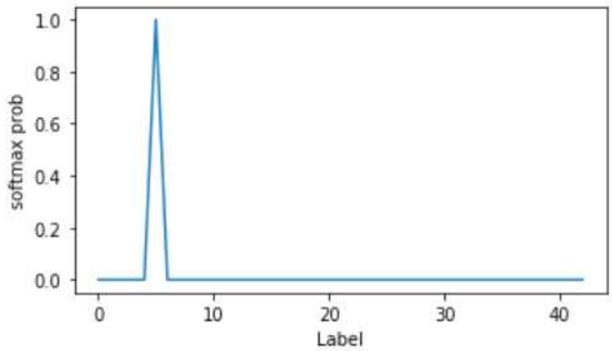
- Test image (5) : End of no passing (Original)



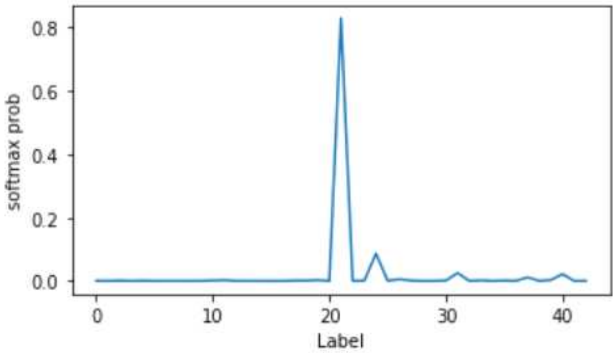
- Test image (6) : Turn left ahead (Eq. Y ch)



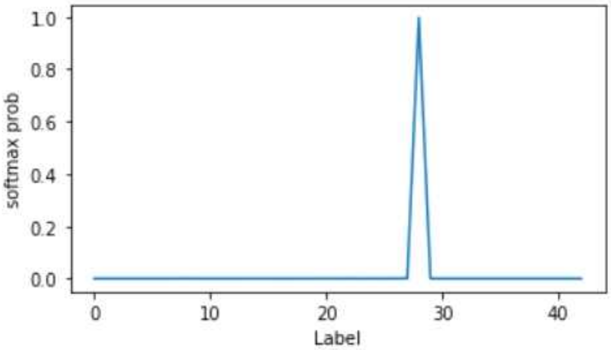
- Test image (7) : Speed limit (80km/h) (Eq. Y ch)



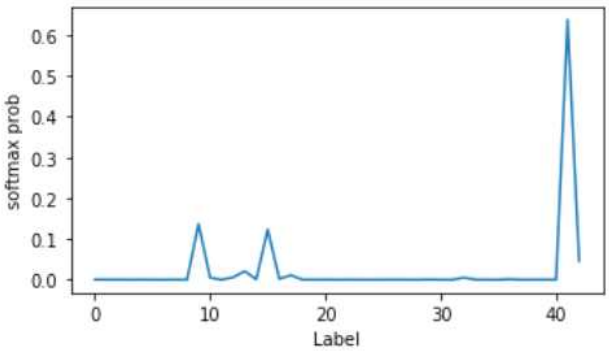
- Test image (8) : Double curve (Eq. Y ch)



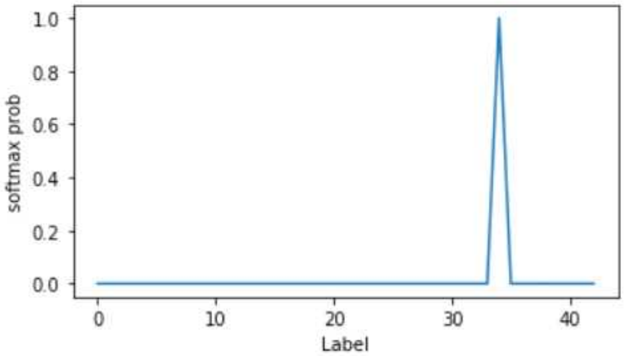
- Test image (9) : Children crossing (Eq. Y ch)



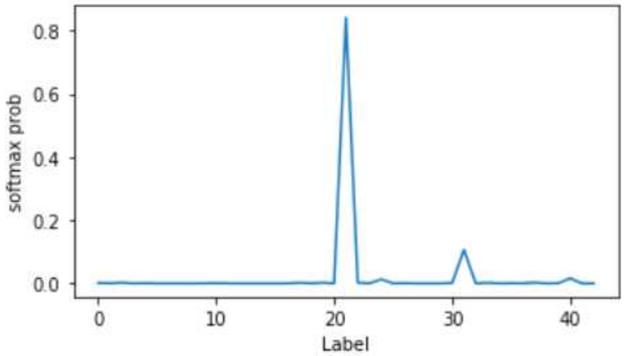
- Test image (10) : End of no passing (Eq. Y ch)



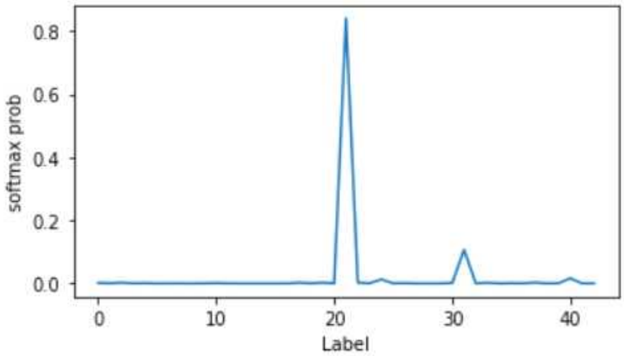
- Test image (11) : Turn left ahead (Rotate)



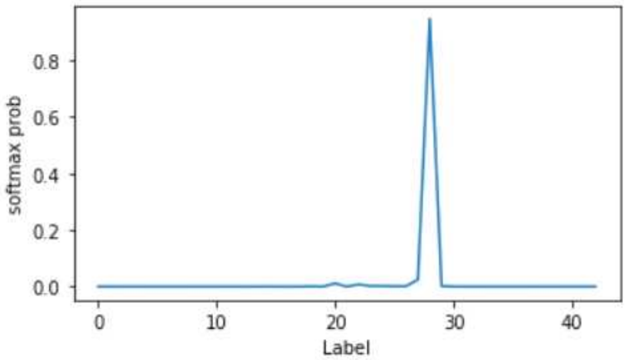
- Test image (12) : Speed limit (80km/h) (Rotate)



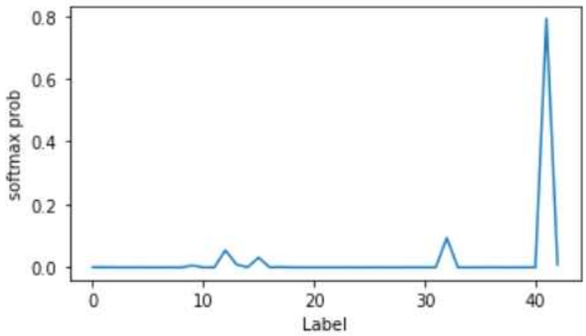
- Test image (13) : Double curve (Rotate)



- Test image (14) : Children crossing (Rotate)



- Test image (15) : End of no passing (Rotate)



- The top five soft max probabilities in the 'Turn left ahead' traffic signs :

No.	Types	Prediction	
		Probability	Result
1	Original	99.994%	Turn left ahead
		0.001%	Ahead only
		0.001%	Roundabout mandatory
		0.001%	Priority road
		0.001%	Go straight or left
6	Eq. Y ch	99.975%	Turn left ahead
		0.012%	Keep right
		0.002%	Go straight or right
		0.002%	Go straight or left
		0.002%	Priority road
11	Rotate	99.985%	Turn left ahead
		0.012%	Keep right
		0.001%	Go straight or left
		0.000%	Priority road
		0.000%	Go straight or right

- The top five soft max probabilities in the 'Speed limit (80km/h)' traffic signs :

No.	Types	Prediction	
		Probability	Result
2	Original	99.992%	Speed limit (80km/h)
		0.004%	Speed limit (120km/h)
		0.001%	Speed limit (50km/h)
		0.001%	End of speed limit (80km/h)
		0.000%	Speed limit (60km/h)
7	Eq. Y ch	99.993%	Speed limit (80km/h)
		0.002%	Speed limit (100km/h)
		0.001%	End of speed limit (80km/h)
		0.001%	Speed limit (60km/h)
		0.000%	Speed limit (120km/h)
12	Rotate	99.992%	Speed limit (80km/h)
		0.003%	Speed limit (100km/h)
		0.002%	End of speed limit (80km/h)
		0.001%	Speed limit (60km/h)
		0.001%	Speed limit (120km/h)

- The top five soft max probabilities in the 'Double curve' traffic signs :

No.	Types	Prediction	
		Probability	Result
3	Original	75.093%	Double curve
		18.026%	Wild animals crossing
		3.003%	Road narrows on the right
		0.895%	Dangerous curve to the left
		0.741%	Speed limit (50km/h)
8	Eq. Y ch	82.881%	Double curve
		8.642%	Road narrows on the right
		2.494%	Wild animals crossing
		2.151%	Roundabout mandatory
		1.127%	Go straight or left
13	Rotate	84.020%	Double curve
		10.648%	Wild animals crossing
		1.612%	Roundabout mandatory
		1.279%	Road narrows on the right
		0.274%	Go straight or left

- The top five soft max probabilities in the 'Children crossing' traffic signs :

No.	Types	Prediction	
		Probability	Result
4	Original	100.000%	Children crossing
		0.000%	Bicycles crossing
		0.000%	Right-of-way at the next intersection
		0.000%	Speed limit (120km/h)
		0.000%	General caution
9	Eq. Y ch	99.708%	Children crossing
		0.091%	Pedestrians
		0.043%	Bumpy road
		0.033%	Dangerous curve to the right
		0.029%	Speed limit (120km/h)
14	Rotate	94.507%	Children crossing
		2.456%	Pedestrians
		1.137%	Dangerous curve to the right
		0.765%	Bumpy road
		0.237%	Road narrows on the right

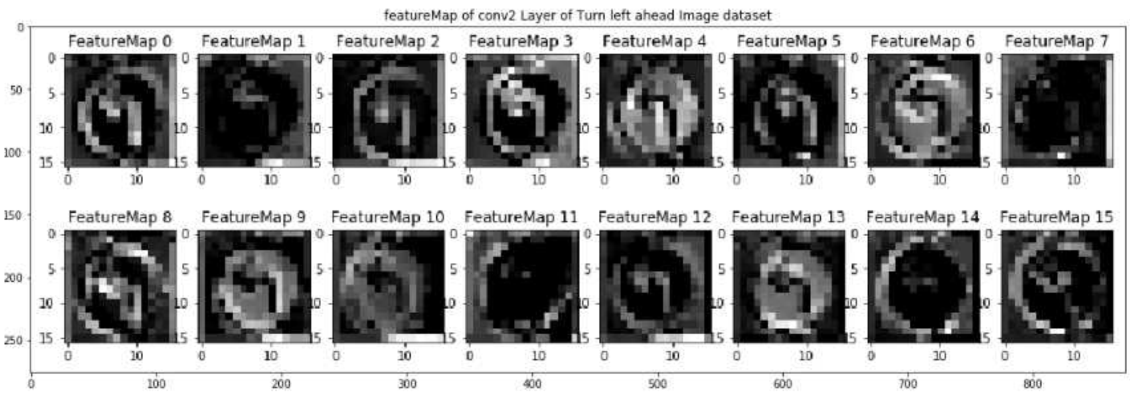
- The top five soft max probabilities in the 'End of no passing' traffic signs :

No.	Types	Prediction	
		Probability	Result
5	Original	94.453%	End of no passing
		3.628%	End of no passing by vehicles over 3.5 metric tons
		0.526%	Vehicles over 3.5 metric tons prohibited
		0.427%	End of all speed and passing limits
		0.236%	Yield
10	Eq. Y ch	63.821%	End of no passing
		13.681%	No passing
		12.334%	No vehicles
		4.564%	End of no passing by vehicles over 3.5 metric tons
		2.094%	Yield
15	Rotate	79.151%	End of no passing
		9.387%	End of all speed and passing limits
		5.474%	Priority road
		3.134%	No vehicles
		0.930%	Yield

4. (Optional) Visualizing the Neural Network (See Step 4 of the Ipython notebook for more details)

1) Discuss the visual output of your trained network's feature maps. What characteristics did the neural network use to make classifications?

- It is shown results of feature-map of conv layer as follows:



- It seems to be compose of low-pass-filter, high-pass-filter, gradient filter, etc of weights of convolution layers